UNIVERSITÄT ZU LÜBECK

From the Institute for Electrical Engineering in Medicine
of the University of Lübeck

Director: Prof. Dr. Philipp Rostalski

Lightweight, Transparent, and Uncertainty-Aware Deep Learning for Diabetic Retinopathy Grading

Dissertation
for Fullfillment of
Requirements
for the Doctoral Degree
of the University of Lübeck

from the Department of Computer Sciences and Technical Engineering

Submitted by

Marlin Sebastian Siebert, M.Sc.
from Wolfenbüttel

Lübeck, 2024

First referee: Prof. Dr. Philipp Rostalski

Sceond referee: Prof. Dr.-Ing. habil. Marcin Grzegorzek

Date of oral examination: February 3rd, 2025

Approved for printing. Lübeck, February 18th, 2025

Abstract

The rising prevalence of patients with diabetes mellitus and diabetic retinopathy (DR) has sparked extensive research on automated, deep learning-based screening systems for DR to improve access to medical care and prevent loss of vision. These, however, commonly lack transparency, limiting their adoption in clinical practice. Besides this societal demand for transparency, regulations on the use of artificial intelligence (AI) in medicine are becoming more stringent, i.e., the recently approved EU's AI Act explicitly requires transparency, interpretability, and human oversight.

This thesis addresses these challenges by developing a transparent, interpretable, uncertainty-aware, and lightweight deep neural network (NN) for predicting the presence and severity of DR from retinal images by incorporating clinical expert knowledge and diagnostic guidelines. To this end, the two-stage concept bottleneck model (CBM) is adapted and a new soft-CBM architecture is designed to enhance predictive performance despite concept incompleteness. These models additionally provide visualizations of DR-related biomarkers that can be used as explanation for the predicted DR severity. Moreover, to enable mobile application on edge devices, improve detection of model failures, and enhance patient safety, the benefit of deep kernel learning (DKL), i.e., a Bayesian NN exploiting Gaussian processes, and various lightweight U^2-Net variants utilized for lesion segmentation on the system's uncertainty calibration and cost-efficiency are explored.

This work demonstrates the performance of the proposed concept-based model to be close to par compared to a state-of-the-art, black-box, vanilla NN — despite the decoupled two-stage model training conducted within this thesis. The proposed lightweight U^2-Nets provide significant performance improvements over both the vanilla U-Net and U^2-Net while reducing the computational load and being competitive with the literature. Hence, the computational cost of the proposed concept-based system is comparable to a vanilla NN optimized for mobile use but inherently provides detailed explanations. An in-depth analysis demonstrates that the adapted DKL framework provides well-calibrated uncertainty estimates when accounting for the feature collapse observed in the literature. Particularly, its uncertainty awareness is found to be superior to commonly adopted Bayesian NNs for task-related out-of-distribution data. Although transferability of the results into clinical practice remains to be demonstrated, the improvements achieved represent a significant step forward in bridging the gap to a mobile applicable, transparent, and uncertainty-aware DR grading system that enables human oversight and intervention of erroneous model predictions in alignment with regulatory requirements.

Zusammenfassung

Die Zunahme Erkrankter mit diabetischer Retinopathie (DR) hat zu zahlreichen Forschungsprojekten geführt, mit dem Ziel die Patientenversorgung mittels automatisierter, Deep-Learning-gestützter DR-Screeningsysteme zu verbessern und Erblindungen zu verhindern. Diese sind jedoch meist intransparent, was ihre Akzeptanz in der klinischen Praxis einschränkt. Neben der gesellschaftlichen Forderung nach Transparenz werden auch die Zulassungsvorschriften für Künstliche Intelligenz (KI) in der Medizin strenger. So verpflichtet das EU KI-Gesetz ausdrücklich zu Transparenz, Interpretierbarkeit und menschlicher Aufsicht.

Ziel dieser Arbeit ist daher die Entwicklung eines transparenten, interpretierbaren, unsicherheitsbewussten und leichtgewichtigen tiefen neuronalen Netzes (NN) zur Vorhersage von DR unter Berücksichtigung von Expertenwissen und Leitlinien. Dazu wird das zweistufige Concept Bottleneck Model (CBM) adaptiert und eine neue soft-CBM-Architektur entwickelt, um die Vorhersagegenauigkeit trotz unvollständiger Konzepte zu verbessern. Die Modelle liefern Visualisierungen von DR-relevanten Biomarkern zur Erklärung des vorhergesagten DR-Schweregrades. Um zudem eine mobile Anwendung auf Edge-Geräten, verbesserte Fehlererkennung und erhöhte Patientensicherheit zu ermöglichen, werden zusätzlich die Vorteile des Deep Kernel Learning (DKL), eines auf Gauß-Prozessen basierenden Bayes'schen NN, und verschiedener leichtgewichtiger U^2-Net-Varianten für die Läsionssegmentierung hinsichtlich ihrer Unsicherheitskalibrierung und Kosteneffizienz untersucht.

Diese Arbeit zeigt, dass dieses System im Vergleich zu einem State-of-the-Art Black-Box NN eine nahezu gleichwertige Genauigkeit erbringt, trotz des unabhängigen Trainings beider Teilmodelle. Die entwickelten leichtgewichtigen U^2-Nets verbessern die Segmentierung im Vergleich zu dem regulären U-Net als auch dem originalen U^2-Net, während sie die Rechenkomplexität reduzieren und auf dem Niveau der Ergebnissen der Literatur liegen. Daher bietet das vorgeschlagene konzeptbasierte System eine vergleichbare Rechenkomplexität wie ein für den mobilen Einsatz optimiertes NN und liefert dabei inhärent detaillierte Erklärungen. Eine eingehende Analyse des DKLs zeigt eine gut kalibrierte Vorhersageunsicherheit, wenn der in der Literatur beobachtete Merkmalskollaps verhindert wird. Insbesondere für aufgabenverwandte, aber verteilungsfremde Daten erweist sich das DKL als den üblicherweise verwendeten Bayes'schen NNs überlegen. Obwohl die Übertragbarkeit der Ergebnisse in die klinische Praxis noch zu zeigen ist, stellen die erzielten Verbesserungen einen bedeutenden Schritt hin zu einem mobil anwendbaren, transparenten und unsicherheitsbewussten DR-Vorhersagesystem dar, welches zugleich Kontrolle und Intervention bei fehlerhaften Vorhersagen im Einklang mit den gesetzlichen Anforderungen ermöglicht.

Acknowledgments

First and foremost, I would like to start by expressing my gratitude to my supervisor Philipp Rostalski for his excellent mentorship, guidance, and support, which have been of great value to my research and personal growth. The freedom to explore and follow my own ideas and interests as well as the encouragement in my work have made this journey both enjoyable and rewarding.

I am profoundly grateful for the opportunity to work on the PASBADIA project, which was kindly funded by the Joachim Herz Foundation. The work on this project has not only been instrumental in the completion of this thesis, but has also significantly enhanced my skills and knowledge in the field.

A special thanks goes to my colleagues at the IME and IMTE, especially Jan Graßhoff and Lukas Boudnik, for creating a fantastic working environement. The fruitful discussions and continuous support over the years have been vital for my progress and the successful completion of this journey. I would also like to thank all reviewers for their valuable feedback, which helped to bring this thesis to a great end.

I am immensely grateful to my dear parents, family, and friends for their constant emotional support, encouragement, and, when necessary, timely distractions that helped me maintain my sanity throughout this demanding process.

Finally, a special note of appreciation and heartfelt thanks goes to Clarissa Pfeufer for her incredible patience, understanding, and endless support. Her belief in me and her continued backing have been a major cornerstone in the successful completion of this thesis.

Thank you all for your contributions and being a part of this journey.

— Marlin Sebastian Siebert, March 2025

Contents

1 | Introduction

> *“ Our future is a race between the growing power of our technology and the wisdom with which we use it. Let’s make sure that wisdom wins. ”*
>
> *— Stephen Hawking (Zeitgeist 2015) [1]*

rise of deep learning

After the progress in research on scaling artificial neural networks (NNs) to real-world tasks slowed down at the end of the 20th century [2], the breakthrough of deep learning (DL), and convolutional neural networks (CNNs) was achieved [3] starting in 2006 [4], and later on in 2012 [5], respectively. This was enabled by the increasing computing power, the emergence of high-performance hardware accelerators such as graphics processing units (GPUs), and the rapid increase in the available amount of digital data [3]. Due to the fast-growing performance of deep neural networks (DNNs), DL quickly became the state-of-the-art paradigm for data-driven modeling inside the field of artificial intelligence (AI), frequently replacing traditional machine learning (ML) methods. Therefore, it quickly has found its way into rather non-critical domains with low regulatory requirements, such as voice assistants and personal recommendation systems utilized in advertising, online shopping, and entertainment services. Most recently, the advent of generative AI models such as large language models (LLMs) and large multimodal models (LMMs) once again led to remarkable progress in image, text, and video creation while simultaneously fueling the debate about the potential misuse and the risks AI poses to society and humanity.

[2] Hinton and Salakhutdinov, “Reducing the dimensionality of data with neural networks” (2006)

[3] LeCun *et al.*, “Deep learning” (2015)

[4] Hinton *et al.*, “A fast learning algorithm for deep belief nets” (2006)

[5] Krizhevsky *et al.*, “ImageNet classification with deep convolutional neural networks” (2012)

1.1 Context and motivation

deep learning in medicine

With further improvements in the predictive performance of NNs and the growth of the volume of diagnostic imaging being higher than experts can cope with [6], the first DL-based decision support systems emerged in the

[6] Peng *et al.*, “Radiologist burnout: trends in medical imaging utilization under the national health insurance system with the universal code bundling strategy in an academic tertiary medical centre” (2022)

clinical domain, particularly in the field of radiology. About 76 % of all AI-based systems cleared by the United States' (U.S.) Food and Drug Administration (FDA) (as of 06/13/2024) were for use in radiology [7]. Thereby, the pioneering DL systems cleared for clinical use are primarily applied to assist practitioners in decision-making, triaging, image processing, and therapy planning [8], [9]. In 2018, the very first DL-supported system to be used without human oversight called LumineticsCore (formerly IDx-DR) was approved by the FDA for use in the high-stakes medical domain that screens patients suffering from diabetes mellitus (DM) for the presence of diabetic retinopathy (DR) from retinal images [10].

[7] U.S. Food & Drug Administration, "Artificial Intelligence and Machine Learning (AI/ML)-Enabled Medical Devices" (2024)

[8] Zhu *et al.*, "The 2021 landscape of FDA-approved artificial intelligence/machine learning-enabled medical devices: an analysis of the characteristics and intended use" (2022)

[9] Lyell *et al.*, "How machine learning is embedded to support clinician decision making: an analysis of fda-approved medical devices" (2021)

[10] Abràmoff *et al.*, "Improved automated detection of diabetic retinopathy on a publicly available dataset through integration of deep learning" (2016)

disadvantages of deep learning

However, despite the great performance of neural networks, translation of DL-supported AI systems into clinical practice remains challenging. A study examining 49 FDA cleared, AI-supported devices in 2021 found only 20 explicitly stating the use of DL-based methods [9]. This is — among other reasons such as open questions about the liability — linked to a lack of general guarantees for robustness and model generalization, as NNs are prone to overfit the training data and often rely on *spurious correlations* due to their high flexibility as universal function approximators [11]. Accordingly, the performance on real-world data and different patient cohorts as well as the robustness for unexpected events have to be thoroughly evaluated before applying the automated decision systems in the high-stakes medical domain [11]. Moreover, there are strong concerns about the lack of transparency and explainability of AI-based systems, particularly in context with their usage in highly safety-critical domains such as precision medicine, which impedes the translation of the methods into practice [12], [13]. In particular, due to the *black-box* characteristic of particularly deep NNs and the model's high complexity, it is difficult if not impossible to interpret the learned underlying functions, limiting the ability to trace the reasoning behind individual predictions. Furthermore, NNs may provide highly overconfident predictions, extrapolate widely, and typically predict with high confidence in regions far away from the training data. Hence, they often provide unreliable and uncalibrated *uncertainty* estimates and are not transparently communicating whether their prediction is based on high evidence given the training data or not.

[11] Petersen *et al.*, "Responsible and regulatory conform machine learning for medicine: a survey of challenges and solutions" (2022)

[12] Wong and Bressler, "Artificial intelligence with deep learning technology looks into diabetic retinopathy screening" (2016)

[13] Barredo Arrieta *et al.*, "Explainable artificial intelligence (XAI): concepts, taxonomies, opportunities and challenges toward responsible AI" (2020)

regulations and responsible AI

Due to these limitations and the potential threat that AI might increasingly pose to societal well-being, a significantly growing community advocates the concept of *responsible AI* (RAI) and of fairness, accountability,

and transparency in machine learning (FAccTML) [11], [14]. Following these demands, the European Union (EU) this year passed the AI Act [15] to regulate the use of AI-based systems to prevent, amongst other things, potential risks to safety, health, and fundamental rights. This regulation specifies a classification scheme that prohibits AI systems with unacceptable risks like social scoring while permitting the distribution and usage of AI systems with minimal or no risks without restrictions. Medical AI-based applications are classified in the second highest class according to this scheme, i.e., they are deemed to pose a high safety risk. Hence, with the enaction of the regulatory framework, a conformity assessment of systems intended for use in the EU according to the proposed AI regulations will be required prior to distribution, in addition to compliance with the EU medical device regulation (MDR) [16]. Successfully transferring AI-based algorithms into clinical practice will require proper risk management, an appropriate degree of *transparency*, and *human oversight* besides the high, evidence-based clinical accuracy and robustness. A central part is the specification of the user's obligations to supervise the system as well as the minimization and prevention of potential failures by means of transparent, explainable, and trustworthy AI algorithms.

[14] Rajpurkar *et al.*, "AI in health and medicine" (2022)

[15] European Commission, Directorate-General for Communications Networks, Content and Technology, "Artificial Intelligence Act" (2021)

[16] European Parliament, Council of the European Union, "Medical Device Regulation" (2023)

"*High-risk AI systems shall be designed and developed in such a way to ensure that their operation is sufficiently transparent to enable [users] to interpret the system's output and use it appropriately.*"—*AI Act (Article 13, §1), European Commission* [15]

"*High-risk AI systems shall be designed and developed in such a way, including with appropriate human-machine interface tools, that they can be effectively overseen by natural persons during the period in which the AI system is in use.*"—*AI Act (Article 14, §1), European Commission* [15]

Moreover, already existing regulations — at least implicitly — *already* demand transparent and explainable AI systems. As an example, the MDR requires the implementation of a quality management system, which must entail post-market surveillance. This includes the continuous monitoring of the system, as well as the reporting and analysis of unexpected outcomes or failures. The implementation of such a management system would benefit from a certain degree of explainability. The EU's General Data Protection Regulation (GDPR) [17], which is not only having an impact on the EU but influences worldwide product development [18], stipulates the right to demand an explanation for decisions to users being affected by automatic decision systems [19]. This technically renders black-box AI systems such as NNs without further explanations non-compliant with the GDPR [11]. Hence, AI-supported medical devices have not only to meet the requirements in the context of high *clinical validity* but — among other ethical aspects — also the demand for a *transparent, self-explanatory* usage of the system that allows for assessing the validity of the predictions.

[17] European Parliament, Council of the European Union, "General Data Protection Regulation" (2016)

"*[The controller shall provide information about] the existence of automated decision-making [and] meaningful information about the logic involved [...].*"—*GDPR (Article 13, §1f), European Parliament* [17]

[18] Ryngaert and Taylor, "The GDPR as global data protection regulation?" (2020)

[19] Goodman and Flaxman, "European Union regulations on algorithmic decision making and a "right to explanation"" (2017)

implications for DR screening

The implications of, e.g., the EU's regulatory framework for the use of AI can be illustrated well by reusing the DR screening application from

above: Patients with DM are encouraged to meet a regular, annual or biennial ophthalmoscopic examination to screen for the onset and monitor the progression of DR [20]–[23]. However, patients living in countries with insufficient medical coverage, or in rural areas often do not have a specialist nearby and, therefore, might not attend these examinations at all. A study conducted between 2010 to 2014 showed that only about 15 % to 26 % of U.S. patients with DM met the screening routine recommended by the American Diabetes Association [24]. The economic impact of eliminating this barrier for diabetic patients in the U.S. in 1994 was estimated to sum to USD$624 million annually assuming about 50 % of the patients with DM did not have access to preventive eye care [25]. Another study conducted in 2017 observed a lack of capacity in ophthalmoscopic healthcare, which led to delayed care and resulted in patients in the United Kingdom (UK) suffering from worsened visual acuity and vision loss [26]. Therefore, a screening routine conducted by, e.g., general practitioners could significantly improve access to retinal examination, reduce the workload for ophthalmologists, and improve medical coverage [23].

To support the former in conducting the retinal examination, mobile, AI-based, automatic screening systems such as LumineticsCore could be exploited. Although ophthalmologists have the required knowledge and training to verify and challenge the system's predictions, non-expert users may not be able to meet the demands of supervision due to a lack of specialized knowledge as well as years of experience and training in DR screening and grading. It becomes obvious, that the demand for human oversight, i.e., the supervisor should

> “ *[...] be able to decide, in any particular situation, not to use the high-risk AI system or otherwise disregard, override or reverse the output of the high-risk AI systems, [...]* ”
>
> — *AI Act (Article 14, §4d) [15]*

is closely linked to the demand for transparency and interpretability, as a sufficiently informed decision is not possible without appropriate explanations of the system's output. Furthermore, an informed decision also inevitably comprises communicating uncertainty related to the prediction to prevent potential failures caused by blind trust in the automated diagnosis system.

[20] American Diabetes Association Professional Practice Committee, “4. Comprehensive medical evaluation and assessment of comorbidities: standards of care in diabetes—2024” (2023)

[21] Deutsche Diabetes Gesellschaft (DDG), “S3-Leitlinie Therapie des Typ-1-Diabetes, Version 5” (2023)

[22] Bundesärztekammer *et al.*, “Nationale VersorgungsLeitlinie Typ-2-Diabetes” (2023)

[23] Wong *et al.*, “Guidelines on diabetic eye care: the international council of ophthalmology recommendations for screening, follow-up, referral, and treatment based on resource settings” (2018)

[24] Benoit *et al.*, “Eye care utilization among insured people with diabetes in the U.S., 2010–2014” (2019)

[25] Javitt *et al.*, “Preventive eye care in people with diabetes is cost-saving to the federal government: implications for health-care reform” (1994)

[26] Foot and MacEwen, “Surveillance of sight loss due to delay in ophthalmic treatment or review: frequency, cause and outcome” (2017)

One solution to comply with these requirements could be the use of *white-box* methods, such as linear regression and decision trees, that are intrinsically interpretable. Nonetheless, NNs often — in particular for high-dimensional input data such as images — yield improved predictive performance over these usually less flexible white-box models. Therefore, a frequently adapted approach is to use NNs and to compute post-hoc explanations such as *saliency maps* [11], [14] that try to highlight regions in the input image the NN puts large emphasis on. However, these are often criticized for being misleading, unreliable, and too abstract [27]–[29]. Another, more promising solution to comply with these requirements could be the use of *gray-box* models that incorporate task-dependent domain knowledge and concepts mimicking the specialists' diagnostic protocol into the black-box AI model. This could be used to trace the reasoning of the AI system at inference time. Exemplarily, intermediate segmentation maps of present lesions enable the DL model to provide a visual explanation for the resulting decision that shows *which regions* are affected by *which pathology* and to *what extent* [30], [31]. This would enable users who are aware of the clinical guidelines for diagnosing the disease but lack the expertise to identify the corresponding pathologies themselves to verify the prediction of otherwise opaque NNs. In the long run, this could ensure compliance with regulations and — more importantly — allow *reliable* and *seamless* use, foster the practitioners' *acceptance* and reasonable *trust* in the AI-based decision support system, and reduce the risk of erroneous predictions.

[27] Adebayo *et al.*, "Sanity checks for saliency maps" (2018)

[28] Arun *et al.*, "Assessing the trustworthiness of saliency maps for localizing abnormalities in medical imaging" (2021)

[29] Saporta *et al.*, "Benchmarking saliency methods for chest x-ray interpretation" (2022)

[30] De Fauw *et al.*, "Clinically applicable deep learning for diagnosis and referral in retinal disease" (2018)

[31] Hansen *et al.*, "Radiographic assessment of CVC malpositioning: how can AI best support clinicians?" (2021)

Moreover, instead of relying on standard NNs, exploiting methods of the Bayesian paradigm such as Bayesian neural networks (BNNs), which promise to provide increased robustness and well-calibrated uncertainty estimates [32, pp. 55–57], could help to mitigate the overconfidence and overfitting issues of NNs. Additionally, they could improve *transparency* w.r.t. communicating the prediction's certainty based on the evidence from the training data. However, a full Bayesian treatment of NNs is a highly nontrivial and computationally demanding task due to the high dimensionality of the parameter space and the strong nonlinearity [32, p. 57], [33], which is an essential part of NNs required to learn complex patterns.

[32] Bishop and Bishop, *Deep Learning: Foundations and Concepts* (2024)

[33] Jospin *et al.*, "Hands-on bayesian neural networks—a tutorial for deep learning users" (2022)

Another potential challenge may arise in the context of the DR screening example discussed above. That is, the implementation in low-income countries and primary care medical offices may be challenging, due to the high cost and limited portability of ophthalmoscopes. Moreover, domicil-

iary visits to conduct telemedical DR screening would be only possible to a limited extent in rural areas where internet connection could be slow or absent and, thus, cloud-based inference is not feasible. To ease the implementation, integrating the automatic AI-based decision pipeline into a mobile diagnostic device could be beneficial. Hence, it would be desirable to deploy the AI pipeline to run on the edge, e.g., on the practitioner's smartphone or directly on a mobile imaging device. This would require the DL pipeline to additionally be *lightweight* in order to meet the strict memory and computational restrictions while keeping inference time short to prevent the AI-based algorithm from being rendered unusable.

1.2 Scope, contribution, and structure

Based on these observations, i.e., the rising demand for mobile applicable, transparent, and explainable DL, this thesis focuses on reducing the gap observed regarding the translation of DL-based detection systems into clinical practice. This is addressed by the development of an automatic decision system that complies with all these requirements — applied for the above-introduced application of DR screening that is conducted by non-specialists. In detail, this comprises the setup of a single, end-to-end, DL-based, decision system for DR severity grading that simultaneously

(1) is explainable w.r.t. general, unspecialized practitioners and medical assistants allowing for sanity-checking and challenging individual predictions,

(2) transparently communicates a reliable, calibrated uncertainty estimate allowing for improved detection of potential failures, and

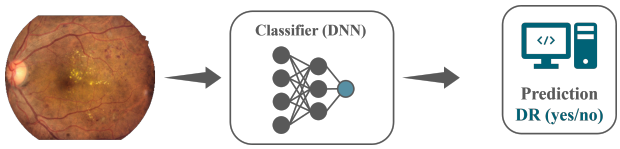
(3) is applicable to mobile devices to enable cost-efficient screening within regions with insufficient medical care.

The main concept of this thesis is depicted in Figure 1.1. To comply with requirement (1), i.e., tackling the lack of explainability of most standard DL methods as visualized in Figure 1.1a, the DL model in this thesis is set up to incorporate clinical expert knowledge by mimicking diagnostic guidelines for DR grading based on color retinal images in line with previous research on this topic [10], [30], [35] and concept-based learning methods [36], [37]. These are designed to enforce the model to learn pre-

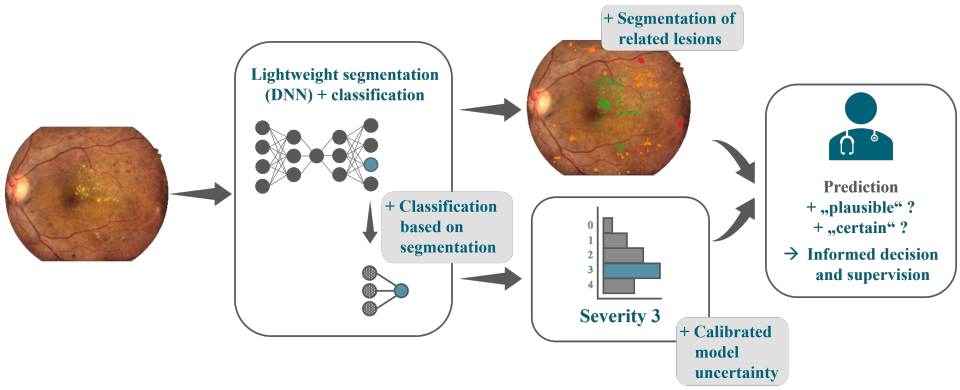
[35] Quellec *et al.*, "Explain: explanatory artificial intelligence for diabetic retinopathy diagnosis" (2021)

[36] Koh *et al.*, "Concept bottleneck models" (2020)

[37] Sarkar *et al.*, "A framework for learning ante-hoc explainable models via concepts" (2022)



(a) Common state-of-the-art DR detection pipelines.



(b) Transparent, interpretable, and lightweight DR severity grading.

FIGURE 1.1: Schematic depiction of the thesis' concept on narrowing the gap between black-box NNs and the deployment of transparent, interpretable DL models fostering acceptance and reliability for trustworthy usage in clinical practice. Retinal images are adapted from [34] with permission.

defined human concepts by introducing an intermediate bottleneck in the NN, which is trained in a supervised manner to align with these concepts. This inherently provides explanations for the model prediction. These diagnostic guidelines in detail comprise splitting the task for DR grading into two steps: First, the presence of DR-related pathological biomarkers within the patient's fundus is determined by means of fine-grained segmentations that can be used to both quantify the affected areas as well as to provide a detailed visualization of the present lesions. Subsequently, the stage of the disease is classified based on these segmentation masks and the international clinical diabetic retinopathy (ICDR) grading scheme [38].

[38] Wilkinson *et al.*, "Proposed international clinical diabetic retinopathy and diabetic macular edema disease severity scales" (2003)

To tackle the lack of calibrated uncertainty of common DL methods and meet requirement (2), state-of-the-art BNNs are deployed and evaluated in this thesis w.r.t. their impact on the overall quality of their uncertainty estimates. Moreover, this work examines whether these could provide a benefit to narrow the gap for clinical application of DL-methods that are transparently communicating their predictive certainty and could improve the detection of potential model failures as well as patient security.

To enable (3), the potential application of the algorithm to a mobile diagnostic device, i.e., running the DL-supported decision algorithm at the edge, lightweight NNs are implemented. In addition, specialized operations suitable for mobile application are analyzed for their benefit in reducing the model's computational complexity while maintaining as much predictive performance as possible. This is particularly relevant, as DL models for semantic segmentation are typically more complex and resource-intensive compared to standard CNN classifiers due to the often-utilized encoder-decoder architecture. That is, the extracted features are upsampled to the original input resolution to produce high-resolution, and high-precision segmentations.

Following the introduction provided in this Chapter 1, the remainder of this thesis is structured as follows:

Chapter 2 provides an overview of the fundamentals of DL, its advantages and disadvantages, such as the black-box characteristic and the lack of guarantees for model generalization followed by a more in-depth discussion of explainable DL methods.

Chapter 3 gives a brief introduction to diabetes mellitus (DM) and a more in-depth overview of diabetic retinopathy (DR) as one of its complications.

Chapter 4 analyzes the suitability of the U^2-Net [39] to be adapted as a lightweight, computationally efficient model architecture. It is a derivative of the famous U-Net [40] proposed for the task of biomedical image segmentation that additionally exploits the fusion of global and local features. For this analysis, the use of feature scaling and depthwise separable convolutions (DCs) applied to the U^2-Net is investigated to derive a reasonable trade-off for both the model complexity and predictive performance for DR related lesion segmentation. This chapter is mainly based on the analysis conducted in [41], first authored by the author of this thesis.

Chapter 5 evaluates the deep kernel learning (DKL) [42] method and two extensions to this approach [43], [44] for their benefit to achieve high-quality uncertainty estimates. By combining Gaussian processes (GPs) — a fully Bayesian, nonparametric approach —

[39] Qin *et al.*, "U^2-Net: going deeper with nested U-structure for salient object detection" (2020)

[40] Ronneberger *et al.*, "U-Net: convolutional networks for biomedical image segmentation" (2015)

[41] Siebert and Rostalski, "Performance evaluation of lightweight convolutional neural networks on retinal lesion segmentation" (2022)

[42] Wilson *et al.*, "Deep kernel learning" (2016)

[43] Liu *et al.*, "Simple and principled uncertainty estimation with deterministic deep learning via distance awareness" (2020)

[44] Tran *et al.*, "Calibrating deep convolutional gaussian processes" (2019)

with NNs into an end-to-end model, this method promises to benefit from the advantages of both approaches. That is, to simultaneously achieve high diagnostic performance and to derive awareness for unknown or difficult operating conditions aiming to enable users to detect potential failures of the DL model based on predictions with low certainty. This chapter was previously published in [45] by the author of this thesis.

[45] Siebert *et al.*, "Uncertainty analysis of deep kernel learning methods on diabetic retinopathy grading" (2023)

Chapter 6 analyzes the concept bottleneck model (CBM) [36] and a newly proposed soft-CBM (sCBM), i.e., two transparency-enhanced, inherently explainable DL architectures. These are derived by joining the approaches analyzed in Chapters 4 and 5 exploiting a concept-based learning approach. That is, a shallow DKL-based classifier is added to the lightweight lesion segmentation model and evaluated with regard to the DR grading task. Even though these models would ideally be trained in an end-to-end manner, this preliminary analysis evaluates the joint model performance solely based on the pretrained segmentation model in order to show the overall feasibility and potential of the proposed approaches.

Chapter 7 summarizes the results and main findings of this thesis and discusses further implications of the desired transparency enhanced and explainable DR grading system as well as exciting future research directions.

2 | Fundamentals and challenges of deep learning

In this chapter, first, a short overview of ML and DL, their general field of application, and commonly used learning paradigms will be provided in section 2.1. This is followed by a description of the fundamentals and basic concepts of ML in section 2.2, which build the foundation for the domain of DL. As a central methodology of this thesis, a more in-depth introduction to the latter, particularly, for DNNs and CNNs will be given in section 2.3 and section 2.3.4. Afterwards, sections 2.4 and 2.5 will discuss the properties and challenges that arise for using NNs regarding model generalization, transparency, and overfitting. Finally, in section 2.6, the concept of Bayesian DL will be introduced, which provides a fundamental and promising approach to improve both model generalization and uncertainty quantification.

2.1 Definition and learning paradigms

machine learning

While AI covers a vastly larger research area and in general a concise definition for *intelligence* is a philosophical question, the terminology is often used interchangeably with ML, i.e., it is referred to algorithmic solutions that are learned from data in contrast to traditional hard-coded, handcrafted algorithms [32, p. 1]. Thereby, standard methods from the ML domain, such as linear and logistic regression, decision trees, random forest (RFs), or support vector machines (SVMs) incorporate manually prepared features based on expert knowledge to which the learning algorithms are applied. In contrast, methods from its DL subdomain *solely* rely on automatic feature extraction, i.e., *task relevant* features are directly learned by the DNN model given a dataset, task setting, and error function. These build — together with the designed model — a complex hypothesis space of possible solutions. With this, properly designing the hypothesis space is

[32] Bishop and Bishop, *Deep Learning: Foundations and Concepts* (2024)

deep learning

a crucial core component of DL, i.e., model or task misspecification might cause the true solution to not be covered by the hypothesis space. When properly trained, DNNs were shown to achieve tremendous performance gains over traditional ML methods, particularly, when dealing with complex, high-dimensional tasks. However, even if the true solution might be covered within the solution space, searching for the optimal solution within this hypothesis space — which is typically solved using optimization algorithms along with gradient descent approaches due to the nonconvexity of the task — is not trivial. A more in-depth discussion about model generalization and that the feature *relevance* for solving a task does not necessarily align with the *true, underlying concepts* of the task that is to be solved will be provided in section 2.4.

supervised learning

Multiple learning paradigms can be distinguished based on the given task and error function, whereby — most frequently — supervised and unsupervised learning as well as mixtures of these are applied. With supervised learning, the goal is to derive a mapping function $f : \mathcal{X} \to \mathcal{Y}$ given a set of target $\mathbf{y} \in \mathcal{Y}$ and their respective input variables $\mathbf{x} \in \mathcal{X}$. This is in general accomplished by minimizing an error function that penalizes the difference between the predicted and true target values of the dataset. Supervised learning is commonly the method of choice for tasks such as regression and classification problems, i.e., predicting continuous or categorical outcomes given a (sufficiently large) number of labeled data.

unsupervised learning

In contrast, unsupervised learning is usually applied whenever target labels are unavailable and aims to directly learn the distribution $p(\mathcal{X})$ that the samples are drawn from [46, p. 14]. A typical unsupervised method is the auto-encoder, which is optimized to yield a function $f : \mathcal{X} \to \mathcal{Z} \to \mathcal{X}$ in order to learn a data low-dimensional encoding $\mathcal{Z}$ from $\mathcal{X}$ [32, p. 188]. This low-dimensional data representation could, e.g., be used to create new samples from the input domain by sampling from $\mathcal{Z}$. Mixtures of these, i.e., self-, semi-, or weak-supervision, are often used to extract meaningful information and biases from unlabeled data to improve the performance of supervised learning models.

[46] Murphy, *Probabilistic Machine Learning: An introduction* (2022)

The analyses conducted in this thesis can be subsumed under the supervised training regime. Thus, the remainder of this chapter will focus on introducing supervised ML and DL methods in more detail.

2.2 Linear models

As stated above, supervised learning aims to find a good model that fits the true, underlying function $f(\cdot)$ mapping from the source $\mathcal{X}$ to the target domain $\mathcal{Y}$ in order to predict the output for *new, unseen* samples from the source domain. With collecting observations for a number of N samples from this source domain, a dataset $\mathcal{D} = \{\mathbf{X}, \mathbf{Y}\}$ can be generated with inputs $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]$ and their respective targets $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N]$ with each $\mathbf{x}_n \in \mathcal{X}$ and $\mathbf{y}_n \in \mathcal{Y}$ for all $n \in N$. However, searching for a sufficient approximation to the true underlying function $f(\cdot)$ is a highly non-trivial task. Furthermore, achieving sufficient performance requires — depending on the task complexity — the use of highly flexible models to capture $f(\cdot)$. In a real-world setting, this is further complicated by the fact that the collected data will be subject to noise due to imperfect measurements and natural fluctuations of the observed processes. Moreover, in case the actual observations cannot be automatically measured — which is most frequently the case in the medical domain — physicians or domain experts have to manually label the data. Hence, also the target labels might be affected by noise due to the subjectivity of the labeling process.

observation noise

label noise

2.2.1 Linear regression

First, simple linear models will be introduced. They build a basic concept of ML and allow to, later on, introduce the concept of NNs, which are by far more flexible models and more suitable to approximate highly complex functions. To this end, the following simplistic regression task is introduced for which the input and target domain are defined as $\mathbf{x} = [x_1, x_2, \ldots, x_D]^\top \in \mathbb{R}^D$ and $y \in \mathbb{R}^1$. In *linear regression*, the true underlying function of the observed data $f(\cdot)$ is assumed to be a linear transformation of the input to the target, i.e.,

$$f(\mathbf{x}, \mathbf{w}) = w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_D x_D = \sum_{d=1}^{D} w_d x_d + w_0 \qquad (2.1)$$

with a vector of unknown weights $\mathbf{w} = [w_1, w_2, \ldots, w_D]^\top$ and a so-called bias term w_0. Intuitively, the weights define how much each feature $\mathbf{x}_d$ contributes to the prediction $f(\mathbf{x}, \mathbf{w})$ which leads to the graph structure as in Figure 2.1. By condensing the bias term into the weight vector[1],

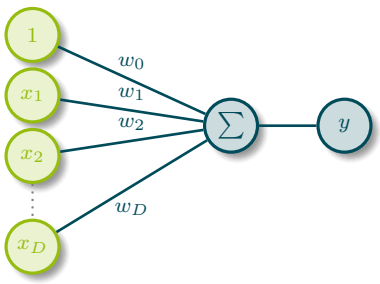
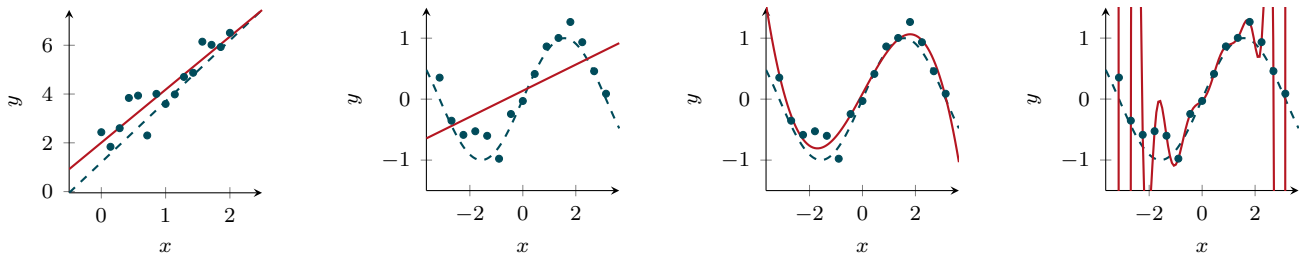


FIGURE 2.1: Simple schematic diagram of a linear regression model. Figure created based on [47].

[1]For convenience, this simple reformulation will be used throughout the remainder of this thesis without explicitly indicating $\mathbf{x}$ and $\mathbf{w}$ to be extended to $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{w}}$.



(a) $f(x) = 2.5x + 1.2$ with $\sigma^2 = 0.7$ (b) $f(x) = \sin(x)$ with $\sigma^2 = 0.2$ (c) $f(x) = \sin(x)$ with $\sigma^2 = 0.2$ and $M = 3$ (d) $f(x) = \sin(x)$ with $\sigma^2 = 0.2$ and $M = 25$

FIGURE 2.2: Linear regression example for a 1d problem with the ordinary least squares solution (red line —) to the samples (blue dots ●) from the true function $f(x)$ (dashed blue line --) that are corrupted by zero-mean, additive Gaussian noise. (a-b) Basic linear regression according to (2.1). Note, that this is equal to applying a transformation with polynomial basis functions where $M = 1$ to the input first. (c-d) Linear regression with applying polynomial transformations with $M > 1$ according to (2.5) and (2.6).

i.e., $\tilde{\mathbf{x}} = [1, x_1, \ldots, x_D]^\top$ and $\tilde{\mathbf{w}} = [w_0, w_1, \ldots, w_D]^\top$, the equation can be further simplified to

$$f(\tilde{\mathbf{x}}, \tilde{\mathbf{w}}) = \tilde{\mathbf{w}}^\top \tilde{\mathbf{x}}. \tag{2.2}$$

To derive a suitable set of weights $\mathbf{w}$, a measurement quantifying the degree to which $f(\mathbf{X}, \mathbf{w})$ matches the target observations $\mathbf{y}$ is required. For linear regression, typically the mean squared error (MSE)

$$\mathcal{L}_{\text{MSE}}(\mathbf{w}) = \frac{1}{N} \sum_{n=1}^{N} (f(\mathbf{x}_n, \mathbf{w}) - y_n)^2 \tag{2.3}$$

is used as error function that has an optimal, closed-form solution when minimized w.r.t. the parameters

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \mathcal{L}_{\text{MSE}}(\mathbf{w}) = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}. \tag{2.4}$$

ordinary least squares

This is called the ordinary least squares solution, which is reached when the error does no longer change, i.e., $\nabla \mathcal{L}_{\text{MSE}} = 0$ [46, p. 371]. A simple example of linear regression with $D = 1$ is depicted in Figure 2.2a.

However, as the name indicates and can be seen from Figure 2.2b, this method is limited to linear problems. However, to conduct non-linear regression, the input space can first be transformed by applying a set of nonlinear basis functions $\boldsymbol{\phi} = [\phi_0, \phi_1, \ldots, \phi_D]^\top$ according to

basis functions

$$f(\mathbf{x}, \mathbf{w}) = \mathbf{w}^\top \boldsymbol{\phi}(\mathbf{x}) \tag{2.5}$$

so that the problem again becomes linearly solvable. This type of linear regression can be represented by a graph as visible in Figure 2.3. Notably, this problem is still linear in the weights and, hence, has a closed-form solution for the global optimum w.r.t. the MSE [46, p. 11]. Exemplarily, for polynomial regression the vector of transformations equals

$$\boldsymbol{\phi} = \left[1, \mathbf{x}, \mathbf{x}^2, \ldots, \mathbf{x}^M\right]^\top. \tag{2.6}$$

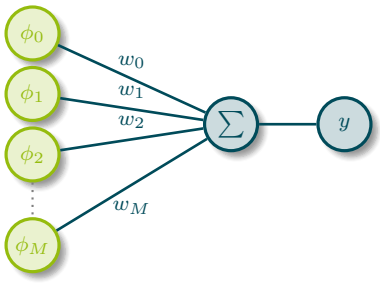


FIGURE 2.3: Simple schematic diagram of a linear regression model with a set of basis functions applied to transform the input. Figure created based on [47].

But also more sophisticated nonlinear functions such as the radial basis function (RBF) [32, p. 179]

$$\phi(\mathbf{x}, \boldsymbol{\mu}, s) = \exp\left(-\frac{\|\mathbf{x} - \boldsymbol{\mu}\|^2}{2s^2}\right) \tag{2.7}$$

are applicable.

However, this design choice for the correct basis function and the choice of the model complexity w.r.t. M are crucial and have to be well thought out, as visible from Figure 2.2. That is, picking an unsuitable basis function, or a model with insufficient flexibility, e.g., polynomial regression with $M = 1$ for fitting a sinusoidal function, can cause *underfitting*, whereas a model with too much flexibility, e.g., polynomial regression with $M = 25$, easily *overfits* and unwantedly starts capturing the noise within the training data by exactly fitting each sample. Both lead to poor generalization to samples different from those entailed in the training data. This can result in arbitrary erroneous predictions as in Figures 2.2b and 2.2d.

To select an appropriate number of basis functions or — more general — an appropriate model for the problem at hand, the model performance is typically measured on a holdout validation set of the training data [32, p. 14]. This process is called *model selection*. A more in-depth discussion of these phenomena and, particularly, how to mitigate model overfitting, which is a striking problem with using NNs, will be given in section 2.4. Moreover, note that — in contrast to, e.g., fixed polynomial basis expansion — the RBF is dependent on the location and scaling parameters $\boldsymbol{\mu}$ and s. This introduces $2 \times M$ additional, so-called *hyperparameters*, which have to be selected, i.e., more sophisticated models required to solve complex problems increase the difficulty of a proper model design.

model selection

2.2.2 Logistic regression

For classification, the task changes from regressing a continuous target $\mathbf{y}$ to learning a function that outputs binary class variables, e.g., $f : \mathcal{X} \to \mathcal{Y}$ with $\mathbf{y} = [0, 1]^C$ and C equals the number of classes [46, p. 7]. For the simplest case with $C = 2$, this can be achieved, e.g., by applying the logistic sigmoid function

$$\sigma(z) = \frac{1}{1 + e^{-z}}, \tag{2.8}$$

which squashes its input to the interval $[0, 1]$, to the output of the linear function in (2.1) according to

$$p(y = 1 \mid \mathbf{x}, \mathbf{w}) = f(\mathbf{x}, \mathbf{w}) = \sigma(\mathbf{w}^\top \mathbf{x}). \tag{2.9}$$

This can be interpreted to yield a probability $p(y = 1 \mid \mathbf{x}, \mathbf{w})$ for a sample to belong to class 1 given the sample $\mathbf{x}$ and the weights $\mathbf{w}$. Thereby, the binary target labels are defined as $y \in \{0, 1\}$ and $p(y = 0 \mid \mathbf{x}, \mathbf{w}) = 1 - p(y = 1 \mid \mathbf{x}, \mathbf{w})$. Based on decision theory, the optimal decision boundary between the two classes would then be given for $p(y = 0 \mid \mathbf{x}, \mathbf{w}) = p(y = 1 \mid \mathbf{x}, \mathbf{w}) = 0.5$ if the goal is to minimize the error rate of the classifier and the equal risks are associated with erroneous classifications of both classes [32, pp. 139–142].

In contrast to linear regression, the output of logistic regression is not linear w.r.t. the weights and, hence, there is no closed-form solution to compute the optimal weights [32, p. 160]. However, as the output of the logistic classifier yields the likelihood for the input to belong to class 1, i.e., one observes $y = 1$, given the input $\mathbf{x}$ and weights $\mathbf{w}$, it is straightforward to optimize the weights so that the observed likelihood on the dataset $\mathbf{X}$ is

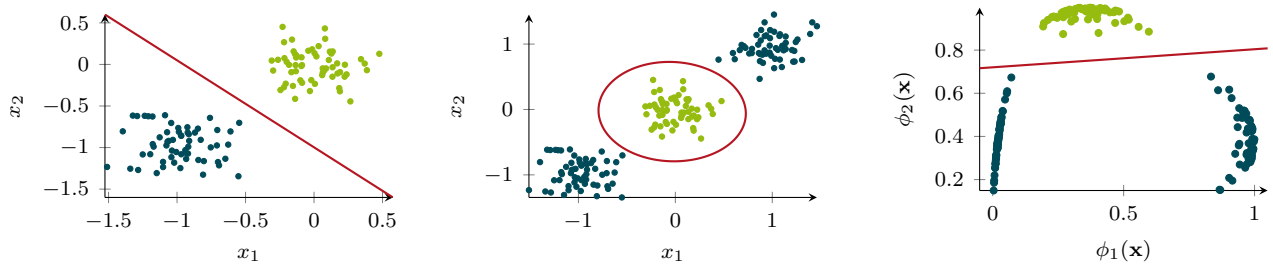
maximum likelihood estimation

maximized — which is called maximum likelihood estimation (MLE). That is, given the data $\mathcal{D} = \{\mathbf{X}, \mathbf{y}\}$, the total likelihood of the model can be computed by

$$p(\mathbf{y} \mid \mathbf{X}, \mathbf{w}) = \prod_{n=1}^{N} p(y = 1 \mid \mathbf{x}_n, \mathbf{w})^{y_n} \left(1 - p(y = 1 \mid \mathbf{x}_n, \mathbf{w})\right)^{(1-y_n)} \tag{2.10}$$

negative log-likelihood

from which the negative log-likelihood (NLL) error function



(a) Simple logistic regression.

(b) Original, linearly inseparable features, with the nonlinear decision boundary derived from (c).

(c) Transformation of the binary classification problem in (b) with two RBFs according to (2.7) with $\boldsymbol{\mu}_1 = (1, 1)$, $\boldsymbol{\mu}_2 = (0, 0)$, and $s_{1,2}^2 = 1$ used as basis functions $\boldsymbol{\phi} = [\phi_1, \phi_2]$.

FIGURE 2.4: Logistic regression example for a binary classification problem with $D = 2$ and samples of class 0 (blue ●) and of class 1 (green ●). The decision boundary (red line —) is displayed for $p(y = 1 \mid \mathbf{x}, \mathbf{w}) = 0.5$. Figure created based on [32, p. 159].

$$\begin{aligned}\mathcal{L}_{\text{NLL}} &= -\log\left(p(\mathbf{y} \mid \mathbf{X}, \mathbf{w})\right) \\ &= -\sum_{n=1}^{N} y_n \log p(y = 1 \mid \mathbf{x}_n, \mathbf{w}) + (1 - y_n) \log\left(1 - p(y = 1 \mid \mathbf{x}_n, \mathbf{w})\right)\end{aligned} \tag{2.11}$$

can be derived [32, p. 160], which is sometimes also called binary cross-entropy (BCE). Thereby, the gradient of the NLL w.r.t. the weights $\mathbf{w}$

$$\nabla \mathcal{L}_{\text{NLL}} = \sum_{n=1}^{N} (p(y = 1 \mid \mathbf{x}_n, \mathbf{w}) - y_n)\mathbf{x}_n \tag{2.12}$$

can be obtained analytically for applying the sigmoid activation [32, p. 160]. This can be exploited to optimize the weights $\mathbf{w}$ using gradient descent, i.e., minimizing the gradient $\nabla \mathcal{L}_{\text{NLL}}$. More general, logistic regression can equally be applied to a set of basis functions $\boldsymbol{\phi}$ that according to (2.5) apply a nonlinear transformation to the input variable $\mathbf{x}$, i.e.

gradient descent

$$p(y = 1 \mid \boldsymbol{\phi}, \mathbf{w}) = \sigma(\mathbf{w}^\top \boldsymbol{\phi}(\mathbf{x})). \tag{2.13}$$

Applying nonlinear transformations to capture complex patterns and dependencies in the data is the key to ML, which is well visualized in Figure 2.4. That is, the linear class boundary of a logistic regressor can only distinguish linearly separable clusters from another [32, p. 158], which is depicted in Figure 2.4a for a simple binary classification problem with $D = 2$. In contrast, for the two classes distributed as given in Figure 2.4b

no straight line exists that correctly separates the classes. However, using the RBF as in (2.7) allows the transformation of the linearly inseparable input space to a linearly separable representation of both classes, as depicted in Figure 2.4c. As any nonlinear transformation could be used because the model remains linear w.r.t. the weights, linear and logistic regression could in general be used to solve any kind of classification and regression problem [32, p. 172]. However, as already stated above, the selection of a sufficient set of basis functions is commonly only applicable to simple tasks due to the increasing complexity of the design choices that have to be made for more complex tasks [32, p. 113]. Even when reverting from fixed to data-dependent basis functions as in the example above, where the parameters $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ were chosen equal to the cluster centers, reasonably choosing the hyperparameters becomes prohibitively difficult for higher dimensional and more complex problems, and building a single basis function per data point is computationally limited to a low number of data samples [32, p. 179].

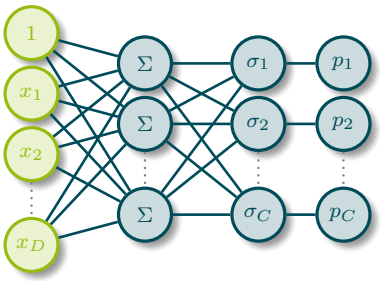


FIGURE 2.5: Simple schematic diagram of a multiclass logistic regression model. Thereby, p_c indicates the likelihood $p(y_c = 1 \mid \mathbf{x}, \mathbf{W})$ for class $c \in \{1, 2, \ldots, C\}$. Figure created based on [47].

For having more than two classes ($C > 2$), usually the generalized logistic softmax function

$$\sigma_c(\mathbf{z}) = \frac{e^{\mathbf{z}_c}}{\sum_{c'=1}^{C} e^{\mathbf{z}'_c}} \tag{2.14}$$

is used instead of the sigmoid function. Thereby, the latent activations $\mathbf{z} = [z_1, z_2, \ldots, z_C]^\top$ also called *logits* are computed by multiple linear models according to

$$\mathbf{z} = \mathbf{W}^\top \mathbf{x} \tag{2.15}$$

with $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_C]$ that can be visualized by the graph in Figure 2.5. Applying the negative logarithm to the model's likelihood for $C > 2$ leads to the cross-entropy (CE)

cross-entropy

$$\mathcal{L}_{CE} = -\sum_{n=1}^{N} \sum_{c=1}^{C} \mathbf{y}_{nc} \log p(y_c = 1 \mid \mathbf{X}, \mathbf{W}) \tag{2.16}$$

that can be derived using one-hot encoded target labels $\mathbf{y}_{nc} = [y_0, y_1, \ldots, y_C] = [0, \ldots, 1, \ldots, 0] \in \mathbb{R}^C$ comprising the binary target labels $y_c \in \{0, 1\}$ for each class $c \in C$. As above, optimal weights according to MLE can be obtained by minimizing $\nabla \mathcal{L}_{\text{CE}}$ [32, p. 162].

2.3 Artificial neural networks

Driven by the drawbacks linked to the complexity of manually searching for appropriate basis functions based on expert knowledge, the research on automatic and entirely data-driven approaches, particularly artificial NNs, has emerged founding the field of DL, which quickly superseded the performance of methods that rely on manual feature extraction. Still, the basic structure of these NNs, which will be introduced in the following, is very similar to the linear models covered in the previous section. However, NNs can simply *learn* to extract the features that they require to solve the task at hand, i.e., they efficiently learn a highly nonlinear, data-dependent, hierarchical set of basis functions that maps the input to a latent representation to which then, e.g., a linear classifier could be applied .

Historically, the idea behind these artificial NNs is adopted from the functioning of mammal and human brains [32, p. 16]. A human neuron as depicted in Figure 2.6 transfers an electrical impulse in case it is activated via its axon to the synapses that connect to other neurons, to which the electrical impulse is transferred via a chemical reaction. Each neuron itself is connected to many different neurons by their dendrites. Thereby, the sensitivity of the dendrites varies and the strength of the connection to other neurons determines whether an incoming activation stimulates or inhibits the neuron. The aggregation of all received electrical potentials from its dendrites determines the magnitude of the forwarded electrical potential of a neuron that is decoded via the frequency the neuron fires.

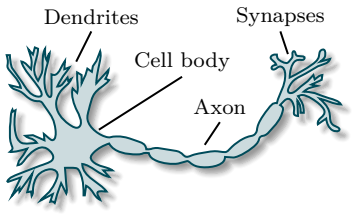


FIGURE 2.6: Simple schematic diagram of a human neuron. Figure derived based on [48].

This biological process was adopted to derive a *simplistic, artificial neuron* model: A single neuron receives the activations $\mathbf{x} = [1, x_1, x_2, \ldots, x_D]$ from its preceding neurons, which are aggregated by weighted sum with weights $\mathbf{w} = [w_0, w_1, w_2, \ldots, w_D]$. Finally, the neuron forwards the *activation potential* h to the subsequent neurons depending on a nonlinear activation function $\phi(\cdot)$ according to

artificial neuron

$$\begin{aligned} \mathbf{a} &= a(\mathbf{x}, \mathbf{w}) = \mathbf{w}^\top \mathbf{x} \\ \mathbf{h} &= h(a(\mathbf{x}, \mathbf{w})) = \phi(\mathbf{a}) = \phi(\mathbf{w}^\top \mathbf{x}). \end{aligned} \tag{2.17}$$

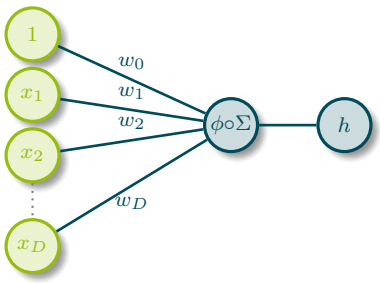


FIGURE 2.7: Simple schematic diagram of an artificial neuron. Figure created based on [47].

The vector $\mathbf{a}$ is referred to as the pre-activation of that neuron. A graph of this neuron model is shown in Figure 2.7 with $\phi \circ \sum$ denoting the sequential application of the nonlinear activation to the weighted sum of the inputs. Similar to the learning process by mammals, which is driven by adapting

the strength of the synapses between the neurons depending on personal experience, the neuron can be trained by adapting the weights $\mathbf{w}$. Note, that this is almost equivalent to the linear regression model as in (2.5), but instead of applying a set of nonlinear transformations to the input the output of the model is transformed by a single nonlinear activation function.

2.3.1 Fully-connected neural networks

A single neuron that is used with the Heaviside step function $\phi = H(\cdot)$ as nonlinear activation is called *perceptron*, which was proposed by F. Rosenblatt [49] in the mid-20th century. Later, using differential calculus and relying on continuous, differentiable activation functions such as $\tanh(\cdot)$ or the logistic sigmoid function as in (2.8) allowed building *multilayer perceptrons* (MLPs) [32, pp. 18–19]. These include multiple hidden layers with many neurons per layer and are defined as

[49] Rosenblatt, *Principles of Neurodynamics: Perceptrons and the Theory of BrainMechanisms.* (1962)

multilayer perceptron

$$\mathbf{y} = f\left(\mathbf{W}^{(1)}, \ldots, \mathbf{W}^{(L+1)}, \mathbf{x}\right) = h^{(L+1)}_{\mathbf{W}^{(L+1)}} \circ h^{(L)}_{\mathbf{W}^{(L)}} \circ \cdots \circ h^{(1)}_{\mathbf{W}^{(1)}} \tag{2.18}$$

$$= h^{(L+1)}\Big(\mathbf{W}^{(L+1)\top} h^{(L)}\Big(\ldots\Big(h^{(1)}\Big(\mathbf{W}^{(1)\top}\mathbf{x}\Big)\Big)\Big)\Big), \tag{2.19}$$

where $\mathbf{W}^{(l)}$ denotes the weight matrix of hidden layer $h^{(l)}(\cdot)$, $l \in \{1, \ldots, L\}$ as in (2.17) and $\mathbf{W}^{(L+1)}$ that of the final output layer. With this, each hidden layer can be interpreted — analogously to the basis function used in linear or logistic regression — as a set of parameterized, nonlinear transformations whose parameters can be learned from the data. As a result, with higher network depth, i.e., a larger number of hidden layers, the model can be optimized to extract hierarchical, highly complex features from the data [3]. For instance, when applied for the classification of cats and dogs from images, early layers in the NN might detect edges and colors, and later layers compose these features first to simple objects such as whiskers, snouts, ears, or eyes and finally to global objects like cats or dogs. When comprising a higher number of hidden layers depicted in Figure 2.8, these models are commonly referred to as *fully connected*, or *feed forward* NNs as in such a network each neuron forwards its activation to *all* the neurons of the subsequent layer. For convenience, the total set of parameters of a NN will be subsumed to $\boldsymbol{\theta} = \{\mathbf{W}^{(1)}, \ldots, \mathbf{W}^{(L)}, \mathbf{W}^{(L+1)}\}$ for the remainder of this thesis.

[3] LeCun *et al.*, "Deep learning" (2015)

fully connected neural network

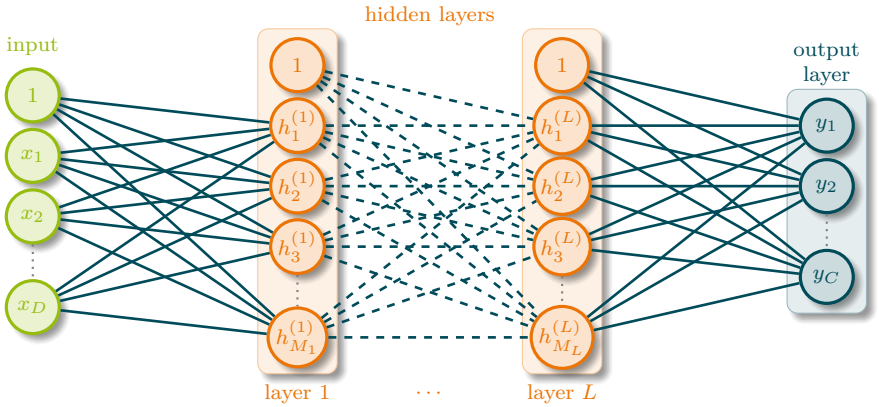


FIGURE 2.8: Schematic diagram of a fully connected neural network with L hidden layers and m_l nodes per layer for $l \in L$. Each node within the hidden and output layers corresponds to a single neuron as depicted in Figure 2.7. Figure created based on [47].

Two-layer NNs are known to be *universal approximators*. That is, a two-layer network can approximate any target function up to arbitrary accuracy, which, however, may require an infinite number of hidden neurons and may not be found by the optimization algorithm [32, p. 182]. Despite this, very deep NNs (DNNs) can more efficiently learn the same functions as a two-layer model due to the *compositional* inductive bias, which is linked to stacking many layers and learning hierarchical features [32, pp. 187–188]. However, despite their astonishing performance, DNNs have the disadvantage that they are black-boxes. Whereas simpler ML models such as linear and logistic regression, decision trees, k-nearest neighbors, or Naive Bayes [50, chap. 5] are intrinsically explainable as one can trace the individual features that lead to the given prediction, more complex and opaque models such as vanilla DNNs do not allow this due to their depth and the nonlinearity of each of the model's layers. These render the interpretation of the learned mapping function difficult or even impossible and prevent relating the learned latent features of the model to human-comprehensible concepts.

deep neural network

black-box

[50] Molnar, *Interpretable Machine Learning* (2022)

2.3.2 Optimization of neural networks

The same error functions as for linear and logistic regression — MSE, NLL, and CE — can be applied for regression and classification tasks when using NNs. In the DL domain, the error function is often referred to as *loss function*. However, due to the nonlinearity of the models, the error function

is no longer convex, and, hence, there is no guarantee of finding the optimal solution [32, p. 195]. Moreover, reaching the global optimum is unlikely due to the high-dimensional solution space.

To efficiently derive a solution to this nonconvex optimization problem, gradient descent approaches are commonly chosen. In detail, NNs can be trained by propagating the gradients of the applied error function $\nabla\mathcal{L}(\boldsymbol{\theta})$ w.r.t. the model weights $\boldsymbol{\theta}$ backward through the model due to the differentiable nonlinearities and by exploiting the chain rule of partial derivatives [32, pp. 233–238]. This process is called *backpropagation* and yields a gradient for each weight of the NN, which can be computed using toolboxes that provide automatic differentiation such as PyTorch or Tensorflow. Thereby, the gradient of the error $\nabla\mathcal{L}(\boldsymbol{\theta})$ is a vector in the direction of the steepest increase of the error. This can be exploited to iteratively update the model weights $\boldsymbol{\theta}$ by computing the gradients for the latest state of the weights and moving a small step in the weight space along the opposite direction of the gradient according to

backpropagation

$$\boldsymbol{\theta}_{(\tau)} = \boldsymbol{\theta}_{(\tau-1)} + \Delta\boldsymbol{\theta}_{(\tau-1)} \tag{2.20}$$

with

$$\Delta\boldsymbol{\theta}_{(\tau-1)} = -\eta\nabla\mathcal{L}(\boldsymbol{\theta}_{(\tau-1)}) \tag{2.21}$$

for iteration τ and *learning rate* η starting from an initial set of weights $\boldsymbol{\theta}_{(0)}$ [32, p. 214]. When computing the gradients $\nabla\mathcal{L}(\boldsymbol{\theta})$ for the whole batch of training data, this process is called *batch gradient descent*. However, in the limit of the large amount of data required to train DL models, processing the total set of training data at each iteration becomes infeasible. Therefore, typically *stochastic gradient descent (SGD)* is used that relates to computing a noisy estimate of the gradients in each iteration based on a single datum or — more commonly — a *minibatch* of the training data [32, pp. 215–216]. An important advantage of using minibatches instead of single dates is that the estimate of the gradient is less noisy due to averaging the gradients across the minibatch [32, pp. 215–216]. This noise can additionally be very helpful to escape local minima or from saddle points, where the gradient of the complete batch would equal zero [32, pp. 215–216]. Iterating the whole set of training data once is referred to as a single *epoch* of training.

stochastic gradient descent

minibatch training

The final model weights, performance, and speed of this optimization

highly depend, among others, on the initialization of the model weights as well as the chosen learning rate. Particularly, setting the learning rate too high — which equals following the gradient with a large step size — can cause the weight updates to oscillate or even lead to divergence [32, p. 218]. To prevent these oscillations, the weight update above (2.21) could be extended by a momentum term [32, pp. 220–222] according to

momentum

$$\Delta\boldsymbol{\theta}_{(\tau-1)} = -\eta\nabla\mathcal{L}(\boldsymbol{\theta}_{(\tau-1)}) + \mu\Delta\boldsymbol{\theta}_{(\tau-2)}. \tag{2.22}$$

Thereby, it is assumed that the update steps have a velocity and inertia, i.e. their direction cannot change abruptly. The parameter μ specifies the strength of this momentum [51]. Furthermore, the effective learning rate is increased if the direction of subsequent gradients is similar, which can additionally speed up convergence [32, p. 221].

[51] Sutskever *et al.*, "On the importance of initialization and momentum in deep learning" (2013)

The more sophisticated and at present most frequently used Adam algorithm [52] estimates exponential moving averages over both the gradients and the squared gradients separately for each weight according to

[52] Kingma and Ba, "Adam: a method for stochastic optimization" (2015)

$$\mathbf{g}_{(\tau)} = \nabla\mathcal{L}(\boldsymbol{\theta}_{(\tau-1)}) \tag{2.23}$$

$$\mathbf{m}_{(\tau)} = \beta_1\,\mathbf{m}_{(\tau-1)} + (1-\beta_1)\,\mathbf{g}_{(\tau)} \tag{2.24}$$

$$\mathbf{v}_{(\tau)} = \beta_2\,\mathbf{v}_{(\tau-1)} + (1-\beta_2)\,\mathbf{g}_{(\tau)} \odot \mathbf{g}_{(\tau)} \tag{2.25}$$

$$\hat{\mathbf{m}}_{(\tau)} = \mathbf{m}_{(\tau)} \Big/ (1-\beta_1^\tau) \tag{2.26}$$

$$\hat{\mathbf{v}}_{(\tau)} = \mathbf{v}_{(\tau)} \Big/ (1-\beta_2^\tau) \tag{2.27}$$

$$\boldsymbol{\theta}_{(\tau)} = \boldsymbol{\theta}_{(\tau-1)} - \eta\,\hat{\mathbf{m}}_{(\tau)} \Big/ \left(\sqrt{\hat{\mathbf{v}}_{(\tau)}} + \epsilon\right) \tag{2.28}$$

with $\odot$ denoting the element-wise product and ϵ a small constant to prevent division by zero. Thereby, equation (2.24) is similar to the momentum correction applied in (2.22) while, however, considering all previous gradients with their influence decaying over time instead of only accounting for the last optimization step. Furthermore, the exponential moving average of the squared gradients computed in equation (2.25) is used in the weight update step in (2.28) to scale the learning rate *per parameter* according to the magnitudes of the preceding steps. That is, parameters that receive larger gradients, which corresponds to moving along a direction with high curvature in the loss landscape[2], are updated with a smaller step-size compared to those with low gradients. Finally, to counter the bias

[2] The terminology loss landscape describes the highly nonconvex error surface that is derived from evaluating the loss function for each point within the high-dimensional parameter space of the model. The interested reader is referred to `https://losslandscape.com/` for some decent visualizations.

introduced by the initial values $\mathbf{m}_{(0)} = 0$ and $\mathbf{v}_{(0)} = 0$, equations (2.26) and (2.27) in the first optimization steps rescale the estimated moment and squared gradients. The adaptive moment parameters are typically chosen as $[\beta_1, \beta_2] = [0.9, 0.99]$ and have to fulfill $\beta_1, \beta_2 \in [0, 1)$. The Adam algorithm is very robust to the initial choice of the learning rate and can help speed up convergence in the highly complex loss landscape.

2.3.3 Vanishing and exploding gradients

Nonetheless, training particularly deep NNs initially remained difficult as these are likely to suffer from vanishing or exploding gradients [32, p. 227]. That is, due to the backpropagation of the gradients by use of the chain rule

$$\frac{\partial \mathcal{L}}{\partial w_i} = \sum_{m_1}^{M_1} \cdots \sum_{m_L}^{M_L} \sum_{c}^{C} \frac{\partial h_{m_1}^{(1)}}{\partial w_i} \cdots \frac{\partial f_c}{\partial h_{m_L}^{(L)}} \frac{\partial \mathcal{L}}{\partial f_c}, \tag{2.29}$$

where w_i is a weight of the m_1-th node in the very first layer $h^{(1)}$, the weights of the first layers in the model do not receive sufficiently large gradients if the factors on the right-hand side have values < 1. Equally, large latent activations can lead to gradients with values > 1 on the right-hand side of (2.29), which could drive the gradients towards infinity.

Nonlinear acitvation functions

Vanishing gradients were particularly problematic at the beginning of the rise of DL, due to using very smooth nonlinearities like the sigmoid activation introduced in (2.17) and depicted in Figure 2.9a or tanh function, which have exponentially decaying gradients for activations far away from zero [32, p. 184]. Despite even having zero gradients for all negative inputs $x < 0$ and being only piece-wise differentiable, the simple rectified linear unit (ReLU) nonlinearity

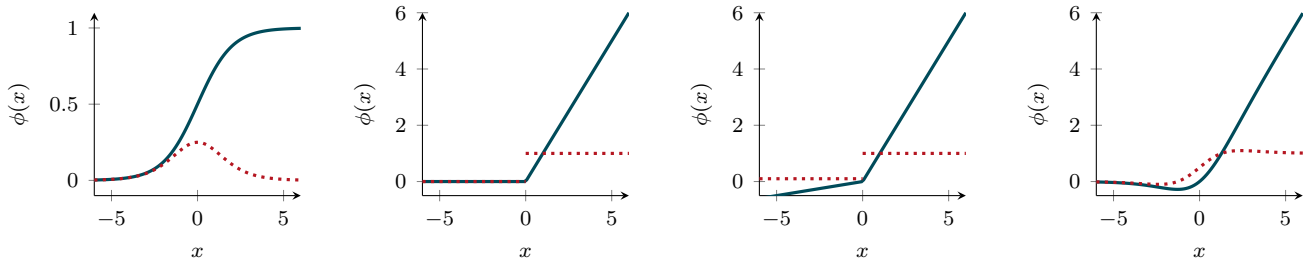
rectified linear unit

$$\mathrm{ReLU}(x) = \max(0, x) \tag{2.30}$$

displayed in Figure 2.9b, which has a constant nonzero gradient for all $x > 0$, was found to be very effective [5], [53]. Thereby, the non-differentiability at $x = 0$ can, in practice, be ignored [32, p. 185]. The emergence of the ReLU led to major improvements in the DL area and is still often used in state-of-the-art DNNs architectures.

[5] Krizhevsky *et al.*, "ImageNet classification with deep convolutional neural networks" (2012)

[53] Glorot *et al.*, "Deep sparse rectifier neural networks" (2011)



(a) Sigmoid (b) ReLU (c) Leaky ReLU (d) Swish ($\beta = 1$)

FIGURE 2.9: Frequently used nonlinear activation functions (—) and their respective gradients (···).

Nonetheless, a large body of alternative activation functions were proposed and, in general, any nonlinear function could be applied. However, simpler activation functions were observed to overall provide better performance [54]. Consequently, two of the frequently used alternatives are the *leaky* ReLU

[54] Ramachandran *et al.*, "Searching for activation functions" (2018)

leaky rectified linear unit

$$\text{lReLU}(x) = \max(0, x) + \alpha \min(0, x), \tag{2.31}$$

that has a small constant gradient for all negative inputs $x < 0$ as depicted in Figure 2.9c, and the Swish activation shown in Figure 2.9d that is defined as

$$\text{Swish}(x) = x\sigma(\beta x) \tag{2.32}$$

with $\sigma(\cdot)$ being the sigmoid activation. The swish function converges for $\beta \to \infty$ to the ReLU activation, seeks to avoid the disadvantage of the plain ReLU to being indifferentiable at $x = 0$, and was shown to provide performance benefits over the ReLU activation[54]. Commonly, it is used with $\beta = 1$ and in this form is referred to as a sigmoid linear unit (SiLU) that can be interpreted as a smooth version of the ReLU activation.

sigmoid linear unit

Data and layer normalization

In addition to using proper activation functions that improve the gradient flow during backpropagation, normalization of data and the intermediate activations of a NN were observed to play an important role in preventing vanishing and more importantly exploding gradients. Large model inputs and intermediate activations may lead to a highly complicated curvature of the loss landscape, resulting in exploding gradients that hinder or — most frequently — diverge the optimization process, as already observed from (2.29). To this end, the input data is typically standardized per feature,

Z-score normalization

i.e., scaled to have zero mean and unit variance, prior to feeding them to the neural network according to

$$\boldsymbol{\mu} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n \tag{2.33}$$

$$\boldsymbol{\sigma}^2 = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \boldsymbol{\mu}) \odot (\mathbf{x}_n - \boldsymbol{\mu}) \tag{2.34}$$

$$\tilde{\mathbf{x}}_n = \frac{\mathbf{x}_n - \boldsymbol{\mu}}{\boldsymbol{\sigma}} \tag{2.35}$$

using the data's mean $\boldsymbol{\mu}$ and standard deviation $\boldsymbol{\sigma}$ [32, p. 226]. The latter is often referred to as Z-score normalization. Furthermore, standardization is typically also applied to intermediate, latent activations of the neural network. A commonly adopted method is the *batch normalization (BN)* [55]. It standardizes the latent activations according to (2.33) to (2.35) where either the output $\mathbf{h}$ or the pre-activation $\mathbf{a}$ of a neuron as in (2.17) are used instead of the input $\mathbf{x}_n$ and the statistics are computed over the elements in the current minibatch of size B. To counter the reduction of the representational capacity of that neuron, which is induced by the standardization, Ioffe and Szegedy introduce an additional affine transformation. That is, the application of BN to the neuron pre-activations $\mathbf{a}$ during training equals

batch normalization

[55] Ioffe and Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift" (2015)

$$\boldsymbol{\mu}_B = \frac{1}{B}\sum_{b=1}^{B}\mathbf{a}_b \tag{2.36}$$

$$\boldsymbol{\sigma}_B^2 = \frac{1}{B}\sum_{b=1}^{B}(\mathbf{a}_b - \boldsymbol{\mu}_B) \odot (\mathbf{a}_b - \boldsymbol{\mu}_B) \tag{2.37}$$

$$\mathrm{BN}(\mathbf{a}_b) = \boldsymbol{\gamma}\left(\frac{\mathbf{a}_b - \boldsymbol{\mu}_B}{\boldsymbol{\sigma}_B}\right) + \boldsymbol{\beta} \tag{2.38}$$

with the trainable weights $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ that are optimized along with the other model weights [55]. As computing minibatch statistics is not possible at test time, the layer stores a running mean and a variance during training that are applied as a substitute for the minibatch statistics. Evidently, if $\boldsymbol{\gamma} = \boldsymbol{\sigma}_B$ and $\boldsymbol{\beta} = \boldsymbol{\mu}_B$ the operation performs an identity mapping. However, *learning the desired* statistics could help to stabilize the distribution of the latent activations within the network and allow using larger learning rates that otherwise could easily cause exploding gradients or the model to be stuck in local minima [55]. Following the motivation at the beginning of this section, subsequent analyses of BN more importantly found it to smooth the

loss landscape, which could allow faster, more stable training [56]. Despite the fact that the exact reasoning on how BN benefits convergence is not fully resolved, it was observed to stabilize and speed up convergence as well as to improve model performance and is a widely adopted technique in state-of-the-art deep learning models [55], [56], [32, p. 229].

[56] Santurkar *et al.*, "How does batch normalization help optimization?" (2018)

Residual connections

In addition to using normalization layers, so-called *residual*or *skip* connections [57] as depicted in Figure 2.10 were found to significantly stabilize optimization by smoothing the loss landscape [58]. They bypass a single layer as highlighted in Figure 2.8 or multiple layers of a NN, which often are referred to as a *residual block* and add the block's output $\mathrm{Bl}_1(\mathbf{h}_0)$ to its input $\mathbf{h}_0$, i.e.,

$$\mathbf{h}_1 = \mathrm{Bl}_1(\mathbf{h}_0) + \mathbf{h}_0. \tag{2.39}$$

[57] He *et al.*, "Deep residual learning for image recognition" (2016)

[58] Li *et al.*, "Visualizing the loss landscape of neural nets" (2018)

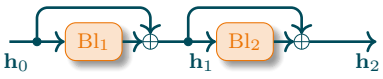
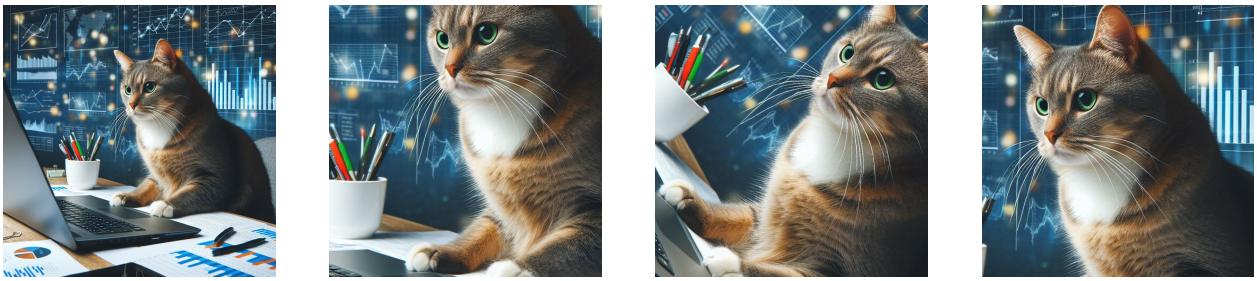


FIGURE 2.10: Graph of two residual blocks.

Thereby, the name stems from the interpretation, that the bypassed block Bl_1 learns the residual $\mathrm{Bl}_1(\mathbf{h}_0) = \mathbf{h}_0 - \mathbf{h}_1$ between the input and *desired* block output. Moreover, the bypassing of a block using residual connections can be interpreted as the addition of the output of two differently deep NNs, one with and one without the respective block. Hence, with concatenating multiple residual layers, running a single forward pass through the resulting NN is equivalent to aggregating the outputs of a set of parallelly executed subnetworks with differing depths. As this nonetheless entails predicting with the full-depth subnetwork, the NN can still exploit the full-depth model capacity. However, the subnetworks with less depth dominate the model output, and with this smooth the loss landscape of the optimization problem [32, p. 275] Moreover, the residual connections ensure that early layers receive sufficiently large gradients during backpropagation, which mitigates the vanishing gradient problem. Hence, using residual connections together with BN can effectively mitigate the occurrence of vanishing and exploding gradients and allows to train very deep NNs with hundreds of layers [57].

2.3.4 Convolutional neural networks

So far, the input data was assumed to have no intrinsic structure. That is, a random permutation of the input features $\mathbf{x} = [x_1, x_2, \ldots, x_D]$ prior to training would not affect model performance as there is no correlation



(a) original (b) scaled (c) scaled + rotated (d) scaled + translated

FIGURE 2.11: Examples of insignificant image transformations w.r.t. the class cat. Image created using Microsoft Designer (DALL·E 3).

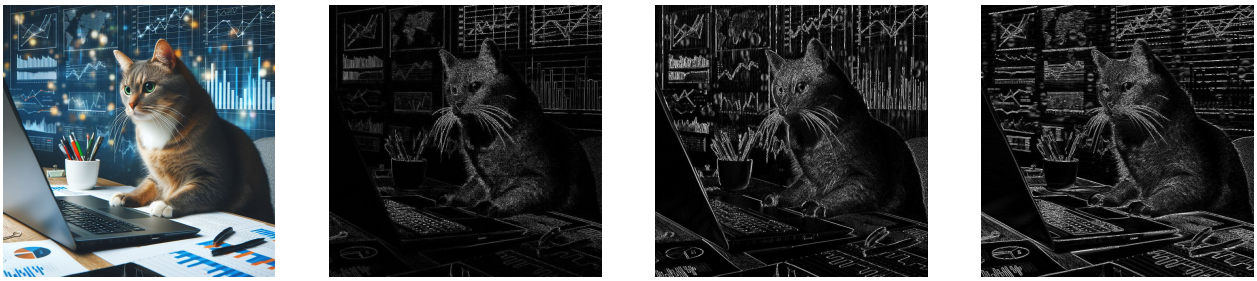
of the features w.r.t. their order. This is often the case for tabular data. However, for many applications, e.g., for natural language, time series, image, and video processing, the data follows an intrinsic structure by means of spatially and temporally neighboring features being highly correlated. For instace, images are typically represented as a 3d-tensor of pixels with height I_h, width I_w, and a number of image channels I_c depending on the image type. Hence, for an RGB-color image, the 3d-tensor is of size $\mathbf{X} \in \mathbb{R}^{I_c \times I_h \times I_w}$ with $I_c = 3$. Thereby, in contrast to randomly drawn pixel values, spatially nearby pixels in natural images are very likely to have similar intensities and colors, which together form homogenous image regions and ultimately form the objects visible in the respective image [32, p. 178].

invariance

Another property of such natural images is that semantically identical objects might appear in very different ways w.r.t. their raw pixel values due to different lighting conditions, image contrast, noise stemming from the image capturing system, or due to simple affine transformations such as image translation, rotation, or scaling of the objects within the image as shown in Figure 2.11. Thus, there exist many transformations to an image that do not change the semantic interpretation of that image. Hence, ideally, any image analysis method applied to natural images should be *invariant* to these insignificant changes of the input. Exemplarily for the task of image classification, the predicted class should remain identical disregarding the appearance of two images that are showing the same object but have different lighting and capturing angles.

equivariance and image segmentation

In contrast to image classification, the task of image segmentation requires the model to predict a class label per pixel in the input image. Transferring the concept of invariance to the latter, the segmentation model



(a) original (b) Laplace filtered (c) Sobel filtered (x) (d) Sobel filtered (y)

FIGURE 2.12: Examples for edge detectors run on the gray-scaled original image. Image created using Microsoft Designer (DALL·E 3).

will be required to be *equivariant* to these transformations, i.e., transformations to the input image should equally be reflected in the segmentation mask. With this, the order of first applying the transformation to the input image and then the segmentation model, or first applying the model and then transforming its output, should not result in any differences.

Despite a fully connected NN could, in theory, *learn* this invariance, it is difficult to gather enough data and come up with a loss function that sufficiently enforces the required invariance by penalizing noninvariant models [32, pp. 289–290]. Moreover, training fully connected NNs on images is highly inefficient and increasingly prohibitive with the image resolution. For instance, the input layer of a fully connected NN that is mapping an input image $\mathbf{X} \in \mathbb{R}^{3\times512\times512}$ to 1000 features would alone have approximately 786 M model weights.

Discrete convolution

Convolutional neural networks (CNNs) were developed to exploit the domain knowledge on image processing by extensively using *filter kernels* $\mathbf{K} \in \mathbb{R}^{k\times k}$. Using fixed, predefined filters is widely adopted in traditional image processing, which are applied, e.g., as edge detectors (Laplace or Sobel filter as depicted in Figure 2.12), lowpass filters for image smoothing and noise removal (mean-, median-, and Gaussian filter), or highpass filters to sharpen images (sharpening, unsharp masking). Although even natural numbers are a valid choice for the kernel size and are typically used within pooling operations as will be introduced later, k is often chosen to be an odd natural number $k = 2o + 1$ with $k, o \in \mathbb{N}$, which for simplicity will be assumed for the remainder of this section.

filter kernels

convolutional operation

These kernels are applied to the images using convolutions. By sliding the kernel over an input image $\mathbf{X} \in \mathbb{R}^{I_h \times I_w}$, each $(k \times k)$-patch $\tilde{\mathbf{X}}$ in the input image is multiplied by the filter kernel to produce a single pixel in the filtered output image $\hat{\mathbf{X}} \in \mathbb{R}^{\tilde{I_h} \times \tilde{I_w}}$. This equals sequentially computing a weighted sum of the *local* activations per input patch. Thereby, the size of these patches $\tilde{\mathbf{X}}$ is defined as the *receptive field* of the convolution, i.e., the total local area in the input that is *seen* by each step of the convolution. Formally, the $[a, b]$-th pixel in the filtered output image $\hat{\mathbf{X}}$ is computed by

$$\tilde{\mathbf{X}} = \mathbf{X}[(a-o):(a+o), (b-o):(b+o)] \tag{2.40}$$

$$\hat{\mathbf{X}}[a, b] = \sum_{k'} \sum_{k''} \tilde{\mathbf{X}}[k', k''] \, \mathbf{K}[k', k''] \tag{2.41}$$

where $k', k'' \in \{-o, \ldots, o\}$ and the colon operator $[a' : b']$ denotes the selection of the given range of matrix entries $[a', a' + 1, \ldots, b' - 1, b']$ from the input image. A full convolution of the image with the filter for all pixel pairs $[a, b], \forall a \in I_h, b \in I_w$ of the output image is typically denoted by

$$\hat{\mathbf{X}} = \mathbf{K} * \mathbf{X}. \tag{2.42}$$

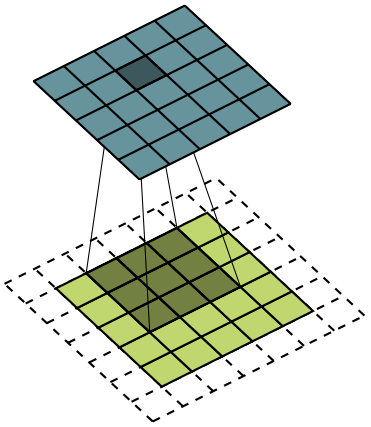


FIGURE 2.13: Visualization of a single step of a 2d convolution showing the kernel matrix (shaded area) being multiplied with a patch from the padded input image (green) to produce a filtered image (blue). Figure derived from [59] with permission.

Note that by restricting the convolution to positions where the kernel does not reach over the input image's border — which is termed *valid* convolution — the filtered image loses $2o$ pixels within each dimension, i.e., $\tilde{I}_{h/w} = I_{h/w} - 2o$. In case the filtered output image is required to have the same resolution as the input image, the latter can be extended by o pixels on each image border either by *padding* the image with a constant value, which is referred to as *same* convolution, or by mirroring or periodically repeating the input image. The process of a single convolution step is depicted in Figure 2.13 using a kernel with $k = 3$ and padding the input image with zeros to retain the input image resolution for the filtered image.

convolution as matrix multiplication

Generally, a convolution as in (2.42) can be reformulated to computing a single matrix multiplication, which equals a sparse, fully connected layer whereby the kernel's weights are tied to the spatial locations of the original kernel that slides over the input [46, p. 469]. Exemplarily, a 2d image

$\mathbf{X} \in \mathbb{R}^{3\times3}$ that is convolved with a (2×2) filter kernel without padding

$$\begin{bmatrix} \hat{x}_{11} & \hat{x}_{12} \\ \hat{x}_{21} & \hat{x}_{22} \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} * \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} \tag{2.43}$$

can be computed as

$$\hat{\mathbf{x}}_{\text{fl}} = \mathbf{W}_{\mathbf{K}} \mathbf{x}_{\text{fl}}^{\top} \tag{2.44}$$

by converting the kernel to the sparse Topelitz-like matrix $\mathbf{W}_{\mathbf{K}}$, i.e.

$$\mathbf{W}_{\mathbf{K}} = \left[\begin{array}{ccc|ccc|ccc} k_{11} & k_{12} & 0 & k_{21} & k_{22} & 0 & 0 & 0 & 0 \\ 0 & k_{11} & k_{12} & 0 & k_{21} & k_{22} & 0 & 0 & 0 \\ 0 & 0 & 0 & k_{11} & k_{12} & 0 & k_{21} & k_{22} & 0 \\ 0 & 0 & 0 & 0 & k_{11} & k_{12} & 0 & k_{21} & k_{22} \end{array}\right] \tag{2.45}$$

and given the flattened input $\mathbf{x}_{\text{fl}}$ and output image $\hat{\mathbf{x}}_{\text{fl}}$

$$\mathbf{x}_{\text{fl}} = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{21} & x_{22} & x_{23} & x_{31} & x_{32} & x_{33} \end{bmatrix} \tag{2.46}$$

$$\hat{\mathbf{x}}_{\text{fl}} = \begin{bmatrix} \hat{x}_{11} & \hat{x}_{12} & \hat{x}_{21} & \hat{x}_{22} \end{bmatrix}. \tag{2.47}$$

An alternative approach to computing the convolution is to transform the image into a representation that flattens the image patches centered around each pixel into a matrix (image-to-column operation). This matrix can then be multiplied with a dense representation of the kernel matrix.

multi-channel convolution

Moreover, convolutions can be generalized for images that have multiple channels $I_{c_in} > 1$ by extending the kernel by a third dimension that equals the number of channels in the input image. That is, given $\mathbf{X} \in \mathbb{R}^{I_{c_in}\times I_h \times I_w}$ the filter kernel can be set up as $\mathbf{K} \in \mathbb{R}^{I_{c_in}\times k\times k}$, which yields a filtered activation map of size $\hat{\mathbf{X}} \in \mathbb{R}^{1\times \tilde{I_h}\times \tilde{I_w}}$ as depicted in Figure 2.14. Accordingly, multiple image features can be extracted by applying a set of $I_{c_{out}}$ independent filters in parallel, which results in a mapping $\mathbb{R}^{I_{c_in}\times I_h\times I_w} \rightarrow \mathbb{R}^{I_{c_{out}}\times I_h\times I_w}$ —similar to using multiple neurons in parallel as in a fully connected NN while retaining the spatial dependencies of the structured input. Note that a convolution with a 1×1 kernel is equivalent to a fully connected layer on feature level and, thus, allows the application to spatially-structured data with variable resolution. As will be seen in the remainder of this work, this is commonly used to reduce the number of features in the output layer of CNNs while retaining spatial information.

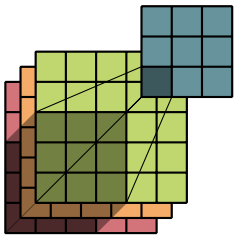


FIGURE 2.14: Visualization of a single step of a convolution showing the $(3 \times 3 \times 3)$ kernel (shaded area) being multiplied with a patch from an input image with 3-channels (red, orange, green) to produce a filtered image (blue). Figure derived from [59] with permission.

CNN architecture and properties

As for the training of fully connected NNs, the filter weights of the convolution filters can be learned from data instead of adopting predefined, hand-crafted filters. Hence, to build a CNN, multiple layers of convolutions as described above can be stacked after another to extract hierarchical features from the input. Analogously to the design of fully connected NNs, each convolutional filter can be accompanied by a bias term $\mathbf{w}_0$, and the filtered output images of a convolutional layer are fed through a nonlinear activation such as the ReLU. Reusing the alternative formulation of the discrete convolution as in (2.44), a node in a layer of a convolutional neural network (CNN) is given by

$$\hat{\mathbf{x}}_{\text{fl}} = \phi\left(\mathbf{W}_{\mathbf{K}}\mathbf{x}_{\text{fl}}^{\top} + \mathbf{w}_0\right) \tag{2.48}$$

whereby the bias term can be absorbed into the weight matrix $\mathbf{W}_{\mathbf{K}}$ analogously to (2.2). With this, the total amount of parameters in a convolutional layer adds up to $I_{c_{out}}\left(I_{c_{in}}k^2 + 1\right)$ and is independent of the image resolution (I_h, I_w). As a result, the input layer of a CNN that makes use of filters with $k = 3$ would only require about 28 k parameters for the exact same settings as for the fully connected network above, i.e., extracting 1000 features of any 3-channel image $(3 \times I_h \times I_w)$. Due to the equivalence to a structure exploiting, sparse fully connected NNs, the previously introduced optimization concepts for training NNs equally apply to training CNNs.

In contrast to the output of the fully connected model, the CNN retains the spatial information through the convolution that can be exploited by subsequent convolutional layers as well. Thereby, the effective receptive field of the deeper convolutional layers increases with the number of stacked convolutional layers, which allows the model to compose the locally extracted features to more global, spatially resolved, complex concepts and objects. To further enlarge the receptive field in order to improve the model's ability to detect global concepts, of course, larger kernels with $k > 3$ can be applied. However, this also increases the required number of model parameters quadratically. Moreover, simply stacking *more* convolutional layers instead of using a single convolution with a large filter kernel restricts the convolution being replaced to be composable to the subsequent application of multiple smaller convolutions [32, p. 301]. Another solution to enlarge the receptive field is to use dilated convolutions [60]. That is, the

receptive field

dilated convolutions

[60] Yu and Koltun, "Multi-scale context aggregation by dilated convolutions" (2016)

input to the convolution is sparsely *sampled* with a spread corresponding to the dilation factor d, as depicted in Figure 2.15 , whereby the number of weights in the convolution remains identical.

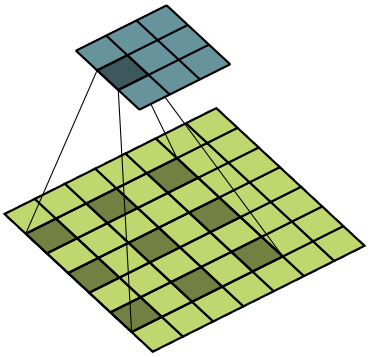


FIGURE 2.15: Visualization of a single step of a 2d dilated (d=2) convolution showing the kernel matrix (shaded area) being multiplied with a patch from the input image (green) to produce a filtered image (blue). Figure derived from [59] with permission.

Despite retaining spatial resolution is a key property of CNNs, the extracted global concepts required, e.g., for image classification, do often not require retaining all of the detailed spatial information, and suppressing small, irrelevant shifts is essential to derive the invariance property required for classification from the intrinsic equivariance of the convolution [32, p. 297]. Furthermore, retaining the full resolution throughout the model is computationally very expensive [32, p. 316], as the resulting model would require massive amounts of memory and processing time for using images with reasonable resolution. Hence, applying dimensionality reduction is an important, integral component for CNN's to perform well and run efficiently. To this end, the spatial resolution of the image is typically reduced gradually with the model depth by using *pooling* or *strided* convolutions. In contrast to regular convolutions, strided convolutions shift the filter kernel with a stepsize $s > 1$, which results in an effective reduction of the image resolution by approximately $\frac{1}{s}$. The concept of pooling is very similar to that of strided convolutions. Typically, the input image is divided into non-overlapping patches by using a kernel with identical size and stride, e.g., $s = k = 2$. However, the pooling layers do not entail learnable parameters. Instead, a simple, fixed function is applied to each *pool* of pixels in the input patch, such as the maximum or mean operator [32, p. 297]. In addition to the dimensionality reduction, applying pooling operations or strided convolutions helps to increase the effective receptive field of subsequent convolutional layers. Often multiple pooling operations are used alternately with convolutional layers that together build the so-called *feature extractor* or *convolutional backbone* of the CNN, which produces a set of expressive, low-resolution features. These are typically reduced by a final pooling operation to a 1d latent feature vector, i.e., all spatial information is removed, to which a small fully connected NN is added, which allows the classifier to take differently scaled input images as input.

strided convolutions

pooling

One of the first prominent and still frequently used model architecture is the VGG16 [61]. Later on, He *et al.* [57] proposed the famous ResNet model family depicted in Figure 2.16, which extensively exploits residual connections to allow deeper, more performant models and build the foundation for many state-of-the-art model architectures.

SOTA CNN architectures

[61] Simonyan and Zisserman, "Very deep convolutional networks for large-scale image recognition" (2015)

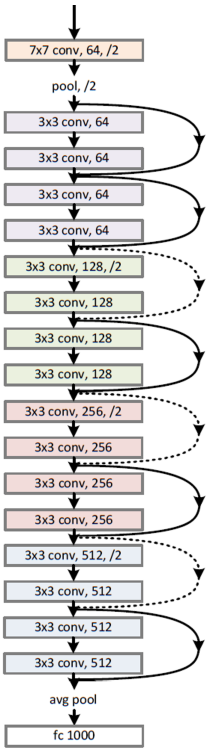


FIGURE 2.16: Schematic diagram of the ResNet-18 architecture whereby the number after the specifier for the type of the convolution relates to the output feature depth, "/2" denotes applying a convolution with stride $s = 2$, the dashed skip connections comprise a linear 1×1 convolution to map the equally increase the features depth of the residual connection, and each convolution comprises sequential application of BN and ReLU. Figure adapted from [57] with permission (© 2016 IEEE).

fully convolutional neural network

[62] Long *et al.*, "Fully convolutional networks for semantic segmentation" (2015)

[40] Ronneberger *et al.*, "U-Net: convolutional networks for biomedical image segmentation" (2015)

In contrast, both global as well as fine-grained local, high-resolution image features are required for the task of image segmentation to generate detailed masks that contain a single class label per pixel in the input image. A simple solution to this problem would be to build a CNN without any dimensionality reduction, i.e., using *same* padding, setting all strides $s = 1$, and applying no pooling. As this in principle would work, it would be computationally prohibitive as stated above. Therefore, a more efficient approach is to adapt the same model structure as for image classification to extract meaningful, global but low-resolution features. However, instead of applying a few final, fully connected layers, the latent features of the model are upsampled to the input image resolution. As upsampling with fixed functions, such as bilinear upsampling, may not result in sufficiently detailed segmentation masks due to the high-dimensionality reduction applied in the feature extractor, fully convolutional neural networks (FCNs) [62] learn the required upsampling operation by applying transposed strided convolutions, i.e., for each pixel in the input image, a patch with size and stride $s > 1$ in the output is generated.

To further improve the details of the final segmentation maps, intermediate high-resolution activations of the feature extractor can be extracted. With upsampling the latent features first to this intermediate resolution, the extracted features can be concatenated to the former prior to applying the final output convolution. This can be taken to the extreme, by extracting the activations of all intermediate resolution stages of the feature extractor. To incorporate these into the upsampling branch, the structure of the feature extractor can be mirrored exploiting either a combination of upscaling and convolution operations or transposed convolutions. Due to the similarity of such a model architecture to autoencoders, the features extractor and the nearly identical upsampling path are typically referred to as *encoder* and *decoder*. One very prominent model of that class is the U-Net [40] depicted in Figure 2.17, which is frequently used for biomedical image segmentation tasks.

With this, the most important properties of convolutions became visible: They introduce the desired bias that images are locally highly correlated and that patterns and image objects are locally invariant to the model architecture. That is, in contrast to using fully connected NNs, subsequent layers in a CNN are only sparsely connected reflecting that the semantic meaning of nearby pixels is likely to be similar, disregarding their spatial

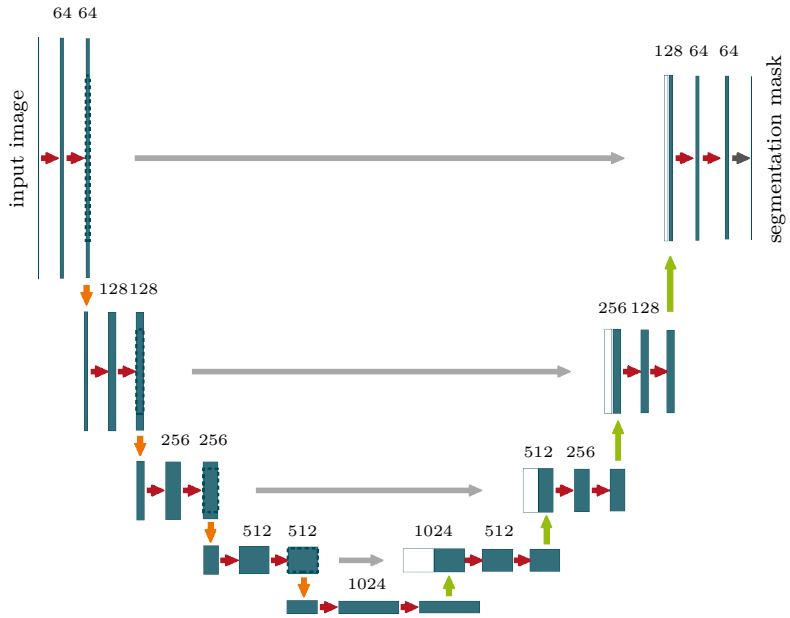


FIGURE 2.17: Schematic diagram of the U-Net architecture with 3 × 3-convolution+ReLU (➡), 1 × 1 conv (➡), max-pool (➡), up-conv (➡), and crop, copy, and concatenation (➡) operations. The numbers indicate the feature depth of each associated convolution block. Figure adapted from [40] with permission.

location. Assuming, for example, a filter detects vertical edges, then the filter detects these edges *disregarding* the spatial location in the input image and stores the information that an edge is present in the respective pixel of the output activation map as the same filter is moved over the whole input image, i.e., the convolution with a filter is inherently equivariant to shifts of objects in the input image. Thereby, as the filter weights remain identical, the weights required to compute the values of all pixels in the output activation are shared, which strikingly reduces the number of model parameters compared to fully connected neural networks. Therefore, training a CNN typically requires less training data. Furthermore, the number of weights in the filter is independent of the input image size and is applicable to images of any resolution without requiring a redesign of the model architecture. Additionally, due to preserving the spatial information of the image, stacking multiple convolutional layers can extract complex, hierarchical, and spatially resolved features.

2.4 Generalization and robustness

The severe overfitting that was observed when applying the overparameterized linear model to the simple regression task in Figure 2.2d highlights the challenges of selecting a model with sufficient capacity for the task at hand to capture the underlying concepts of the finite set of training data

without overfitting the noise in order for the model to generalize on unseen data. That is, the model should ideally be independent of the finite set of data samples. Hence, training the model on different datasets collected from the same domain and for the same task should yield the same optimal model. In principle, the optimal model could be derived in the limit of infinitely many data [32, p. 124], which is infeasible in practice. Particularly in the medical domain, it is typically very difficult to collect large, high-quality sets of labeled data due to the associated costs, the time-consuming labeling process, and the limited availability of data due to data privacy regulations [11]. As a result, model selection involves a tradeoff between a sufficient capacity to capture essential variations in the finite set of training data and achieving a good data fit (low bias) while limiting model flexibility to prevent overfitting (low variance). However, model flexibility typically increases variance while decreasing bias, and, therefore, the goal of model selection is to find the optimal level of flexibility that minimizes both bias and variance [32, pp. 125–126]. This is known as *bias-variance* tradeoff.

[11] Petersen *et al.*, "Responsible and regulatory conform machine learning for medicine: a survey of challenges and solutions" (2022)

bias-variance tradeoff

2.4.1 Model flexibility and regularization

In practice, estimating the optimal tradeoff between model bias and variance, i.e., the degree of model flexibility in terms of the optimal number of depth and width of a NN, is non-trivial. Therefore, the NNs are often rather overparameterized, i.e., they are designed to comprise more parameters than necessary to fit the training data, and the model complexity is controlled by applying *regularization* techniques.

regularization

Collecting as much as possible labeled data for model training effectively regularizes the model complexity and mitigates model overfitting [32, pp. 11–12]. Moreover, in case a large amount of unlabeled data is available, unsupervised learning strategies can be exploited by, e.g., pretraining the NN's model weights in order to mitigate model overfitting [3]. If the collection of more data is not feasible, it may be straightforward to reduce the model's flexibility by incorporating explicit or implicit bias into the model structure. As discussed in detail in the previous section, using CNNs is a very effective method to explicitly encode domain knowledge and introduce a bias to a model w.r.t. processing structured data such as images and time series.

data augmentation

To strengthen the equi- or invariance of CNNs to irrelevant transforma-

tions such as translation, rotation, or scaling, *data augmentation* can be applied. this additionally induces a bias to the optimization process and is a very effective, frequently applied method in image processing [32, p. 258]. Data augmentation artificially enriches the available training data by including plausibly transformed replications of the original samples, which could have been naturally captured without actually collecting new data. By keeping the target fixed, i.e., the class label in case of image classification, the model is encouraged to learn a mapping function that is invariant to the applied augmentation. Data augmentation can either be utilized to artificially increase the number of training samples prior to training or — in case of relying on stochastic optimization — be applied randomly to the current minibatch during the optimization process. For the task of image segmentation, where the model is required to be equivariant, identical transformations must also be applied to the target masks.

image preprocessing

Additionally, image preprocessing techniques can be used to eliminate variance from the data apriori, using domain knowledge. However, similar to the use of hand-crafted feature extraction, learning (approximately) invariant mapping functions usually outperforms manually designing highly specialized preprocessing pipelines to remove irrelevant variances [32, p. 257]. Therefore, in the field of image analysis, preprocessing is frequently only applied, e.g., for image resizing and contrast enhancement.

weight regularization

Overfitting was observed to relate to learning large weights for the individual basis functions, as these yield more rapidly varying outputs utilized to minimize the error function by capturing the data noise [32, p. 11]. Hence, in contrast to strictly limiting the model's capacity by using a fixed small, number of layers and nodes in the NN, the effective capacity of the learned mapping function can also be controlled by including a *regularization* term to the loss of function that can, e.g., induce sparse model parameters or penalize large weights. That is, adding a penalty for the magnitude of the weights to the loss function proposed in (2.3) according to

$$\tilde{\mathcal{L}}_{\mathrm{MSE}}(\mathbf{w}) = \mathcal{L}_{\mathrm{MSE}}(\mathbf{w}) + \frac{\lambda}{2}\|\mathbf{w}\|^2 \tag{2.49}$$

can help to mitigate overfitting to noise in the training data [32, p. 13]. Thereby λ is a hyperparameter that controls the effective model complexity. Particularly, for $\lambda \to \infty$ all weights are driven to zero. This parameter has to be chosen carefully since high values for λ, i.e., a high penalty for

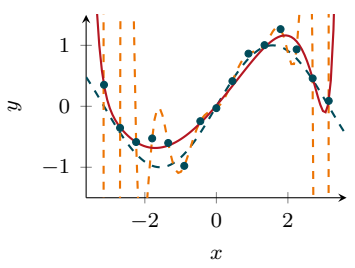


FIGURE 2.18: Regularized (red line —) and nonregularized ordinary least squares solution (dashed orange line - -) to the samples (blue dots ●) from the true function $f(x) = \sin(x)$ (dashed blue line - -).

larger weight magnitudes, possibly result in bad performance and underfitting similar to Figure 2.2b, whereas a small penalty does potentially not prevent overfitting. However, if well chosen, the additional regularization term can sufficiently constrain the effective model complexity, as visible from Figure 2.18, which shows the ordinary least squares solution of the regression problem in Figure 2.2d overlaid by the ordinary least squares solution using the identical number of polynomial basis expansions ($M = 25$) and additionally applying weight regularization as in (2.49) using the Euclidean-/L2-norm with $\lambda = 7.5 \times 10^{-3}$. This technique is equally applicable to training NNs, i.e., the gradient used for the parameter update in the stochastic gradient descent algorithm in (2.21) would be computed as

$$\nabla \tilde{\mathcal{L}}(\boldsymbol{\theta}) = \nabla \mathcal{L}(\boldsymbol{\theta}) + \lambda \boldsymbol{\theta} \tag{2.50}$$

for using the L2-norm $\|\boldsymbol{\theta}\|_2^2$ as regularization and in this context referred to as *weight decay* or *L2-norm regularization* [32, p. 260].

L2-norm regularization

Overfitting of NNs can also be targeted by tracking the model performance on a separate split of the training data and stopping the iterative optimization in case the error on this validation set no longer decreases or even starts to increase, which indicates the model to overfit the training data. This technique is called *early stopping* and — under the assumption that the effective model capacity grows with the number of optimization steps — can be understood as applying weight decay [32, p. 267].

early stopping

Another popular and very effective regularization method for NNs is called *dropout* [63], which is depicted in Figure 2.19. As the name suggests, the output activations of individual neurons are randomly dropped during training, i.e., set to zero, with a survival probability p. Similar to the intention behind residual connections, this equals randomly sampling subnetworks of differing capacities, which, however, share all remaining parameters for each forward pass. Intuitively, dropout strengthens the independence of the model weights as these cannot rely on the presence of the other neurons and, hence, reduces the coadaptation of neurons, which otherwise would encourage the model to overfit the training data [63]. At test time, all neurons will be kept in the model, which corresponds to averaging the predictions of all implicit subnetworks. To keep the expected input activation of each node at test time similar to that during training, the output of each neuron has to be scaled by the survival probability p [63].

dropout regularization

[63] Srivastava *et al.*, "Dropout: a simple way to prevent neural networks from overfitting" (2014)

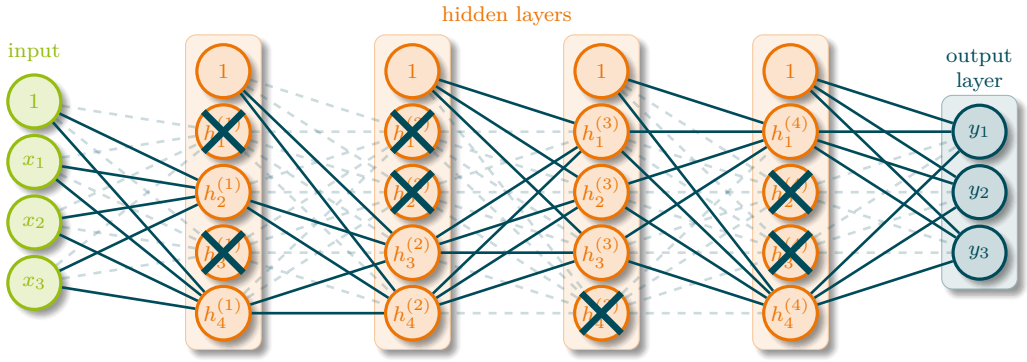


FIGURE 2.19: Schematic diagram of a single forward pass of fully connected neural network with dropout enabled. Each neuron is kept with probability p. Deactivated neurons for this forward pass might be active for another one. Figure created based on [47].

This can also be interpreted as implicitly learning an ensemble of models. Ensemble learning is a frequently applied method in the ML domain combining multiple, potentially weak base models to a strong and more powerful group of models. Whereas only a single model is effectively learned by using dropout, multiple independent CNNs can be trained to form an explicit ensemble of models by, e.g., averaging their outputs. The main reason why ensemble learning can improve over using a single, fixed set of parameters, is in line with the model selection problem: Aggregating the output of multiple solutions to the nonconvex optimization process typically improves on relying upon a single, locally optimal solution. Thereby, the individual ensemble members might overfit the training data, but the diversity of the individual members — stemming from the stochastic nature of the optimization processes, the random weight initialization, using random data splits (bootstrapping), or using different model architectures and hyperparameters [64] — mitigates the overfitting and can improve robustness [32, pp. 277–279]. As will be discussed in section 2.6, this idea is very similar to Bayesian inference and Bayesian model averaging.

ensemble learning

model averaging

[64] Wenzel *et al.*, "Hyperparameter ensembles for robustness and uncertainty quantification" (2020)

2.4.2 Robustness and spurious correlations

Having trained a well-parameterized model that achieves a good performance w.r.t. the holdout validation set of the training data does not guarantee the model to generalize to real-world data and applications [65]. One reason for this is a phenomenon called *shortcut learning*, i.e., the opti-

[65] Geirhos *et al.*, "Shortcut learning in deep neural networks" (2020)

shortcut learning

mization process yields a solution based on *spurious correlations* of, e.g., image features that coincidentally align with the target variable but are not causal for the target. That is, the NN might learn to focus on specific patterns contained in the training data that are *useful* to solve the task — such as unique noise patterns that are specific to a scanner type — but are not causal, i.e., they do not relate to the true underlying concepts of the problem — such as the presence of disease-specific pathologies. As a result, testing the model on in-distribution (ID) test data typically works well. However, NNs are often used to make predictions on images that were acquired differently, e.g., in a different hospital, and might be sampled from a different underlying distribution, e.g., due to geographic prevalence changes of the disease or due to being trained on data from secondary care with the intention to deploy the model in primary care. When exposed to such data shifts, the model predictions could be arbitrarily wrong.

[66] Zech *et al.*, "Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: a cross-sectional study" (2018)

A striking example of this phenomenon was uncovered by Zech *et al.* [66] who trained a CNN for the detection of pneumonia from a dataset of X-ray scans collected from two different institutions. They revealed the final model to increasingly rely on the institution and the scanner type at which the image was captured to predict the presence of pneumonia instead of focussing on pathological areas in the X-ray scan. Consequently, if remaining undetected and not explicitly taken care to find and remove such biases from the training data, spurious correlations can prevent the model from generalizing to real-world test data which may cause unreliable predictions. This can pose a major risk, particularly for the application of DL in the medical domain, to harm patients or users affected by such a system.

adversarial examples

[67] Szegedy *et al.*, "Intriguing properties of neural networks" (2014)

[68] Zhang *et al.*, "Adversarial robustness through the lens of causality" (2022)

[69] Su *et al.*, "One pixel attack for fooling deep neural networks" (2019)

[70] Finlayson *et al.*, "Adversarial attacks on medical machine learning" (2019)

[71] Goodfellow *et al.*, "Explaining and harnessing adversarial examples" (2015)

A similar effect is observed for *adversarial examples* [67], which can be interpreted by means of exploiting spurious correlations [68]. That is, small, intentional perturbations of the input of a NN can easily change the model prediction. These changes can be noise, which is imperceptible to the human eye [67], the change of a single pixel [69], or the application of image transformations such as rotations [70]. To counter this susceptibility, adversarial training, i.e., adding adversarial examples to the training data, proved effective for improving robustness against adversarial examples [71]. Although adversarial attacks might seem unlikely to occur in the medical domain given that images are often processed directly from the output of, e.g., an X-ray scanner, they highlight the fragility of NNs.

2.5 Transparency and explainability

As discussed above, NNs are susceptible to phenomena such as overfitting and shortcut learning. This is a serious problem because NNs are black-boxes, and it is typically impossible to trace the reasoning of such deep and complex models rendering detection of these failures difficult. Hence, rigorous testing on real-world data is essential to detect model overfitting and a potential poor generalization performance [11]. However, depending on the task complexity, it might be prohibitively difficult to design such rigorous test scenarios that comprise every possible failure case. Therefore, to prevent unexpected model failures both during development and after deployment, DL algorithms should be designed as *transparent* and *explainable* as possible [11]. This would not only help developers and deployers, but also users and affected individuals to understand and trace the model's reasoning, foster trust and acceptance, and comply with regulatory framworks such as the EU AI Act [11].

In the broader context of AI, *transparency* can be defined as

> “ *[...] the capability to describe, inspect and reproduce the mechanisms through which AI systems make decisions and learn to adapt to their environment, and the provenance and dynamics of the data that is used and created by the system.* ”
>
> — *V. Dignum [72, p. 54]*

This relates to making every *information* related to the development of the AI system, e.g., training data, model architecture, weights, and optimization design *available* to the relevant stakeholders, as long as it does not conflict with other legal concerns such as intellectual property and patient privacy [11]. Note, that communicating, e.g., the model architecture and weights would be deemed transparent w.r.t. this definition and also would allow inspection of the model. However, it does neither imply interpretability nor explainability[3] due to the black-box characteristic of DNNs. The more refined definition of *algorithmic transparency*, in contrast, requires

> “ *[...] the ability of the user to understand the process followed by the model to produce any given output from its input data.* ”
>
> — *A. Barredo Arrieta [13]*

[3] The key difference between interpretability and explainability is that a model can intrinsically be interpretable and explanations can be used to explain both non- and interpretable models [13].

That is, explicitly targeting the transparency of the entirety of the model design and optimization, algorithmic transparency opposes the black-box model design, which prevents tracing and, hence, understanding the mechanisms of the model's reasoning and links transparency with the demand for explainability.

Moreover, another important aspect comprised in this definition of transparency is to transparently communicate predictive uncertainty [11], [13], i.e., to inform the user when the system should not be relied on and, hence, reject the model prediction, in order to detect potential model failures. With this, explanations for the prediction should be complemented by an additional proxy for the trustworthiness of the prediction [11]. This could either be the case if the current sample can not be classified to one or another class with sufficient evidence or the training data did not include any, or only a few, similar examples and, hence, the model has to extrapolate to predict. Standard NNs, however, often are highly overconfident leading to unreliable uncertainty calibration [73]. The reasons for this pathologic overconfidence and the promising solution to rely on the Bayesian methods will be discussed in more detail in the subsequent section 2.6.

[73] Guo *et al.*, "On calibration of modern neural networks" (2017)

To address the challenges related to the black-box nature of NNs, there is a large field of research on explainable artificial intelligence (XAI), i.e.,

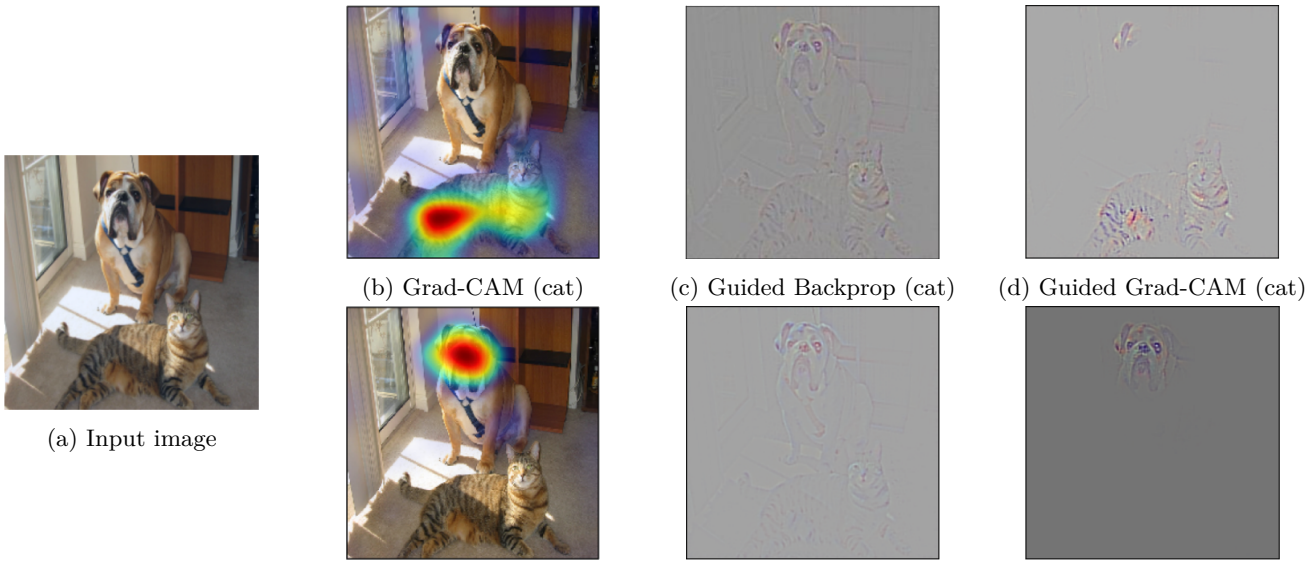
> *"[...] one that produces details or reasons to make its functioning clear or easy to understand."*
>
> — *A. Barredo Arrieta [13]*

Thereby, explanations can be derived on different levels, i.e., locally (for individual samples) or on a global level (for the model itself). Furthermore, explanations can be computed either post- or ante-hoc, i.e., after model training or by designing inherently interpretable models. As a result, XAI can obey very different forms, ranging from a simple reduction of the model complexity that eases the interpretability of the learned mapping function over visualizations or text generations to tools that allow the users to interact with the model [13].

2.5.1 Feature exploration

The influence of input features on a model or individual model predictions could, e.g., be explored using partial dependency (PD) [74] or individual

[74] Friedman, "Greedy function approximation: a gradient boosting machine." (2001)



(a) Input image (b) Grad-CAM (cat) (c) Guided Backprop (cat) (d) Guided Grad-CAM (cat)

(e) Grad-CAM (dog) (f) Guided Backprop (dog) (g) Guided Grad-CAM (dog)

FIGURE 2.20: Saliency maps computed for the class cat (b–d) and class dog (e–g) by using Grad-CAM [76] (b, e), Guided Backprop [77] (c, f), and Guided Grad-CAM (d, g), which is a combination of both the former. Figure reproduced from [76] with permission.

conditional expectation (ICE) [75] plots. However, these are usually limited to a few features, do not comprehensively capture feature interactions [50, sec. 8.1.4 and 9.1.3], and are therefore ill-suited for interpreting CNNs.

[75] Goldstein *et al.*, "Peeking inside the black box: visualizing statistical learning with plots of individual conditional expectation" (2015)

A conceptually simple method to explore the features of a CNN is visualizing their intermediate activation maps, e.g., by optimizing the pixel values of random noise input images to maximizing the activation of a specific convolution filter in the model [32, p. 304], [50, sec. 10.1]. However, while giving insights into the general learning process of CNNs, these visualizations are very ambiguous, abstract, and do not allow explicitly relating individual activations to dedicated interpretable concepts [50, sec. 10.1.4]. Furthermore, generating these visualizations is very cumbersome and it might be impossible to grasp the combined effect of all filter activations due to the sheer number of individual units in deep CNNs [50, sec. 10.1.4].

2.5.2 Saliency maps

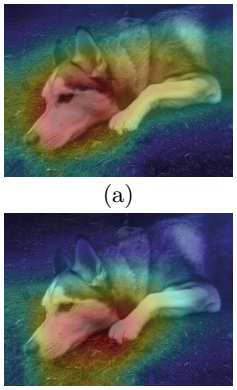
Another, more sophisticated tool than visualizing intermediate feature activations is to compute *saliency maps*, which can be used to highlight areas or structures in the input image that contributed the most to an individual class of the model prediction, as depicted in Figure 2.20. There

are many different methods to compute saliency maps, some of which are Grad-CAM [76] and Guided Backprop [77], which exploit the backpropagation of the gradients for the predicted or selected class and are either directly computed w.r.t. the input image or projected to it [50, sec. 10.2]. These were, for instance, used to show that the model focuses on the tokens used to decode patient orientation in X-ray scans that are placed differently between institutions [66] or manual markings to be indicative of malignant melanoma [78]. However, the benefit of saliency maps is very controversial [27]–[29], [79]. That is, they do not give insights to *what* exactly is of interest in the region highlighted but only that there is *something* of interest *to the model*, leaving the user behind with the decision if the highlighted region indeed shows decision-relevant features or not [80]. Often, they highlight ambiguous areas in the image that do not align with human intuition about which part of the image actually is important to correctly classify it, as, e.g., can be seen in Figure 2.20 (b) and (c) where parts of the background, the edges within the image, and also the dog seem to be important to classify the cat. Moreover, they tend to provide identical explanations for multiple classes or provide explanations for classes that are not present in the image as depicted in Figure 2.21.

[76] Selvaraju *et al.*, "Grad-CAM: visual explanations from deep networks via gradient-based localization" (2019)

[77] Springenberg *et al.*, "Striving for simplicity: the all convolutional net" (2015)

[78] Winkler *et al.*, "Association between surgical skin markings in dermoscopic images and diagnostic performance of a deep learning convolutional neural network for melanoma recognition" (2019)



(a)

(b)

FIGURE 2.21: Evidence according to a saliency map for (a) a Siberian husky and (b) a transverse flute. Figure reproduced from [79] with permission.

[27] Adebayo *et al.*, "Sanity checks for saliency maps" (2018)

[28] Arun *et al.*, "Assessing the trustworthiness of saliency maps for localizing abnormalities in medical imaging" (2021)

[29] Saporta *et al.*, "Benchmarking saliency methods for chest x-ray interpretation" (2022)

[80] Ghassemi *et al.*, "The false hope of current approaches to explainable artificial intelligence in health care" (2021)

2.5.3 Surrogate models

Moreover, global surrogate models, i.e., intrinsically interpretable models such as decision trees, or linear and logistic regression can be trained to approximate the output of a more complex black-box model [50, sec. 8.6]. However, using global surrogate models are only approximations that cannot capture the black-box model in every detail. Otherwise, the black-box model would not be required and the interpretable model could be used instead. Consequently, also the explanations of the surrogate have at least in parts to be erroneous, which could render the explanation untrustworthy failing their intended use [79].

Similarly, local surrogate models, such as local interpretable model-agnostic explanations (LIME) [81], can be used. These are trained to capture the local output of a black-box algorithm for individual samples [50, sec. 9.2]. Consequently, the model itself can be interpreted to explain the black-box prediction. This local model can either be trained on a set of features derived from perturbing the current input that is to be explained, or by

local interpretable model-agnostic explanations

[81] Ribeiro *et al.*, "'Why should I trust you?': explaining the predictions of any classifier" (2016)

reusing neighboring samples from the training data weighting their importance to the local explanation model by the distance to the current sample by, e.g., using an exponential kernel. For using CNNs and image data, instead of randomly perturbing single pixels, which are not expected to change the output of the model drastically,[4] pixel clusters with, e.g., similar colors called superpixels could be switched on or off [50, sec. 9.2.3]. A striking disadvantage of this approach is its dependency on the selection of the neighboring data or the perturbation applied to the sample causing instabilities and unreliable explanations [50, sec. 9.2.5].

[4] However, as introduced above when discussing adversarial examples, this could indeed happen.

Shapley additive explanations

[82] Lundberg and Lee, "A unified approach to interpreting model predictions" (2017)

Similarly, Shapley additive explanations (SHAP) [82] can be used to assign an importance value to each feature of the prediction for a particular sample [50, sec. 9.6]. The score is based on the concept of Shapley values, which measure how much a single feature on average contributes to the model's prediction across *all* possible combinations of feature values. Being grounded on game theory, Shapley values give rise to holistic explanations [50, sec. 9.5]. Moreover, global model explanations can be derived by, for example, averaging the instance-wise computed Shaply values across all training samples, allowing to compute a global feature importance for the trained model [50, sec. 9.6]. However, exactly computing Shapley values is prohibitively expensive for high-dimensional feature spaces and they usually have to be approximated using Monte Carlo sampling [50, sec. 9.6]. Another prominent approximation is called Kernel SHAP, which adapts LIME in combination with a linear regression model but replaces the exponential with the Shapley kernel, which allows the computation of the Shapley values from the resulting weighted linear regression, albeit again ignoring feature dependencies [82]. Another disadvantage of SHAP is that the computation of the Shapley values requires the availability of the data, which might conflict with privacy policies and — despite the profound underlying theory — can be misleading [50, sec. 9.6].

2.5.4 Prototype and concept learning

So far only post-hoc explaination methods were considered either to globally explain the model or individual data instances. In contrast to this, a large research community focuses on developing intrinsically interpretable models that go beyond such simple models as linear regression or decision trees. These have been shown to provide comparable performance to ordi-

nary black-box NNs, an interface to the model that allows inspecting the reasoning of the model for individual decisions, and global interpretations for the model. Thereby, these are computed ante-hoc, i.e., the explanations are directly derived during prediction, in the form of, e.g., prototypes [83]–[85] or concepts [36], [86] that are either implicitly learned from the training data or explicitly encoded in the model design as intermediate feature representations. Although the former typically yield human-comprehensible explanations without requiring additional labeled data, they might not always be fully faithful and might lack precision w.r.t. the explanation [87]. Furthermore, they were observed to show irrelevant background regions and lack similarity between the prototype and the actual image patch leading to unclear explanations [88] or are susceptible to image compression artifacts that break the network's reasoning [89].

[83] Li *et al.*, "Deep learning for case-based reasoning through prototypes: a neural network that explains its predictions" (2018)

[84] Chen *et al.*, "This looks like that: deep learning for interpretable image recognition" (2019)

[85] Carmichael *et al.*, "Pixel-grounded prototypical part networks" (2024)

[36] Koh *et al.*, "Concept bottleneck models" (2020)

[86] Chen *et al.*, "Concept whitening for interpretable image recognition" (2020)

[87] Gautam *et al.*, "This looks more like that: enhancing self-explaining models by prototypical relevance propagation" (2023)

[88] Davoodi *et al.*, "On the interpretability of part-prototype based classifiers: a human centric analysis" (2023)

[89] Hoffmann *et al.*, "This looks like that... Does it? Shortcomings of latent space prototype interpretability in deep networks" (2021)

In contrast to learning the prototypes from the data, methods such as CBM [36] exploit concept annotations instead, which are provided together with the image's class label. That is, CBMs explicitly incorporate these concepts as an intermediate representation. This representation is optimized to align with the target concept annotations in a supervised manner. The final prediction is then derived based on these concepts. In addition to intrinsically providing explanations for the predictions of the model, the bottleneck design promises to mitigate shortcut learning, i.e., to improve model generalization and reduce the susceptibility to adversarial attacks [90] and was observed to be very data efficient [36]. Moreover, the concepts can additionally be used to intervene the prediction by correcting them and, afterwards, updating the prediction of the model [36]. This furthermore allows, e.g., creating counterfactual explanations (examples for which the model predicts a different outcome) in order to analyze the relevance and influence of specific concepts to the model's prediction [50, sec. 10.3.5]. A more detailed introduction of the promising CBM approach will be given in Chapter 6 in which the method is adapted to build a transparent, uncertainty-aware, and lightweight DL-based NN for DR grading.

concept bottleneck model

[90] Ciravegna *et al.*, "Logic explained networks" (2023)

intervenability

2.6 Bayesian deep learning

The following section will introduce the types and importance of uncertainty, particularly in the context of automatic decision-making in the medical domain, followed by discussing the insufficient uncertainty calibration

and awareness of deterministic NNs and the benefit of exploiting Bayesian inference through BNNs to improve these. Parts of this section have previously been published in a peer-reviewed journal article [45] first-authored by the author of this thesis. He conceived and performed the methodology, experiments, and analyses, and wrote the manuscript in collaboration with the co-authors of the publication, who provided guidance and feedback on the methodology, experiments, and manuscript.

[45] Siebert *et al.*, "Uncertainty analysis of deep kernel learning methods on diabetic retinopathy grading" (2023)

2.6.1 Importance and types of uncertainty

To allow a well-informed rejection of uncertain predictions, improve detection of model failures, minimize misclassification risk, and improve predictive transparency of the system, high-quality uncertainty information is required that reliably communicates whether a NN *does or does not know*. Typically, as displayed in Figure 2.22, two types of uncertainty are distinguished: the irreducible *aleatoric* uncertainty caused by inherent data noise, and the reducible *epistemic* uncertainty resulting from a lack of knowledge about the true model, which is therefore also known as model uncertainty [91], [92]. Both these types of uncertainty information provide a distinct value for decision-making: Exemplarily for a classification task as in Figure 2.22 (b), a properly calibrated aleatoric uncertainty estimate can indicate an inherently difficult decision, i.e., whether a sample is very close to a decision boundary separating classes with partly overlapping support in the training data. Additionally, epistemic uncertainty allows inferring whether or not similar examples to the sample at hand were included in the training data, i.e., the given decision boundary is backed up by sufficient training data, and, thus, a prediction can be made with high evidence [91], [92]. With this, well-calibrated epistemic uncertainty could enable the detection of samples with different pathologies related to other diseases that are unfamiliar to the model as similar examples were missing from the training data, which is called out-of-distribution (OOD) detection. In this regard, both the aleatoric and epistemic predictive uncertainty are important for building trust and enabling seamless and transparent usage, particularly within the high-stakes medical domain [11], [93]. Moreover, epistemic uncertainty could be used for identifying new data samples for active learning [94] purposes, as well as detecting data shifts [95], which could provide benefit to a continual learning setting where the model is

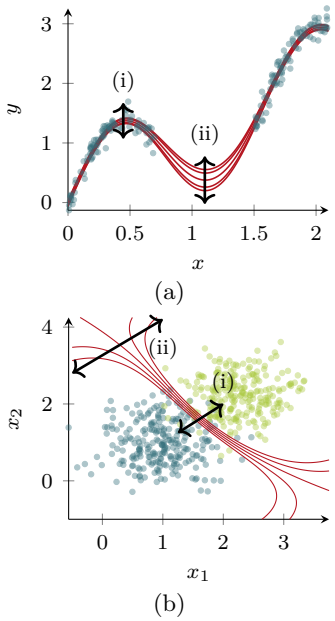


FIGURE 2.22: Schematic visualization of (i) aleatoric and (ii) epistemic uncertainty for (a) a regression and (b) classification task example. The red lines (—) show a set of selected, different possibilities for either regressed curves or potential decision boundaries. Blue (●) and green (●) dots denote the training samples.

aleatoric and epistemic uncertainty

[91] Smith and Gal, "Understanding measures of uncertainty for adversarial example detection" (2018)

[92] Hüllermeier and Waegeman, "Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods" (2021)

[93] Bhatt *et al.*, "Uncertainty as a form of transparency: measuring, communicating, and using uncertainty" (2021)

[94] Houlsby *et al.*, "Bayesian active learning for classification and preference learning" (2011)

[95] Ovadia *et al.*, "Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift" (2019)

adaptively trained on new data generated by the automatic referral system. Hence, providing both reliable aleatoric and epistemic uncertainty estimates is highly desirable for such a screening system as subject to this work. It sometimes might not be possible to compute these individually, or the distinction between them might be blurred and dependent on the data representation or task hypothesis [92]. Therefore, frequently, only the total predictive uncertainty is measured, which comprises a mixture of both aleatoric and epistemic uncertainty [91], [92].

2.6.2 Uncertainty in deterministic neural networks

Ordinary NNs, however, usually do not provide sufficiently calibrated uncertainty estimates in either of the two: Recall the definition of the dataset $\mathcal{D} = \{\mathbf{X}, \mathbf{Y}\}$ with $\mathbf{X} \in \mathbb{R}^{N \times D}$, $\mathbf{Y} \in \mathbb{R}^{N \times C}$, and N samples introduced in section 2.2.2. Moreover, the dataset's underlying, unknown distribution is approximated using a NN $f(\cdot)$ according to

$$\mathbf{y}_i \approx p\left(y_c = 1 \mid \mathbf{x}_i, \boldsymbol{\theta}\right) = \sigma(f\left(\mathbf{x}_i, \boldsymbol{\theta}\right)) \tag{2.51}$$

for $i = 1, \ldots, N$ and with the model likelihood $p\left(y_c = 1 \mid \mathbf{x}, \boldsymbol{\theta}\right)$[5], the softmax activation $\sigma(\cdot)$ according to (2.14), and the weights $\boldsymbol{\theta} = \{\boldsymbol{\theta}_{\mathrm{FE}}, \boldsymbol{\theta}_{\mathrm{FC}}\}$ of the NN. Then, from a Bayesian perspective, optimizing the weights $\boldsymbol{\theta}$ on the given data yields a maximum a posteriori (MAP) estimate

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta} \in \Theta} p\left(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\theta}\right) p\left(\boldsymbol{\theta}\right), \tag{2.52}$$

with the prior probability distribution over the model weights $p(\boldsymbol{\theta})$. This can be induced by, e.g., applying L2-norm regularization [33], [96]. The MAP estimate considers only a single possible, locally optimal mapping function corresponding to a specific set of parameters that fits the training data well leading, for instance, to one of the decision boundaries depicted in Figure 2.22 (b). However, there typically are many different models that achieve a similar or — due to the nonconvexity of the optimization problem — even better training error. Moreover, NNs extrapolate to regions far away from the training data and yield a specific decision boundary which results in predicting with high confidence in regions with no data evidence [97]. Following the high flexibility of the NNs and the stochasticity of the training process, all these different solutions are likely to result in

[5] Note that in the remainder of this thesis the model likelihood $p\left(y_c = 1 \mid \mathbf{x}, \boldsymbol{\theta}\right)$ will be abbreviated as $p\left(y \mid \mathbf{x}, \boldsymbol{\theta}\right)$ for simplicity.

maximum a posteriori

[33] Jospin *et al.*, "Hands-on bayesian neural networks—a tutorial for deep learning users" (2022)

[96] Blundell *et al.*, "Weight uncertainty in neural network" (2015)

[97] Hein *et al.*, "Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem" (2019)

very different decision boundaries far away from the training data. There is no means of deciding which model performs best outside the training distribution — a phenomenon, which is called model underspecification [11], [98, p. 639]. Due to the ignorance of the MAP estimate of this parametric uncertainty, NNs cannot inform the user about the lack of data support and model uncertainty, i.e., they can only express aleatoric uncertainty [33]. However, caused by the NN's susceptibility to overfitting [63], it even often provides overconfident predictions and fails to provide well-calibrated aleatoric uncertainty estimates [73].

model underspecification

[98] Murphy, *Probabilistic Machine Learning: Advanced Topics* (2023)

2.6.3 Bayesian neural networks

A possible solution to this lack of uncertainty awareness would be the use of BNNs, which, in contrast, assume the model parameters to be uncertain in order to model a posterior distribution over the parameters. This can be derived by updating the prior belief on the distribution of the model parameters $p(\boldsymbol{\theta})$ by the model's likelihood $p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\theta})$ w.r.t. to the given data using Bayes' theorem

Bayes' theorem

$$p(\boldsymbol{\theta} \mid \mathbf{X}, \mathbf{y}) = \frac{p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(\mathbf{y} \mid \mathbf{X})}. \tag{2.53}$$

Therbey, $p(\mathbf{y} \mid \mathbf{X}) = \int p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\theta}') p(\boldsymbol{\theta}') \, d\boldsymbol{\theta}'$ is called the marginal likelihood or model evidence. Computing the Bayesian model average (BMA)

Bayesian model average

$$p(y \mid \mathbf{x}^*, \mathbf{X}, \mathbf{y}) = \int p(y \mid \mathbf{x}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta} \mid \mathbf{X}, \mathbf{y}) \, d\boldsymbol{\theta} \tag{2.54}$$

at prediction for a test sample $\mathbf{x}^*$, i.e. aggregating the model predictions $p(y \mid \mathbf{x}^*, \boldsymbol{\theta})$ for *all* possible sets of parameters weighted by their posterior probability $p(\boldsymbol{\theta} \mid \mathbf{X}, \mathbf{y})$ to yield the given training data, leads to the BNN's posterior predictive distribution that inherently captures parameter, hence, epistemic uncertainty [33]. This process is also known as marginalization [99] and, in a nutshell, could be imagined as taking a weighted average of the models which lead to the regressed curves and decision boundaries displayed in Figure 2.22 and all other mapping functions contained in the space of possible solutions. Moreover, Bayesian treatment of the parametric uncertainty by means of the BMA promises to mitigate model overfitting [100, pp. 161–165] and yield better-calibrated aleatoric

[99] Abdar *et al.*, "A review of uncertainty quantification in deep learning: techniques, applications and challenges" (2021)

[100] Bishop, *Pattern Recognition and Machine Learning* (2006)

uncertainty estimates [33]. While BNNs are mathematically well-defined, computing the posterior is often analytically intractable [33], [99].

Markov chain Monte Carlo

Markov chain Monte Carlo (MCMC) methods, such as the Hamilton Monte Carlo (HMC) algorithm, asymptotically guarantee to sample from the exact posterior [33] and are therefore often used as gold-standard for uncertainty quantification. However, they are computationally very expensive as they require an unknown number of iterations to converge [99] and do not scale well with large data and models [33]. Alternatively, the posterior can be approximated using variational inference (VI), i.e., optimizing a variational distribution $q_{\boldsymbol{\theta}_{\text{VI}}}(\boldsymbol{\theta})$ parameterized in $\boldsymbol{\theta}_{\text{VI}}$ to be close to $p(\boldsymbol{\theta} \mid \mathbf{X}, \mathbf{y})$ by minimizing the Kullback-Leibler (KL) divergence [33]

variational inference

$$\mathcal{L}_{\text{VI}}(\boldsymbol{\theta}_{\text{VI}}) = \text{KL}\left[q_{\boldsymbol{\theta}_{\text{VI}}}(\boldsymbol{\theta}) \parallel p(\boldsymbol{\theta} \mid \mathbf{X}, \mathbf{y})\right] \tag{2.55}$$

and computing the BMA by marginalizing over the variational distribution of the intractable exact posterior $p(\boldsymbol{\theta} \mid \mathbf{X}, \mathbf{y})$ according to [99]

$$p(y \mid \mathbf{x}^*, \mathbf{X}, \mathbf{y}) \approx \int p(y \mid \mathbf{x}^*, \boldsymbol{\theta}) q_{\boldsymbol{\theta}^*_{\text{VI}}}(\boldsymbol{\theta})\, d\boldsymbol{\theta}. \tag{2.56}$$

Yet, directly optimizing the KL divergence in (2.55) again involves computing the intractable exact posterior. As an approximation, optimal variational parameters $\boldsymbol{\theta}^*_{\text{VI}}$ can be obtained by maximizing the evidence lower bound (ELBO) of the model [99]

evidence lower bound

$$\begin{aligned} \mathcal{L}_{\text{ELBO}}(\boldsymbol{\theta}_{\text{VI}}) &= \int q_{\boldsymbol{\theta}_{\text{VI}}}(\boldsymbol{\theta}) \log p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\theta})\, d\boldsymbol{\theta} - \text{KL}\left[q_{\boldsymbol{\theta}_{\text{VI}}}(\boldsymbol{\theta}) \parallel p(\boldsymbol{\theta})\right] \\ &\leq \log p(\mathbf{y} \mid \mathbf{X}), \end{aligned} \tag{2.57}$$

which maximizes the evidence of the data marginalized over q and, in contrast to (2.55), solely involves minimizing the KL divergence between the variational and posterior probability distributions $q_{\boldsymbol{\theta}_{\text{VI}}}(\boldsymbol{\theta})$ and $p(\boldsymbol{\theta})$. Mostly, optimization of the ELBO is accomplished using stochastic variational inference (SVI) [33] by exploiting SGD, i.e., computing the ELBO per minibatch, which allows scaling the VI algorithm to large data and models, e.g., by applying the Bayes-by-backprop algorithm [96].

stochastic variational inference

Despite their computational superiority to MCMC, VI methods may be less accurate due to sampling from the approximate variational distribution q at test time instead of the true posterior p. Moreover, both MCMC and VI

"*In fact, we shall see that we can get uncertainty information from existing deep learning models for free — without changing a thing.*"—*Y. Gal* [101]

methods still have several disadvantages and are more difficult to deploy compared to standard DL methods [33]. Hence, simple and in practice frequently used Bayesian approximations are deep ensembles (DEs) [102] and Monte Carlo dropout (MCD) [103], which are derived from standard-practice DL algorithms.

[102] Lakshminarayanan *et al.*, "Simple and scalable predictive uncertainty estimation using deep ensembles" (2017)

[103] Gal and Ghahramani, "Dropout as a bayesian approximation: representing model uncertainty in deep learning" (2016)

Monte Carlo dropout

Adopting the same training scheme as for traditional model training where dropout is used as regularization [63], MCD can be deployed by additionally dropping activations at test time with the same probability as during training, which corresponds to drawing Monte Carlo samples from the model's underlying parameter distribution and can be interpreted as a form of approximate VI under specific assumptions [103]. As applying dropout for BNN training would not be considered as regularization but as being part of the variational posterior, additionally applying, e.g., L2-norm regularization is necessary to form the prior $p(\boldsymbol{\theta})$ over the parameter distribution [33]. Although the implementation and training of MCD-based BNNs are straightforward, the method can only capture local uncertainty and, hence, might not capture the true posterior well [33].

deep ensembles

Similarly, DEs, i.e., averaging multiple independently optimized deterministic model instances as described in section 2.4.1 — can capture different modes of the true posterior due to the stochasticity of the training process. However, each model instance of the ensemble represents a Dirac delta sample of the posterior that in turn is not able to capture local uncertainty [33], [104] and, thus, does not allow computing the ELBO [33]. Whereas Lakshminarayanan *et al.* state that building ensembles would not be Bayesian, both DEs and MCD can be considered to resemble a simplistic approximate BMA, which showed to improve both predictive performance and the quality of the uncertainty estimates upon simple MAP estimators [33], [95], [99], [104], [105] and sampling from a single mode of the posterior by, for instance, using VI [98, p. 650]. Despite providing strong baselines and the simplicity w.r.t. model implementation and training, both methods increase the computational complexity by either having to sample the model when using MCD or to run inference repeatedly for each ensemble member as well as loading and storing a multitude of model parameters in case of using DEs.

[104] Wilson and Izmailov, "Bayesian deep learning and a probabilistic perspective of generalization" (2020)

[105] Ashukha *et al.*, "Pitfalls of in-domain uncertainty estimation and ensembling in deep learning" (2020)

3 | Causes and treatment of diabetic retinopathy

As diabetic retinopathy (DR) is a frequent complication of diabetes mellitus (DM), the following section 3.1 will first introduce the basic pathogenesis and global societal impact of DM. Subsequently, in section 3.2, the pathology and treatment of DR as well as an overview of the implications and challenges for the medical care system w.r.t. the screening for the disease caused by the rising amount of patients that suffer from DM will be given, motivating the need for automated screening methods.

3.1 Diabetes mellitus

DM is a pandemic, metabolic disorder that is characterized by hyperglycemia resulting from a deficient or entirely impeded insulin secretion, an impaired insulin action, or a combination of both [106]. In 2021, approximately 537 million people worldwide (approx. 10 %) were affected by DM [107]. The majority of patients suffer either from type 1 DM (T1DM) or type 2 DM (T2DM) [106], whereby the latter make up about 90 % of all diabetes cases [108]. Due to increasingly prolonged lifespans, sedentary lifestyle changes such as reduced physical activity and high-calorie diet, and increasing urbanization, the prevalence of DM is expected to rise over the next years [108], [109], [110] projecting more than 780 million people being affected by DM in 2045 [107].

Insulin is a hormone produced by the β-cells in the pancreatic islets of Langerhans that mediates the cellular glucose uptake [111]. T1DM is characterized by the destruction of these β-cells, which is typically caused by an autoimmune response, is associated with a genetic predisposition, and commonly manifests in juvenile patients, leading to an absolute lack of insulin secretion and, hence, to increasing blood glucose levels [112][106]. In contrast, in T2DM, hyperglycemia results from impaired insulin action

[106] Punthakee *et al.*, "Definition, classification and diagnosis of diabetes, prediabetes and metabolic syndrome" (2018)

[107] Magliano *et al.*, *IDF Diabetes Atlas* (2021)

[108] Saeedi *et al.*, "Global and regional diabetes prevalence estimates for 2019 and projections for 2030 and 2045: results from the International Diabetes Federation Diabetes Atlas, 9th edition" (2019)

[109] Teo *et al.*, "Global prevalence of diabetic retinopathy and projection of burden through 2045" (2021)

[110] Zheng *et al.*, "Global aetiology and epidemiology of type 2 diabetes mellitus and its complications" (2017)

[111] Wilcox, "Insulin and insulin resistance." (2005)

[112] American Diabetes Association, "Diagnosis and classification of diabetes mellitus" (2011)

caused by insulin resistance, i.e., a decreased insulin sensitivity of the cells, and later on can lead to a secretory defect [106]. Often, it is a manifestation of a metabolic syndrome that among others is characterized by obesity and hypertension [106]. Accordingly, associated risk factors for T2DM are age, obesity, diet, a lack of physical activity, as well as a genetic predisposition [108], [112].

Due to the lack of insulin production, insulin replacement therapy is required for patients with T1DM to regulate their blood glucose level as well as the glucose uptake of cells, to prevent acute complications such as life-threatening ketoacidosis, and to survive [112]. In contrast, as insulin secretion and action often remain at least partially functional, therapeutic approaches and prevention for patients suffering from T2DM or prediabetes primarily aim to control the blood glucose level, prevent the occurrence of comorbidities by lifestyle interventions, as well as avoid the total failure of insulin secretion and only revert to pharmacologic treatment if the non-pharmacologic treatment fails [20]. The practical guidelines on the treatment of both T1DM and T2DM, for example of the American Diabetes Association (ADA) [20] or the German Diabetes Association (Deutsche Diabetes Gesellschaft, DDG) [21] as well as the german National Healthcare Guidelines (Nationale Versorgungs Leitlinie, NVL) [22], stipulate continuous monitoring, adaption, and discussion of the personalized management plan that include regular screening by specialists for DM associated complications. Thereby, a poorly controlled, chronic hyperglycemia severely increases the risk for micro- and macrovascular damage, affecting — among others — the eye, kidney, as well as the cerebral and cardiovascular vessel system leading to higher morbidity and mortality within the diabetic patient population and, thus, is associated with a decreased life expectancy and quality of life [106], [113], [114, p. 100].

[20] American Diabetes Association Professional Practice Committee, "4. Comprehensive medical evaluation and assessment of comorbidities: standards of care in diabetes—2024" (2023)

[21] Deutsche Diabetes Gesellschaft (DDG), "S3-Leitlinie Therapie des Typ-1-Diabetes, Version 5" (2023)

[22] Bundesärztekammer *et al.*, "Nationale VersorgungsLeitlinie Typ-2-Diabetes" (2023)

[113] World Health Organization, "Definition and diagnosis of diabetes mellitus and intermediate hyperglycaemia: report of a WHO/IDF consultation" (2006)

[114] Hien *et al.*, *Diabetes-Handbuch* (2013)

3.2 Diabetic retinopathy

DR is the most frequent complication of DM [115] for both T1DM and T2DM and is one of the leading causes of blindness in the working-age population [109], [116]. The global prevalence among patients with DM is estimated to be at about 22.27 % to 35.4 % [109], [116], [117] with over 103 million people being affected by DR in 2020 [109], which, however, varies significantly, e.g., geographically [109]. The DR prevalence for pa-

[115] Simó-Servat *et al.*, "Diabetic retinopathy in the context of patients with diabetes" (2019)

[116] Solomon *et al.*, "Diabetic retinopathy: a position statement by the american diabetes association" (2017)

[117] Ting *et al.*, "Diabetic retinopathy: global prevalence, major risk factors, screening practices and public health challenges: a review" (2016)

tients suffering from T1DM is estimated to be higher than for those with T2DM [117], [118] and is higher in rural areas [117]. Moreover, it is strongly correlated to the duration of DM, the quality of the glycemic control, i.e., the presence of chronic hyperglycemia, and suffering from hypertension [115], [116], [119]. Thereby, the genetic heritability and frequent variations in the glycemic level were found to have an additional influence on the susceptibility of patients with DM to develop DR [115]. Other yet not fully examined risk factors are assumed to exist [115].

[118] Haider *et al.*, "Disease burden of diabetes, diabetic retinopathy and their future projections in the UK: cross-sectional analyses of a primary care database" (2021)

[119] Yau *et al.*, "Global prevalence and major risk factors of diabetic retinopathy" (2012)

There are ambiguities about the development of the DR prevalence, i.e., there are both studies that are predicting a decrease [117] and studies that are predicting an increase [118] of the fraction of people suffering from DR over time. Nonetheless, as the number of individuals diagnosed with DM is expected to significantly grow in the future, also the number of patients at risk of developing DR is expected to increase: Assuming a constant prevalence of about 20 %, more than 160 million people might be affected by DR in 2045 [109]. With annual costs of USD$493 million arising for direct costs associated with US citizens suffering from DR in 2004 [120], this increase is expected to impose a growing financial burden in addition to the increased workload on the healthcare system [109], [115]. This is particularly concerning for countries with limited medical care. As a consequence, screening patients at risk for early signs of DR is important to prevent vision loss and progression of DR by timely treatment and reduce indirect costs related to vision loss.

[120] Rein, "The economic burden of major adult visual disorders in the United States" (2006)

3.2.1 Pathogenesis and classification

DR is a frequent complication of DM, which results from a microangiopathy that affects the retinal vascular and, finally, neural system [116], [121, p. 301]. With the progression of the disease and of vascular sclerosis, the structural damage to the vessel walls can cause capillary occlusion and, consequently, retinal ischemia [122], [121, p. 301]. Furthermore, it can cause an increased vascular permeability induced by a loss of vascular endothelial cells and pericytes as well as the thickening of the basal membrane [122]. In addition, there is growing evidence that neurodegeneration, i.e., apoptosis of retinal neurons, which leads to retinal malfunction might happen even prior to any microvascular abnormalities [115], [122]. Typical early signs of the onset of microvascular degeneration are microaneurysms

[121] Grehn, *Augenheilkunde* (2019)

[122] Nian *et al.*, "Neurovascular unit in diabetic retinopathy: pathophysiological roles and potential therapeutical targets" (2021)

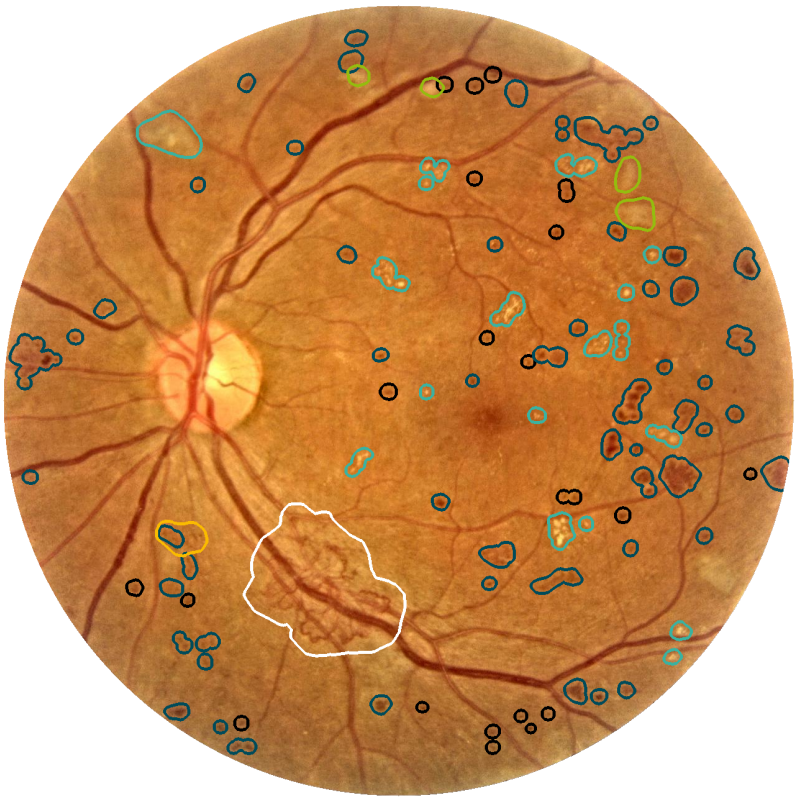


FIGURE 3.1: A color fundus image showing the central part of the retina along with the most important DR-related pathologies, i.e., microaneurysm (MA) (●), haemorrhage (HE) (●), hard exudate (HX) (●), cotton wool spot (CWS) (●), neovascularization (NV) (○), and intraretinal microvascular abnormality (IRMA) (●), as seen in ophthalmoscopy. Image adapted from [125].

(MAs), i.e., small, local dilations of the retinal vessels, which can rupture and result in increased leakage of blood components causing retinal haemorrhages (HEs) and exudative lesions such as hard exudates (HXs) [123, p. 249] and finally leads to non-perfusion of capillaries followed by retinal ischemia [122]. These early retinal lesions can often be cured by timely treatment and usually do not cause visual impairment [121, p. 301]. However, if remaining untreated the diabetic, pathologic changes of the retina worsen and cotton wool spots (CWSs) (swelling of the axons in the retinal nerve fiber layer also often referred to as soft exudates), venous beading (VB) (alternating changes of the venous caliber), and intraretinal microvascular abnormalitys (IRMAs) (capillary widening adjacent to capillary occlusions) arise as early signs for retinal ischemia [124], [121, p. 302]. Finally, the hypoxia can induce a proliferative vessel growth, i.e., neovascularization (NV), triggered through the dissemination of vascular endothelial growth factors (VEGFs) to compensate for the reduced perfusion, which in turn increases blood leakage forming vitreous and preretinal HEs, damages the retinal neurons [116], [121, p. 301], and may lead to detachment of the retina as

[123] Walter and Plange, *Basiswissen Augenheilkunde* (2017)

[124] Kollias and Ulbig, "Diabetic retinopathy" (2010)

Table 3.1: The ICDR severity scale [38] with its respective ETDRS classification levels [117], [127].

DR severity level		Clinical findings	Corresponding ETDRS levels
0	no apparent DR	—	Level 10 (DR absent)
1	mild NPDR	MAs only	Level 20 (very mild DR, MAs only)
2	moderate NPDR	More than mild (1), less than severe (3)	Levels 35, 43, 47 (mild, moderate, and moderately severe NPDR including intraretinal HEs, HX, and CWSs as well as mild to moderate VB and IRMAs)
3	severe NPDR	No signs of PDR and any of the following: (i) > 20 intraretinal HEs in each quadrant (ii) VB in 2+ quadrants (iii) IRMAs in 1+ quadrant	Level 53 (severe NPDR)
4	PDR	Presence of NV or vitreous/preretinal HEs	Levels 61,65,71,75,81,85 (PDR, high-risk PDR, very severe or advanced PDR)

well as irreversible blindness [122], [121, p. 303]. Figure 3.1 shows an example image of a retina that comprises the most important DR-associated lesions for estimating disease progression as introduced above.

Diabetic macular edema

A complication of DR is the diabetic macular edema (DME). It is defined as a blood vessel leakage (edema), which affects the macular, i.e., the retinal area with the highest density of photoreceptors at whose center the so-called fovea is located. Consequently, it can cause a rapid loss of visual acuity [121, p. 305]. In general, this can manifest at any DR severity (sDR) level but commonly emerges with the presence of more severe DR [122].

However, both advanced stages of DR and DME could remain asymptomatic for a long time [23] followed by a rapid worsening of visual acuity, which can lead to blindness due to the advanced progression of the disease severity. This highlights the urgency to adhere to the proposed screening routines. That is, yet showing no signs of DR, patients with DM should be screened annually for early signs such as MAs, and monitored with a higher frequency with the progression of the disease [116].

[23] Wong *et al.*, "Guidelines on diabetic eye care: the international council of ophthalmology recommendations for screening, follow-up, referral, and treatment based on resource settings" (2018)

disease severity classification

The guidelines proposed within the Early Treatment Diabetic Retinopathy Study (ETDRS) [126], [127] are often referred to as gold standard to classify the sDR levels . This classification scheme, however, is rather complicated for use in clinical practice [38], [117]. Therefore, DR is usually categorized according to the less complex ICDR disease severity grading scheme [38] into five severity stages according to Table 3.1: (0) no apparent retinopathy, (1) mild-, (2) moderate-, and (3) severe non-proliferative DR (NPDR), as well as (4) proliferative DR (PDR).

[126] Davis *et al.*, "Studies of retinopathy: methodology for assessment and classification with fundus photographs" (1985)

[127] Early Treatment Diabetic Retinopathy Study Research Group, "Fundus photographic risk factors for progression of diabetic retinopathy: etdrs report number 12" (1991)

[38] Wilkinson *et al.*, "Proposed international clinical diabetic retinopathy and diabetic macular edema disease severity scales" (2003)

3.2.2 Treatment

[128] Wykoff *et al.*, "Risk of blindness among patients with diabetes and newly diagnosed diabetic retinopathy" (2021)

Despite being one of the most frequent reasons for loss of vision [109], [115]–[117], [128], DR can be therapeutically treated well, i.e., delayed, and even reversed if detected early [116]. The onset and progression of DR typically can be prevented or delayed by minimizing the above-introduced risk factors, particularly, by tightly managing the glycemic level and lowering blood pressure [116]. More severe stages can actively be treated, i.e., proliferation can be slowed or stopped with intravitreous injection of anti-VEGF. Moreover, applying panretinal laser photocoagulation to patients with moderate NPDR or worse can prevent or reverse vascular proliferation by reducing ischemia whereby the retina mostly remains functional [121, p. 306]. However, there is still a lack of sufficient understanding of the pathogenesis. Improvements in the research of the disease could further refine treatment [115], [122] and lower the probability of blindness that particularly increases with the DR severity [128] as shown in Figure 3.2. This, again, emphasizes the importance of screening patients affected by DM for early signs of DR [117] and — in context of the rising demand — to improve access and cost efficiency of existing screening programs that currently are primarily conducted by a comparably low number of ophthalmic specialists.

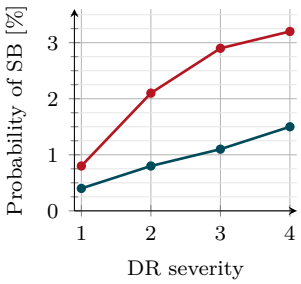


FIGURE 3.2: Estimated probability to develop substantial blindness (SB) after one (—) and two years (—) for patients being newly diagnosed with the respective DR severity and good visual acuity according to [128].

Following the proposed treatment scheme, referable DR (rDR), sometimes also referred to as more-than-mild DR (mtmDR), usually comprises the severity stages moderate, severe, and proliferative DR (2-4). These are the severity levels at which patients should be examined by specialists in order to discuss potentially required treatment options aside from the optimization of risk factors. Vision-threatening DR (vtDR) accordingly comprises the sDR levels severe NPDR (3) and PDR (4) that require urgent treatment [23], [117].

4 | Transparency through visualizing affected areas

Adding to the motivation in the introduction of this thesis, this chapter starts giving an outline of the importance of visualizing affected areas for a successful implementation of an AI-based DR screening system that allows for human supervision, as well as the challenges and state-of-the-art methods for implementing a lightweight but high-performance segmentation model in section 4.1. Subsequently, in section 4.2, the methodology for the analysis of the utilized segmentation model conducted in this chapter is described, followed by the results and a discussion of the findings in section 4.3. Finally, section 4.4 summarizes the chapter's findings and provides an outlook on future research directions to improve upon the proposed architecture for lightweight DR lesion segmentation. Parts of this chapter have previously been published in a peer-reviewed conference paper [41] first-authored by the author of this thesis, for which he conceived and performed the methodology, experiments, and analyses, and wrote the manuscript. The paper was prepared in collaboration with the co-author of the publication, who provided guidance and feedback on the methodology, experiments, and manuscript.

[41] Siebert and Rostalski, "Performance evaluation of lightweight convolutional neural networks on retinal lesion segmentation" (2022)

4.1 Motivation and related work

Based on the importance of screening diabetic patients regularly for early signs of the disease due to the high risk of people suffering from diabetes to evolve DR [109], [117] and the rising trend of patients suffering from DM, DL-based diagnostic algorithms could be leveraged to detect the presence and severity of DR. With this, general practitioners could help screening patients with DM in rural areas where access to a specialized medical examination is limited [117] and lower the equally growing screening workload on ophthalmologists. DL-assisted DR detection systems are of high research

[109] Teo *et al.*, "Global prevalence of diabetic retinopathy and projection of burden through 2045" (2021)

[117] Ting *et al.*, "Diabetic retinopathy: global prevalence, major risk factors, screening practices and public health challenges: a review" (2016)

interest due to the excellent diagnostic performance that can be achieved using NNs.

However, as introduced in Chapter 1 and discussed in more detail in section 2.5, the lack of transparency impairs the trust in black-box DL algorithms and the expressive power of the diagnoses. This is inconvenient for highly safety-critical domains such as medicine, in which it is important to understand why a prediction was made, to verify the decision, and to prevent severe misdiagnoses. Moreover, as required for instance by the regulatory frameworks for AI [15] and data protection [17], this kind of transparency is not only a desirable model feature but is mandatory in the EU for market authorization. Thus, providing insights into the decision-making and visualizing intermediate results should be mandatory to allow clinicians to supervise the automatic DL system and verify or challenge the model's prediction. In particular, highlighting retinal lesions present in the patient's retina and associated with the potentially vision-threatening disease can improve the physician's understanding of the DL system's prediction. This allows for challenging the model of whether the disease is present and to asses how reliable the prediction is [11]. To this end, the DL grading system can be accompanied by or ideally be built based on a segmentation model that provides fine-grained masks of DR related lesions in addition to the predicted DR grade.

[15] European Commission, Directorate-General for Communications Networks, Content and Technology, "Artificial Intelligence Act" (2021)

[17] European Parliament, Council of the European Union, "General Data Protection Regulation" (2016)

[11] Petersen *et al.*, "Responsible and regulatory conform machine learning for medicine: a survey of challenges and solutions" (2022)

However, a disadvantage of typically applied CNNs in the biomedical regime is the increased computational requirement compared to regular classification models resulting from the large spatial resolution of the intermediate activation maps computed in the encoder and decoder part of the CNN. To evade the additional high cost of high-performance computing hardware that can run state-of-the-art DL models, developing lightweight, mobile, edge device applicable screening systems with good performance gains increasing attention, as the deployment of those is less expensive and can improve usability and availability in areas with limited access to medical care.

Therefore, this work applies and analyzes the suitability of various complex instances of the U^2-Net [39] for the task of DR related lesion segmentation by evaluating its predictive and computational performance regarding a possible future edge device implementation. The U^2-Net is a recent extension to the famous U-Net [40], which is known to achieve high performance for segmenting biomedical images. According to the authors, the

[39] Qin *et al.*, "U^2-Net: going deeper with nested U-structure for salient object detection" (2020)

[40] Ronneberger *et al.*, "U-Net: convolutional networks for biomedical image segmentation" (2015)

U^2-Net shows state-of-the-art performance for salient object detection when trained from scratch, while being designed to maintain low memory cost and keep input image resolution high. Hence, the U^2-Net promises to be a good baseline for building a high-performant, lightweight segmentation model that could be applied to edge devices, and — due to rooting back to the original U-Net — bears the potential to generalize well on medical image data, particularly for the task of lesion segmentation. In detail, by applying mobile application-optimized convolutions, i.e., DC [129], varying the network capacity as well as evaluating the impact of multi-task and ensemble learning, the objective of this chapter is to assess and evaluate (a) the suitability of the U^2-Net to the task of biomedical image segmentation, (b) the segmentation performance when training the U^2-Net on small data from scratch, (c) the benefit of single-, dual-, and multi-task training, and (d) the segmentation accuracy and computational load of the differently complex instances of the model and, hence, the suitability of the architecture for edge device implementation. To this end, the performance of the baseline U-Net is compared to the U^2-Net and state-of-the-art results in the literature on the one hand. On the other hand, the various computationally complex models proposed in this work and deployed in differing task settings are benchmarked to find a good trade-off between computational cost and the segmentation performance regarding the retinal lesions.

[129] Howard *et al.*, "MobileNets: efficient convolutional neural networks for mobile vision applications" (2017)

4.1.1 Related work

With the rising performance of deep learning algorithms over the last two decades, developing artificial intelligence-based automatic medical diagnostic algorithms gained tremendous research interest. This applies to the task of DR detection as well, leading to the first algorithm being approved for use in clinical care [10]. In addition to that, also mobile DR detection systems [130] using lightweight neural networks such as the MobileNetV2 [131] are brought into research focus.

As introduced in section 2.4, deep learning algorithms are not guaranteed to generalize to real-world application data despite their excellent performance and their black-box character does in general not allow the stakeholders to understand the model's reasoning. Therefore, some recently proposed studies exploit saliency methods to highlight areas in the retinal fundus that contributed to the model's prediction the most [130],

[10] Abràmoff *et al.*, "Improved automated detection of diabetic retinopathy on a publicly available dataset through integration of deep learning" (2016)

[130] Natarajan *et al.*, "Diagnostic accuracy of community-based diabetic retinopathy screening with an offline artificial intelligence system on a smartphone" (2019)

[131] Sheikh and Qidwai, "Using MobileNetV2 to classify the severity of diabetic retinopathy" (2020)

[132]–[135]. Other studies leverage lesion-attentive approaches to guide the disease severity grading and detection [136]–[138] or derive the prediction based on the detected DR related lesions [10], [30], [35]. As will be discussed in more detail in section 6.4, computing precise segmentation masks of lesion-specific pathologic areas in the retina could give further insights and, as a result, improve classification. This, however, is a highly demanding task due to the lack of a sufficient amount of annotated data as well as the lesions being commonly very small and also sometimes poorly distinguishable from noise and other retinal structures and lesions.

To circumvent the problem of data availability for DR lesion segmentation tasks, semi- or weakly-supervised as well as adversarial training strategies can be used to leverage information of image-level graded datasets [138], [139] such as EyePACS [140] or Messidor [141]. Particularly, Playout *et al.* [139] propose to train a U-Net for red and bright lesion segmentation in a regular supervised manner but with an additional binary classifier added at the model's bottleneck that is trained to detect if any lesion is present in the image. By interpreting images with a DR severity score greater than 0 to have at least a single lesion, they exploit the large number of images contained in the EyePACS dataset using weak supervision for model training. Moreover, Zhou *et al.* [138] propose a semi-supervised, adversarial training strategy that exploits both pixel- and image-level annotated data to improve both DR-related multi-lesion segmentation and DR grading. To this end, they use a complex model pipeline including a U-Net, which uses efficient Xception [142] blocks, i.e., DC that are bypassed by a residual connection including a 1×1 convolution. To explicitly account for an edge device suitable segmentation, Guo *et al.* [143] propose the LWENet architecture with about 1.9 M parameters only comprising little downsampling and, instead, using inception-inspired [144] multi-scale convolutional modules for lightweight HXs segmentation. The model was trained in a two-stage training procedure that exploits image level DR grades to pretrain the feature extractor of the segmentation model and is running at approximately 11.1 fps for an input image size of (1440×960) pixels on an Nvidia GTX 1080Ti.

Accounting for the small lesion size of, for example, MA and HXs, high-resolution input images and multi-scale approaches can boost segmentation performance by fusing important information from local and global image features into the model's prediction [145]–[147]. In detail, Guo *et al.* [147]

[132] Gargeya and Leng, "Automated identification of diabetic retinopathy using deep learning" (2017)

[133] Gondal *et al.*, "Weakly-supervised localization of diabetic retinopathy lesions in retinal fundus images" (2017)

[134] Quellec *et al.*, "Deep image mining for diabetic retinopathy screening" (2017)

[135] Wei *et al.*, "Learn to segment retinal lesions and beyond" (2021)

[136] Yang *et al.*, "Lesion detection and grading of diabetic retinopathy via two-stages deep convolutional neural networks" (2017)

[137] Wang *et al.*, "Zoom-in-Net: deep mining lesions for diabetic retinopathy detection" (2017)

[138] Zhou *et al.*, "Collaborative learning of semi-supervised segmentation and classification for medical images" (2019)

[30] De Fauw *et al.*, "Clinically applicable deep learning for diagnosis and referral in retinal disease" (2018)

[35] Quellec *et al.*, "Explain: explanatory artificial intelligence for diabetic retinopathy diagnosis" (2021)

[139] Playout *et al.*, "A novel weakly supervised multitask architecture for retinal lesions segmentation on fundus images" (2019)

[140] Dugas *et al.*, "Diabetic retinopathy detection" (2015)

[141] Decencière *et al.*, "Feedback on a publicly distributed image database: the messidor database" (2014)

[142] Chollet, "Xception: deep learning with depthwise separable convolutions" (2017)

[143] Guo *et al.*, "A lightweight neural network for hard exudate segmentation of fundus image" (2019)

[144] Szegedy *et al.*, "Going deeper with convolutions" (2015)

[145] Yan *et al.*, "Learning mutually local-global U-nets for high-resolution retinal lesion segmentation in fundus images" (2019)

[146] Sarhan *et al.*, "Multi-scale microaneurysms segmentation using embedding triplet loss" (2019)

[147] Guo *et al.*, "L-seg: an end-to-end unified framework for multi-lesion segmentation of fundus images" (2019)

propose the L-Seg model based on a pretrained [61] backbone to perform multi-lesion segmentation that exploits fusing multi-scale features from different stages of the VGG16 to keep the high-resolution information of the (1440×960)-sized input image to produce fine-grained segmentation maps. Moreover, Yan *et al.* [145] propose a Dual-U-Net architecture fusing the multi-scale information of a local and global U-Net, processing both high-resolution image patches and the complete downscaled image in parallel to prevent the loss of important information and boosting the segmentation performance of small lesions. Similarly, Sarhan *et al.* [146] exploit multiple NNs working on different scales to increase the model's receptive field while simultaneously retaining the full image resolution. For the final MA lesion prediction, they fuse the segmentation maps of these models.

[61] Simonyan and Zisserman, "Very deep convolutional networks for large-scale image recognition" (2015)

4.2 Methods

In the following section, firstly the utilized study data and applied preprocessing will be introduced in sections 4.2.1 and 4.2.2. Subsequently, details on the U^2-Net architecture, its implementation, and the conducted model scaling will be provided in sections 4.2.3 to 4.2.5. Finally, the experimental setup, i.e., the training and evaluation protocol deployed to analyze the U^2-Net architecture for the task of DR lesion segmentation (sections 4.2.6 and 4.2.7), will be described.

4.2.1 Study data

For training, the publicly available IDRiD [34] is used in this chapter.[6] Besides containing 516 high-resolution retinal fundus images with annotations for sDR grading, the dataset comprises 81 retinal fundus images accompanied with fine-grained segmentation masks of the — particularly in early stages — most important biomarkers for detection of DR, i.e., microaneurysms (MAs), haemorrhages (HEs), hard exudates (HXs), as well as cotton wool spots (CWSs). All image samples $\mathbf{x} \in \mathcal{X}$ and their corresponding lesion masks $\mathbf{s} \in \mathcal{S}$ were included within this subset of the IDRiD dataset upon consensus of the reviewers. Adding to the lesions masks, pixel-level annotations of the optic disc (OD), as well as center coordinates of the latter and the fovea are additionally included for all images of the segmentation subset. The dataset is published with a predefined training/testing split with 54/27 images, which will be adhered to within the

[34] Porwal *et al.*, "Indian diabetic retinopathy image dataset (IDRiD): a database for diabetic retinopathy screening research" (2018)

[6] The data is publically available and was hosted in the context of the Diabetic Retinopathy Segmentation and Grading Challenge at the IEEE International Symposium on Biomedical Imaging (ISBI-2018) workshop. `https://idrid.grand-challenge.org/` (Last accessed: 11/14/2023)

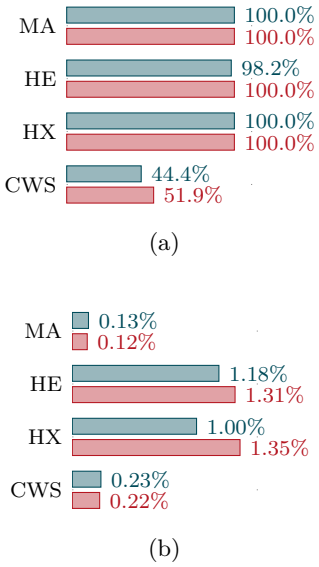


FIGURE 4.1: Distribution of (a) the retinal images that contain the respective lesions and (b) of the foreground pixels for each class in the training (■) and test (■) data of the IDRiD dataset.

following experiment. The distribution of the total available lesions for the training and testing data set as well as the percentage of the foreground pixels per lesion are shown in Figure 4.1. 20 % of the training data are randomly selected as validation data at the start of each training run to track the training performance and conduct model selection. This results in a total number of 43 images used for model training.

4.2.2 Image preprocessing and augmentation

The applied image preprocessing comprises cropping the images to the visible retinal disc with a semi-automatic algorithm using binary thresholding, resizing the images to (512×512) pixels, and applying contrast limited histogram equalization (CLAHE). To this end, the input images are firstly converted to the HSV color space and afterwards, the CLAHE operation is applied to the value channel to equally enhance the contrast of all image color channels. Examples of the contrast-enhanced images from the IDRiD dataset with their corresponding lesions are provided in Figure 4.2. After converting the images back into RGB color space, the images are standardized to have zero mean and unit variance.

As the training dataset only comprises 43 images in total, strong data augmentation is applied at training time to the input images to artificially enlarge the available training data and reduce model overfitting. In detail, the images of each minibatch are augmented online by random affine transformations, i.e., random image rotation with $\pm 90°$, scaling by $\pm 40\,\%$, and translation by $\pm 10\,\%$ w.r.t. the maximum image width I_w and height I_h as well as horizontal and vertical flipping both with probability $p = 0.5$.

4.2.3 Image segmentation models

To combine both high segmentation performance and low computational cost while training on a small medical image dataset from scratch, this work implements the U^2-Net with different model capacities and analyzes its suitability for DR lesion segmentation w.r.t. edge device implementation. As a reference model, the U-Net is used as baseline architecture.

U-Net for biomedical image segmentation

As introduced in section 2.3.4 and depicted in Figure 2.17, the latter is an encoder-decoder-shaped fully convolutional network explicitly designed for biomedical image segmentation on a small amount of data [40]. The encoder is built to extract meaningful features using blocks that sequentially apply

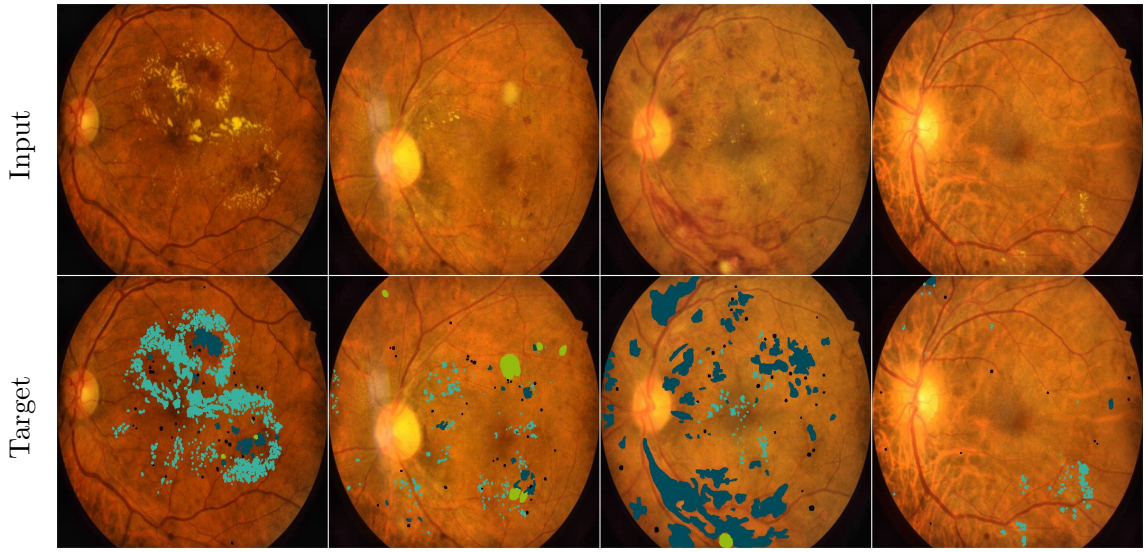
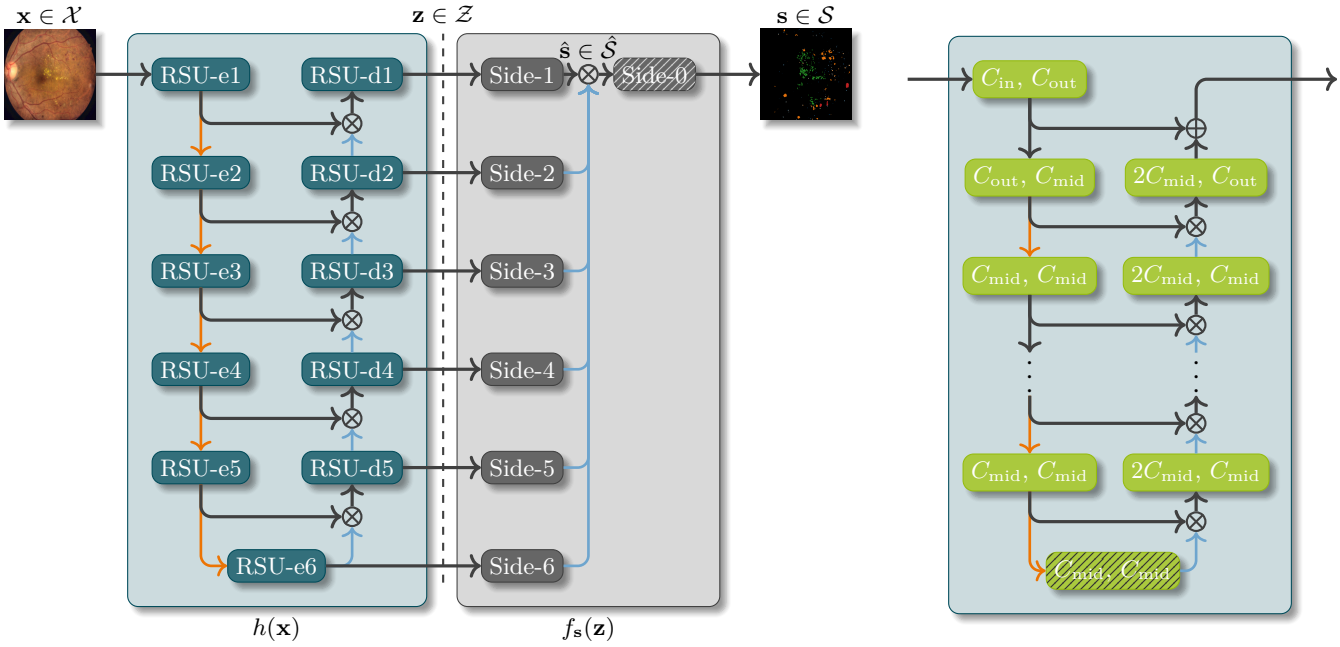


FIGURE 4.2: Exmples of the preprocessed images of the IDRiD [34] dataset in the first row with corresponding segmented lesions MAs (●), HEs (●), HXs (●), CWSs (●) in the second row.

a convolution layer followed by the nonlinear ReLU activation while scaling down the input image resolution using max-pooling operations. Thereby, the feature depth is doubled at each downsampling stage to increase the receptive field of the model. Mirroring the encoder structure, the decoder reconstructs fine-grained segmentation masks from the learned latent encoding by using skip connections to exploit high-resolution features from the intermediate layers in the encoder. This enables the U-Net to use both global and local features to produce high-resolution and precise segmentation masks in a single forward pass of the network requiring only a few images, which is beneficial for the task of biomedical image segmentation where data is usually scarce [40].

U^2-Net — a nested U-Net

Adopting the encoder-decoder structure with the intermediate skip connections, Qin *et al.* [39] propose the U^2-Net, displayed in Figure 4.3a, that makes use of a nested U-Net structure, which is referred to as residual U-block (RSU) as depicted in Figure 4.3b. The core concept of these U-shaped blocks is to extract rich multi-scale features already at early layers while retaining the high resolution of the input activation maps leading to an increased model depth and, hence, capacity, which Qin *et al.* observed to outperform the vanilla U-Net. Thereby, the computational cost of the RSU block better scales with the block's feature depth compared to other frequently adapted convolutional modules, i.e., residual-, dense-, or



(a) U^2-Net (b) RSU-X

FIGURE 4.3: Schematic diagram of the U^2-Net architecture. Legend: RSU-X (■), Conv(3×3)+BN+ReLU (■), dilated Conv(3×3)+BN+ReLU (▨), Conv(1×1) (■), Conv(1×1)+sigmoid (▨), max-pooling (→), upsampling (→), copy (→), concatenation (⊗), addition (⊕).

inception-inspired blocks, while performing similarly or better [39]. Hence, despite significantly increasing the total number of model parameters, the deployment of the nested RSU blocks only slightly increases inference time and memory consumption while reducing the computational costs in terms of multiply-accumulates (MACs) operations compared to the vanilla U-Net as will be shown in section 4.2.5.

To further improve model performance, Qin *et al.* designed the U^2-Net to predict a set of multi-scale features that are aggregated by linear (1×1) convolutions to first predict global and local saliency maps and finally the final segmentation masks. That is, the encoder-decoder backbone of the U^2-Net learns a function $h : \mathcal{X} \to \mathcal{Z}$ that maps the input images $\mathbf{x} \in \mathcal{X}$ to a number of latent multi-scale features $\mathbf{z} \in \mathcal{Z}$.[7] Second, the saliency maps $\hat{\mathbf{s}} \in \hat{\mathcal{S}}$ and the lesions segmentation masks $\mathbf{s} \in \mathcal{S}$ are derived by the shallow mapping function $f_{\mathbf{s}} : \mathcal{Z} \to \hat{\mathcal{S}} \to \mathcal{S}$. To derive the saliency maps, additional deep supervision $(\mathcal{L}_1, \ldots, \mathcal{L}_6)$ is applied to the side outputs of each resolution stage within the decoder while scaling the side outputs to the target mask's resolution beforehand. As a result, the Qin *et al.* claim a

[7]Please note that images will henceforth be denoted as a vector ($\mathbf{x}$). That is, in contrast to the correct notation for images as a 3d-tensor ($\mathbf{X}$) as introduced section 2.3.4 the dimensions corresponding to the image's spatial resolution are neglected for notational convenience.

good predictive performance with using the U^2-Net preserving fine-grained structures, enabling training on small data from scratch, and alleviating the design of lightweight, mobile architectures due to the independence of large pretrained backbones and the comparably small computational load. With this, they also show the U^2-Net to outperform sequentially cascaded U-Nets, such as the stacked hourglass network [148], which follow a similar idea, i.e., allowing the model to fuse global and local feature information to improve model performance.

[148] Newell *et al.*, "Stacked hourglass networks for human pose estimation" (2016)

4.2.4 Model implementation

For the following analysis, all models are implemented and trained using the PyTorch [149] framework.

[149] Paszke *et al.*, "PyTorch: an imperative style, high-performance deep learning library" (2019)

In this work, the baseline model is adapted according to the U-Net architecture [40] comprising five resolution stages with 64-, 128-, 256-, 512-, and 1024-dimensional feature depth, respectively. Thereby, BN layers are added to each convolutional layer prior to applying the nonlinear activation. Moreover, all convolutional layers are implemented to use zero-padding in order to maintain the input resolution throughout the model, i.e., so that the output matches the input resolution.

U-Net — implementation

The two U^2-Net variants as introduced by Qin *et al.* are implemented without changes, which are herein referred to as U^2-Net-O and U^2-Net-M. Both networks apply five consecutive RSU blocks with subsequent max-pooling operations in the encoder as well as an additional RSU block at the model's bottleneck. The two networks only differ in the number of features and, hence, model size and capacity.

U^2-Net — implementation

In addition, the U^2-Net's feature depth is further scaled down by reducing the RSU block's number of features by 50 % and 75 % w.r.t. the U^2-Net-M. The two resulting lightweight model instances will in the following be referred to as U^2-Net-S and U^2-Net-XS. A detailed description of the model parameterization is given in Table 4.1.

U^2-Net — depth scaling

Moreover, in addition to each U^2-Net variant, a model that exploits DC is implemented. This specific convolutional layer factorizes a regular convolution into two sequential operations, i.e., a depth- and a point-wise convolution. While the former processes the input spatially, i.e., applies a single filter for each channel or a group of channels, the latter uses a 1×1 convolution to expand or reduce the channel depth of the depth-wise

U^2-Net — depthwise separable convolution

TABLE 4.1: Setup of the differently scaled U^2-Nets. The first and second column displays the number of input and middle feature channels of each RSU block ($e_i, i = 1, \ldots, 6$) in the encoder, respectively. The decoder modules ($d_i, i \in 1, \ldots, d_5$) are adapted so that the number of features in the RSU module of the decoder d_i equals that of the encoder e_i for $i \in 2, \ldots, 5$ and the third column denotes the output features for RSU-d$_1$. The fourth column displays the number of downscaling operations applied in each RSU block, where the * indicates that dilated convolutions are used as replacement for consecutive convolution+pooling layers operations in the respective RSU blocks.

	C_{in}						C_{mid}						C_{out}	RSU depth					
Model	e_1	e_2	e_3	e_4	e_5	e_6	e_1	e_2	e_3	e_4	e_5	e_6	d_1	e_1	e_2	e_3	e_4	e_5	e_6
U^2-Net-O	3	64	128	256	512	512	16	32	64	128	256	256	64	7	6	5	4	4*	4*
U^2-Net-M	3	64	64	64	64	64	16	16	16	16	16	16	64	7	6	5	4	4*	4*
U^2-Net-S	3	32	32	32	32	32	8	8	8	8	8	8	32	7	6	5	4	4*	4*
U^2-Net-XS	3	16	16	16	16	16	4	4	4	4	4	4	16	7	6	5	4	4*	4*

convolution. This lowers the number of parameters, memory footprint, and required MAC operations significantly while retaining a large quantity of the convolutional capacity compared to regular convolutional layers. Hence, DCs are frequently used, e.g., in the MobileNet [129] and its subsequent versions, to reduce the computational cost of the model architecture and enable edge device application while keeping high performance. To evaluate if this sparse convolution can be beneficial for further reduction of the U^2-Net's computational load while maintaining sufficient performance, every convolution within the differently sized models is interchanged with a DC, except for each RSU block's first and the final U^2-Net's output convolution. These models will subsequently be referred to as U^2-Net-O-DC, U^2-Net-DC-M, U^2-Net-DC-S, and U^2-Net-DC-XS.

multi-task learning

Although one might assume that single-task (ST) segmentation outperforms multi-task (MT) optimization as each model could specialize to a single lesion and multi-task learning might pose a more complex optimization problem, the presence of the DR lesions is not independent. Therefore, multi-task training could potentially boost the segmentation performance if the model has enough capacity to capture all necessary features for all four segmentation tasks simultaneously [32, p. 255]. Moreover, performing multi-task lesion segmentation can help to further decrease memory and computational cost. Hence, to analyze the potential benefit of multi-task learning and whether the hard parameter sharing between the individual tasks impairs model performance, multi-lesion segmentation models based on the regular U^2-Net variants are set up in addition to the single-task model instances that are trained to segment all four available lesions in the IDRiD dataset, i.e., MAs, HEs, HXs, and CWSs. Adding to the multi-task

[32] Bishop and Bishop, *Deep Learning: Foundations and Concepts* (2024)

Model	DC	Frame rate [fps]	MACs [G]	Parameters [M]	Model size (fp32) [MB]	GPU-RAM [MB]
U-Net	-	33.81 (0.32)	218.95	31.04	124.17	586.94
U^2-Net-O	-	33.27 (0.18)	141.53	43.87	175.48	798.34
	✓	42.69 (0.28)	80.63	17.66	70.65	692.76
U^2-Net-M	-	62.50 (0.82)	51.01	1.13	4.52	579.36
	✓	63.65 (0.23)	33.09	0.66	2.63	576.50
U^2-Net-S	-	62.69 (0.18)	13.03	0.29	1.14	289.80
	✓	63.87 (0.72)	8.72	0.18	0.70	290.48
U^2-Net-XS	-	61.71 (0.13)	3.39	0.07	0.29	147.45
	✓	64.44 (0.23)	2.41	0.05	0.20	147.43

TABLE 4.2: Overview of the computational complexity of the implemented single-task networks at inference running on a GeForce RTX 3090 (24 GB) with an AMD Epyc 7413 (128 GB) using PyTorch (v1.8.0) and a randomly sampled input image of size ($3 \times 512 \times 512$) for 500 iterations. The median and interquartile range (IQR) are given for the measured frame rate.

setup, also dual-task (DT) training will be examined, i.e., the models are optimized to either segment the tuple of MAs and HEs or HXs and CWSs simultaneously, which might ease model optimization but retains the important dependency between each of the two similar appearing lesions. With this, inference time and the overall number of model parameters as well as the computational cost for model training can be significantly reduced by half and a quarter, respectively, when compared to using four individual CNNs in parallel.

4.2.5 Computational complexity of the implemented models

In Table 4.2, the different single-task model architectures are compared to each other w.r.t. their computational demands at inference time. To assess the viability of the networks for potential mobile implementation, a series of metrics have been employed. These include the frames per second (fps) measured during inference, as well as the computational complexity expressed as the number of MAC operations, the number of parameters, the network storage size for a single-precision model, and the RAM allocated on the GPU by PyTorch during inference are reported. The required number of MAC operations of the models is approximated using the toolbox *thop* [150]. Thereby, the computational cost of the dual- and multi-task models nearly equals that of a single model in single-task-configuration, introducing only a marginal computational overhead as only the final convolutional layers are affected by changing the number of output classes. Note, however, that using multiple single-task or dual-task models in parallel to segment all four lesions is quadrupling or doubling the reported computational cost,

[150] Zhu, "Lyken17 / pytorch-OpCounter" (2022)

respectively, and either the runtime or the memory consumption compared to using a single multi-task model.

Considering for instance an Nvidia Jetson Nano, i.e., a very power-efficient and lightweight microcomputer paving the integration of AI to edge devices with 4 GB of shared CPU and GPU memory, an ensemble of three U^2-Net-XS models in single-task mode (~1.78 GB) or U^2-Net-S models in dual-task mode (~1.74 GB) are expected to seamlessly run on such a device, provided that the individual model instances are executed in parallel. In contrast, the larger model variants potentially exceed the available memory considering that some memory will be reserved by the OS and the required code libraries. Moreover, the computational complexity of a single U^2-Net-XS model is comparable to that of the EfficientNet-B0 [151], a state-of-the-art CNN classifier designed for mobile application. That is, the U^2-Net-XS requires less MAC operations (2.41 GMACs vs. 2.16 GMACs) and has a lower memory usage (147.45 MB vs. 182.64 MB) although comprising a dedicated decoder for deploying both models with the identical input resolution.

[151] Tan and Le, "EfficientNet: rethinking model scaling for convolutional neural networks" (2019)

4.2.6 Training protocol

general training hyperparameters

To analyze the performance of all the U^2-Net variants for the DR-related lesion segmentation in comparison to the baseline U-Net, each of the above-described models is trained ten times on the IDRiD dataset without pre-training. The training was conducted on an Nvidia Tesla-V100/A100 (32/-40 GB) along with an Intel(R) Xeon(R) Platinum 8168 CPU over 400 epochs. A single training interation takes about 1.5 h per model in single-task mode for the regular U^2-Net-O and about 1 h for the small U^2-Net-XS. For model optimization, Adam is used with a learning rate of 5e−4, L2-norm regularization of 1e−4 and a plateau scheduler with a factor of 0.75 and patience of 15 epochs. In the case of dual- and multi-task training, each class is treated as a separate task, i.e., the loss is computed on the channel-wise sigmoid-activated model output instead of correlating the output across the lesions using softmax activation. This was observed to yield better results within preliminary experiments. Furthermore, the number of epochs and the patience of the scheduler are increased to 600 and 800 as well as 30 and 50 for dual- and multi-task training respectively, to ensure model convergence.

loss function

The class distribution of the fore- and background classes is heavily imbalanced, which would have to be accounted for. That is, the number of background pixels severely exceeds that of the foreground pixels across the lesions. Moreover, the distribution of the foreground class differs between the classes, i.e., small and less frequent (or even often absent) lesions such as MAs and CWSs make up less volume compared to larger and more frequent lesions. Given these imbalances, the error function has to be designed with great care to ensure that the model pays equal attention to all lesions, and does not simply predict that every pixel is part of the background.

To this end, a weighted fusion of the Dice loss $\mathcal{L}_{\mathrm{D}}$ and the focal binary cross-entropy (fBCE) [152] $\mathcal{L}_{\mathrm{fBCE}}$ is applied to the predicted segmentation mask $\mathbf{s}_c = p(s_c = 1 \mid \mathbf{x}, \boldsymbol{\theta})$ over the current minibatch according to

[152] Lin *et al.*, "Focal loss for dense object detection" (2017)

$$\mathcal{L}_\eta(\tilde{\mathbf{s}}, \mathbf{s}) = \frac{1}{C} \sum_{c}^{C} \Big(\alpha \cdot \mathcal{L}_{\mathrm{fBCE}} \left(\tilde{\mathbf{s}}_c, \mathbf{s}_c, \gamma \right) + (1 - \alpha) \cdot \mathcal{L}_{\mathrm{D}} \left(\tilde{\mathbf{s}}_c, \mathbf{s}_c \right) \Big) \cdot \tilde{w}_c \tag{4.1}$$

with the target $\tilde{\mathbf{s}}_c$ for c-th class, and $\tilde{w}_c$ being a weighting factor only active during dual- and multi-task training, boosting the learning of underrepresented classes with low volume in the current minibatch. In detail, the weight is computed as $\tilde{w}_c = w_c / \sum_{c' \in C} w_{c'}$ with $w_c = n_{\mathrm{bg},c} / n_{\mathrm{fg},c}$ and $n_{\mathrm{fg}}, n_{\mathrm{bg}}$ denoting the number of fore- and background pixels in the current target mask $\tilde{\mathbf{s}}$. Thereby, w_c is set to zero in case a class is absent throughout the minibatch, i.e., $n_{\mathrm{fg}} = 0$, to prevent the model from degrading to constantly predict empty masks, which mainly affects CWS segmentation as this lesion is only present in about half the images of the IDRiD dataset.

deep supervision

To enable deep supervision of the model, the error function according to (4.1) is applied to the saliency maps $\hat{\mathbf{s}} \in \hat{\mathcal{S}}$ of the U^2-Net at every depth layer of the network, i.e., to the outputs of Side-η with $\eta \in \{1, \ldots, H\}$, in addition to the final model output $\mathbf{s} \in \mathcal{S}$ (Side-0). Following the implementation by [39], the total loss applied to the loss is computed as

$$\mathcal{L} = \frac{1}{H+1} \sum_{\eta=0}^{H} \mathcal{L}_\eta. \tag{4.2}$$

applying equal weights to the loss of each side output and the final model output.

focal binary cross-entropy

For using the fBCE, the regular BCE as introduced in (2.11) has to be adopted to

$$\mathcal{L}_{\text{fBCE}}(\tilde{\mathbf{s}}_c, \mathbf{s}_c, \gamma) = -\frac{1}{BI_hI_w}\sum_{b}^{B}\sum_{h}^{I_h}\sum_{w}^{I_w}\tilde{\mathbf{s}}_\zeta(1-\mathbf{s}_\zeta)^\gamma \log(\mathbf{s}_\zeta) + \tilde{\mathbf{s}}_\zeta \mathbf{s}_\zeta^\gamma \log(1-\mathbf{s}_\zeta) \quad (4.3)$$

for the minibatch comprising B samples and with $\zeta = c, b, h, w$. Thereby, the newly introduced scaling factor γ forces the model to concentrate on more difficult cases and improves the robustness of the loss to the class imbalance between fore- and background pixels [152]. With setting $\gamma = 0$ the fBCE reverts to the regular BCE. In contrast, the loss can be tuned to be less sensitive to the large proportion of easy-to-classify background (or foreground) pixels by setting $\gamma > 0$, as the weight for the loss of a specific pixel in the predicted segmentation map $\mathbf{s}_c$ becomes smaller the closer the predicted value for that pixel is to zero (or to one). In this work, the fBCE was used with $\gamma = 1$.

Dice loss

[153] Ma *et al.*, "Loss odyssey in medical image segmentation" (2021)

Analogously, the Dice loss is chosen in addition to the fBCE as it is inherently robust to class imbalances between fore- and background pixels [153] and is more indicative to the main target metrics, which will be applied to measure model performance for segmentation tasks as described in section 4.2.7. The Dice loss can be derived from the Dice similiarty coefficient (DSC)

$$\text{DSC}(\tilde{\mathbf{s}}_c, \mathbf{s}_c) = \frac{2|\tilde{\mathbf{s}}_c \cap \mathbf{s}_c|}{|\tilde{\mathbf{s}}_c| + |\mathbf{s}_c|} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} \quad (4.4)$$

that is measuring the overlap of the predicted and target segmentation masks with the number of true positive (TP), false positive (FP), and false negative (FN) predicted pixels in the minibatch output $\mathbf{s}_c$ and $|\cdot|$ counts the elements in its input that exceed a predefined threshold [154]. Note that the DSC is independent of the true negative (TN) predicted pixels that in medical image segmentation tasks make up the largest fraction of pixels. Hence, the score is insensitive to such an imbalance between the fore- and background pixels. By using the dot product $\langle\cdot,\cdot\rangle$ operator instead of computing $|\tilde{\mathbf{s}}_c \cap \mathbf{s}_c|$, $|\tilde{\mathbf{s}}_c|$, and $|\mathbf{s}_c|$, a differentiable soft DSC score [154] can be dervied, i.e.

[154] Sudre *et al.*, "Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations" (2017)

$$\text{sDSC}(\tilde{\mathbf{s}}_c, \mathbf{s}_c) = \frac{2\langle\tilde{\mathbf{s}}_c, \mathbf{s}_c\rangle + \epsilon}{\langle\tilde{\mathbf{s}}_c, \tilde{\mathbf{s}}_c\rangle + \langle\mathbf{s}_c, \mathbf{s}_c\rangle + \epsilon} \quad (4.5)$$

with ϵ being introduced to prevent numerical instabilities. With this, the Dice loss is given by

$$\mathcal{L}_{\mathrm{D}}(\tilde{\mathbf{s}}_c, \mathbf{s}_c) = 1 - \mathrm{sDSC}(\tilde{\mathbf{s}}_c, \mathbf{s}_c). \tag{4.6}$$

Note that $\tilde{\mathbf{s}}_c$ and $\mathbf{s}_c$ are the target and predicted segmentation masks over the complete minibatch, i.e., the loss is computed for the whole minibatch — in contrast to the fBCE score that computes the loss at pixel-level. However, both the DSC and Dice loss are sensitive to the total number of foreground pixels. That is, if only a few pixels belong to the foreground class, missing some of these causes a massive decrease in the observed DSC, which can lead to a noisy error and unreliable convergence. Since the Dice loss furthermore encourages overconfident and uncalibrated predictions [155], the balancing parameter between the two loss terms was chosen as $\alpha = 0.75$, prioritizing the fBCE to smooth the Dice loss and calibrate the predicted segmentation masks. This was observed in a set of manual experiments to yield a good tradeoff within preliminary experiments.

[155] Mehrtash *et al.*, "Confidence calibration and predictive uncertainty estimation for deep medical image segmentation" (2020)

4.2.7 Validation protocol

The model with the best validation DSC score computed with a fixed threshold of 0.5 was chosen for the final model evaluation. At test time, the performance of the predicted segmentation masks is measured in terms of area under the precision-recall curve (AUPRC), DSC, and Hausdorff-distance (HD) between predicted and target masks.

area under the precision-recall curve

Thereby, precision (P) and recall (R) are defined as $\mathrm{P} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FP})$ and $\mathrm{R} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FN})$ [32, p. 148]. That is, a high precision and recall indicate a low number false positives (FPs) and false negatives (FNs) in the predicted segmentation mask, respectively. Similar to the DSC, the AUPRC does not rely on the number of true negative predictions, which renders the metric — in contrast to the area under the receiver-operating-characteristic (AUROC) score — less susceptible to data imbalances in which the number of background pixels strongly outweighs that of the foreground class. As the harmonic mean of precision and recall — that is often also referred to as F1-Score — equals the DSC, it can be directly derived from the precision-recall curve. Hence, within the analysis of this chapter, the thresholds for mask binarization are derived from the lesion-

specific AUPRC test score to maximize the harmonic mean of both precision and recall for each class individually.

Hausdorff-distance

Due to the sensitivity of the DSC and, hence, the AUPRC to single-pixel-level changes in the case of only a few foreground pixels being present, the HD is computed as an additional metric. This, in contrast, measures the distances between the contours of the predicted and target regions belonging to the foreground class [156] and, hence, is less susceptible to a mismatch of a few pixels w.r.t. to the exact lesion boundaries. In detail, the HD is defined as [157, pp. 290–293], [158]

[156] Yeung *et al.*, "Unified focal loss: generalising dice and cross entropy-based losses to handle class imbalanced medical image segmentation" (2022)

[157] Hausdorff, *Grundzüge der Mengenlehre* (1914)

[158] Huttenlocher *et al.*, "Comparing images using the hausdorff distance" (1993)

$$\mathrm{HD}(\mathbb{C}, \mathbb{C}') = \max\left(\delta(\mathbb{C}, \mathbb{C}'), \delta(\mathbb{C}', \mathbb{C})\right) \tag{4.7}$$

$$\delta(\mathbb{C}, \mathbb{C}') = \max_{\mathfrak{c} \in \mathbb{C}} \min_{\mathfrak{c}' \in \mathbb{C}'} \|\mathfrak{c} - \mathfrak{c}'\|_2, \tag{4.8}$$

where $\mathbb{C}$ and $\mathbb{C}'$ are sets comprising the coordinates of the lesion border pixels for the predicted and target segmentation mask. In this work, the distance is computed using the *surface-distance* toolbox provided by DeepMind[8] that in contrast to (4.8) computes a robust variant of the HD, i.e., it estimates the 95 % percentile of the distances between the surfaces of the predicted and target lesion segmentations based on the covered total foreground area. As the HD is undefined if either one or both the prediction and target masks are empty, the surface distance is in this work set to the maximum possible distance between both masks, i.e., the image diagonal, when no lesion is present but the network's prediction is not empty and vice versa. This only affects the CWS segmentation as the other lesions are present in every image of the test set.

[8] `https://github.com/deepmind/surface-distance` (last accessed: 01/23/2022)

deep ensembles

Additionally, the effect of model ensembling on the tradeoff between predictive performance and computational complexity for every but the largest U^2-Net is evaluated. To build the DEs — that will be referred to as U^2-Net-M†, U^2-Net-S†, and U^2-Net-XS† — the outputs of the top three performing models based on the AUPRC score are averaged and evaluated according to the above validation scheme.

4.3 Results and Discussion

In the following section 4.3.1 first the results for the regular U^2-Net will be provided and discussed, followed by the analysis of the benefit introduced by using DCs in section 4.3.2. A detailed evaluation of the effect of dual-

Model	DC	MA $\mu_{1/2}$	MA IQR	HE $\mu_{1/2}$	HE IQR	HX $\mu_{1/2}$	HX IQR	CWS $\mu_{1/2}$	CWS IQR	$\overline{\text{AUPRC}}$
U-Net	-	**35.8**	(1.3)	54.0	(4.5)	75.2	(5.8)	66.0	(2.2)	57.8
U^2-Net-O	-	34.3	(2.9)	62.2	(3.3)	78.2	(11.4)	**73.5**	(2.8)	62.1
	✓	34.5	(2.2)	64.2	(2.9)	72.1	(11.9)	70.7	(1.9)	60.4
U^2-Net-M	-	35.6	(1.8)	**64.8**	(1.7)	**80.3**	(6.9)	71.5	(3.7)	**63.0**
	✓	35.0	(2.1)	62.5	(3.9)	70.7	(8.7)	72.2	(6.3)	60.1
U^2-Net-S	-	34.9	(2.4)	62.9	(2.3)	75.4	(9.9)	68.2	(8.6)	60.4
	✓	33.6	(3.7)	57.6	(3.4)	77.9	(4.2)	66.0	(5.8)	58.8
U^2-Net-XS	-	32.5	(1.5)	56.5	(3.9)	78.0	(8.8)	61.2	(10.7)	57.1
	✓	24.5	(16.4)	49.4	(22.3)	71.1	(13.7)	58.9	(12.7)	51.0

TABLE 4.3: Test results on the IDRiD dataset for the differently scaled U^2-Net models and their corresponding DC-based model architectures given as median ($\mu_{1/2}$) AUPRC [%] across the individual training runs per lesion with the IQR enclosed in brackets and the overall average median AUPRC ($\overline{\text{AUPRC}}$) [%] across all four lesions.

and multi-task training on the performance will be given in section 4.3.3. Finally, in section 4.3.4, the results will be compared to the related literature.

4.3.1 Performance analysis of the U^2-Net

The results of the analysis w.r.t. the U^2-Net feature depth scaling and the DC-based model variants are presented in Table 4.3, showing the lesion-specific as well as overall average median AUPRC score.

U^2-Net-O vs. U-Net

Having significantly more parameters and an additional depth layer compared to the U-Net, the U^2-Net-O is expectedly observed to outperform the U-Net on the lesion segmentation task by a relative performance gain of 7.4 % w.r.t. the average AUPRC score across all four lesions. Thereby, the most improvement is observed for the HE (15.2 %) and CWS (11.4 %) segmentation, while the HX segmentation is only slightly better (4.0 %). Conversely, the median MA segmentation performance is observed to be slightly worse compared to the U-Net (−4.2 %). However, despite the raw increase of model parameters by about 41.3 %, the U^2-Net-O saves about 35.4 % of multiply-accumulates (MACs) and, hence, retains nearly identical inference runtime, which highlights the high computational efficiency of the U^2-Net's architectural design.

scaled U^2-Nets

Notably, despite the reduced model size and capacity, the U^2-Net-M overall outperforms the U^2-Net-O by a relative performance increase of about 1.4 % w.r.t. the average AUPRC. As a result, the U^2-Net-M is observed to be on par with the regular U-Net concerning the MA segmentation. However, the CWS segmentation performance is observed to be not as good as for the U^2-Net-O. Expectedly, further downscaling of the U^2-Net's feature depth, i.e., using the U^2-Net-S and U^2-Net-XS, is overall observed to decrease the AUPRC performance. However, the average

AUPRC of the U^2-Net-XS is only marginally inferior compared to the regular U-Net ($-1.2\,\%$). Thereby, the size of the model architecture is reduced from 31.04 M to 0.07 M parameters, the computational complexity w.r.t. the MAC operations decreased by about 98.5 %, and the frame rate increased by a factor of approximately 1.8. Moreover, the U^2-Net-S is observed to achieve a relative performance increase by about 4.5 % over the U-Net while still reducing the required number of MAC operations by about 94 % and only comprising $< 1\,\%$ of the U-Net's parameters.

per-trial results

The top row in Figure 4.4 shows the corresponding boxplots to the results of the observed AUPRC performance per lesion and trial according to Table 4.3 as well as for the Dice similiarty coefficient (DSC) and Hausdorff-distance (HD). From these, it becomes apparent that the low performance of the U^2-Net-O compared to that of the U^2-Net-M mainly originates from a reduced HE segmentation performance and a high spread of the HX segmentation quality. This decrease in the AUPRC and DSC performance of the U^2-Net-O is, however, not visible in the observed HD performance, suggesting that the U^2-Net-O rather fails to precisely capture the exact lesion boundaries but still correctly detects the *presence* of the HEs as well as the U^2-Net-M. Similarly, the AUPRC and DSC performance for the HX segmentation show a high variation that is not visible for the measured HD and, in contrast, even shows increasingly good performance for the HX segmentation quality with growing U^2-Net model capacity. The opposite is, in contrast, observed for the CWS segmentation, i.e., the AUPRC and DSC scores approximately remain high with using the more capacitive model variants. This is presumably related to the lesion's appearance, i.e., CWS are rather large, whitish homogenous areas of the lesion, and small pixel variations might have a smaller effect on the overall AUPRC and DSC performance. Conversely, the HD is very high and shows a wide spread — presumably being caused by the strong penalty when no lesion is present and the model predicts the of a CWS or vice versa.

Regarding the overall minimum HD achieved when comparing the different lesions across all model variants, the HX segmentation performs best ($< 35\,\text{px}$), followed by the MA ($< 95\,\text{px}$), HE ($< 130\,\text{px}$), and, finally, the CWS ($< 195\,\text{px}$) segmentation performance, showing that the former are detected best on instance level, whereas more false positives or negatives occur for the other lesions. This opposes the AUPRC performance observed for the MA and HE segmentation, which shows the latter to outperform

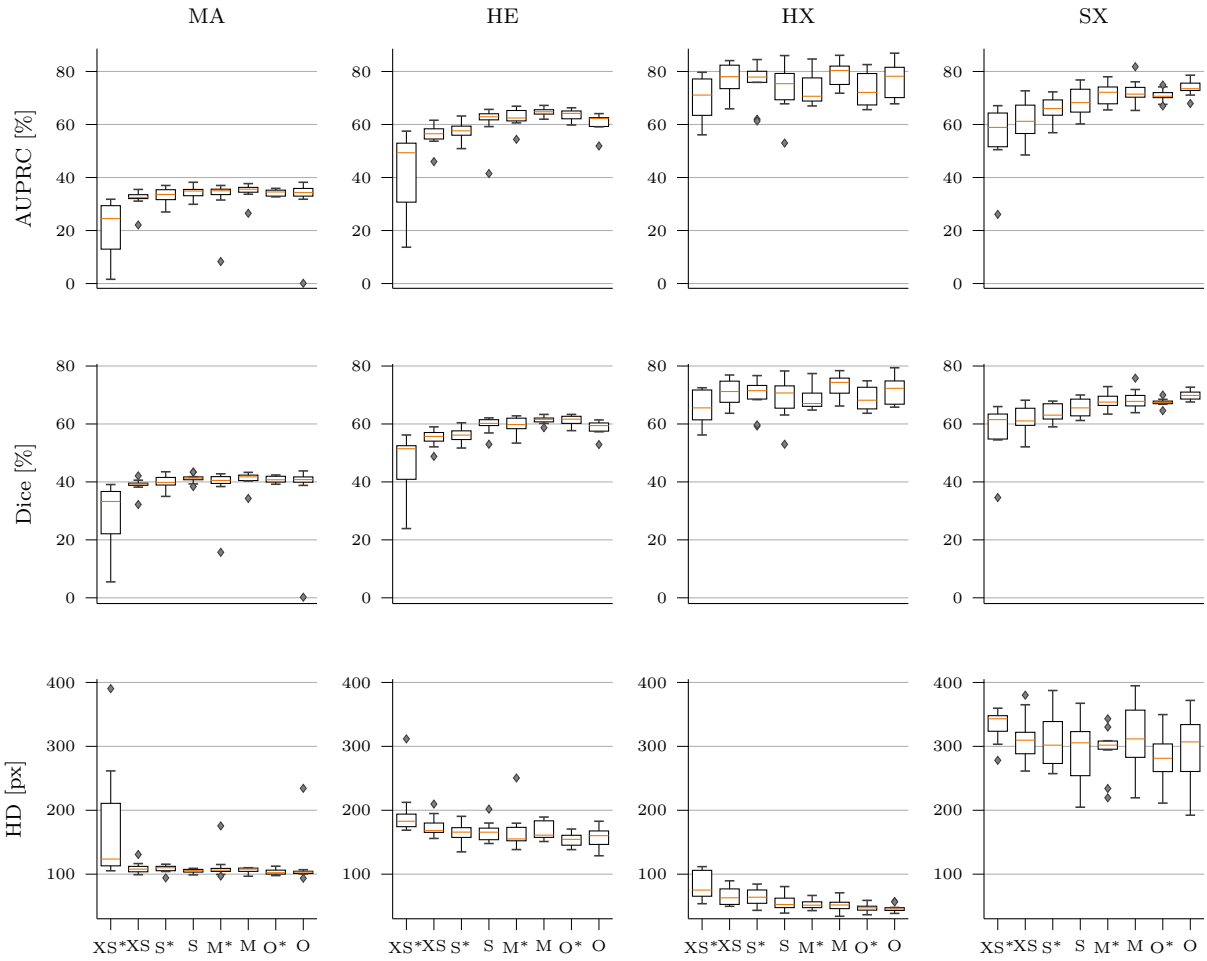


FIGURE 4.4: Boxplots of the AUPRC, DSC, and HD performance of the scaled U^2-Nets and their respective versions using depthwise separable convolution (marked by *) over ten training runs for each lesion. Sorted by model size in ascending order from left to right.

the former. These results underline the high sensitivity of the AUPRC and DSC score to small pixel-level variations, which conversely do not affect the lesion detection performance on the instance-level that is better captured by the HD. Evaluating the detection performance on the instance level, i.e., the sensitivity and specificity for each lesion, in more detail would be of great interest for future work.

Interestingly, the findings for the MA segmentation performance show the smallest deviation across all applied metrics despite the very small lesion size showing a great consensus across the individual trials and regular U^2-Net variants. That is, the segmentation of MAs does not benefit

as much from the increased model capacity, which results in the median AUPRC of the U^2-Net-XS being only moderately lower than that of the U^2-Net-O ($-5.2\,\%$) in contrast to the CWS segmentation performance that is significantly stronger impaired ($-16.7\,\%$).

4.3.2 Performance analysis of the U^2-Net-DC

effect of depthwise separable convolutions on cost

Recalling Table 4.2, the decrease of the computational cost introduced by using DCs has the greatest benefit for the larger model variants, particularly for the original sized U^2-Net-O, and significantly diminishes for the smaller models. This is consistent with the fact that decoupling of depth- and spatial- convolutions by using depthwise separable convolutions has a greater effect on convolutions with large feature depths. Exemplarily, using DC only marginally decreases the model size as well as the required MAC operations and even increases the inference frame rate comparing the U^2-Net-XS to U^2-Net-DC-XS.

performance U^2-Net-DC

As observable from the findings presented in Table 4.3, the performance of the DC-based U^2-Nets is — based on the reduced model expressiveness — slightly worse compared to their respective U^2-Net variants that make use of regular, full-capacity convolutions. Moreover, solely the U^2-Net-DC-S outperforms the next smaller, regular U^2-Net variant, i.e., the U^2-Net-XS. Moreover, the U^2-Net-DC-XS fails to provide a reliable segmentation performance exhibiting a high spread across the individual trials throughout the applied metrics as visible from Figure 4.4 — most predominantly for MA and HE segmentation. This could be related to a borderline sufficient model capacity that renders capturing essential features required for the segmentation task difficult. However, a few trials converged well leading to a considerable segmentation performance, i.e., the maximum AUPRC observed for the MA segmentation of the U^2-Net-DC-XS equals 31.8, which is comparably close to the median performance of the U^2-Net-XS.

Nonetheless, the observed results suggest that the reduced computational complexity by using DCs does in general not compensate for the observed loss of performance. This — at least to some extent — can be attributed to the fact that the usage of DCs is not fully optimized using current computing accelerators [159]. That is, the additional kernel calls introduced by the point-wise convolutions exceed the runtime of a normal convolution when dealing with a large number of small filters, which results in a lower frame

[159] Tan and Le, "EfficientNetV2: smaller models and faster training" (2021)

rate compared to their counterparts without DCs. However, the potential of lowering computational cost by DCs could be better exploited with more optimized toolboxes [160] that might result in an improved performance-cost tradeoff.

[160] Lu *et al.*, "Optimizing depthwise separable convolution operations on GPUs" (2022)

4.3.3 Performance analysis of the multi-task training

The results for the dual- and multi-task training provided in Table 4.4 show the former to overall yield a benefit compared to the single-task mode. In particular, the MA and HE segmentation and the smaller model variants benefit from the parameter sharing, leading to superior performance compared to the single-task model training. That is, the median MA and HE AUPRC performance are improved by about 6.7 % and 1.5 %, respectively, when using the U^2-Net-M in dual instead of single-task configuration. This suggests that the shared feature representation helps the U^2-Net to better discriminate between HEs and MAs, which is a hard task as both lesions can appear as small red dots within the color fundus image.

dual-task training (MA & HE)

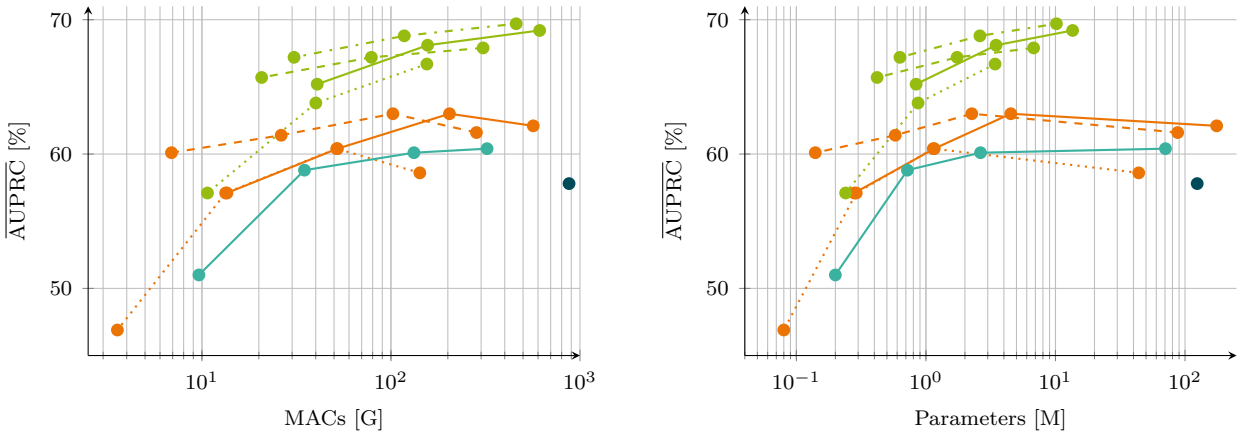
Interestingly, this observation does not apply to the segmentation performance of the HXs and CWSs, i.e., the U^2-Net shows to benefit less from a shared feature representation. In particular, in case the single-task model has enough capacity, i.e., for the U^2-Net-O, the parameter sharing even hurts the quality of the segmentation w.r.t. the AUPRC. This might be caused by the difference between both lesions being discriminative enough to distinguish HXs and CWSs more easily from one another. That is, although both lesions appear as whitish lesions in the retina, the HX are typically smaller and sharply delineated whereas the latter appear as rather large and more diffuse lesions.

dual-task training (HX & CWS)

Similarly, the results show the multi-task models to visibly suffer from the hard parameter sharing across all four tasks — even for the model with the highest capacity, i.e., the U^2-Net-O. That is, mostly the HE and HXs segmentation performance is impaired whereby the results for the other two lesions, i.e., the CWSs and MAs, in general, outperform at least the single-task training setting. From these findings, exploiting a mixture of single- (HX | CWS) and dual-task (MA & HE) training promises to provide the best performance.

multi-task training

Figure 4.5 summarizes the results presented in the Tables 4.2 to 4.4 by depicting the average median AUPRC performance across all lesions in



(a) (b)

FIGURE 4.5: Comparison of the U-Net (●), the differently scaled U^2-Nets (—), the U^2-Net-DC models (—), as well as the DE-based U^2-Nets (—) deployed in single-task- (solid —), dual-task- (dashed - -), multi-task-(dotted ···), and mixed single/dual-task mode (dash-dotted -·) w.r.t. the average median AUPRC ($\overline{\text{AUPRC}}$) across all lesions in dependence to (a) the total computational cost in MACs and (b) the total number of parameters.

TABLE 4.4: Test results on the IDRiD dataset for the different U^2-Net model variants comparing single-, dual-, and multi-task segmentation performance given as median ($\mu_{1/2}$) AUPRC [%] across the individual training runs per lesion with the IQR enclosed in brackets and the overall average median AUPRC ($\overline{\text{AUPRC}}$) [%] across all four lesions.

Model	Task	MA		HE		HX		CWS		$\overline{\text{AUPRC}}$
		$\mu_{1/2}$	IQR	$\mu_{1/2}$	IQR	$\mu_{1/2}$	IQR	$\mu_{1/2}$	IQR	
U^2-Net-O	ST	34.3	(2.9)	62.2	(3.3)	78.2	(11.4)	**73.5**	(2.8)	62.1
	DT	37.9	(3.5)	63.8	(3.7)	72.2	(5.7)	72.3	(3.6)	61.6
	MT	37.7	(1.1)	58.1	(2.6)	70.0	(10.3)	71.0	(3.0)	58.6
U^2-Net-M	ST	35.6	(1.8)	64.8	(1.7)	**80.3**	(6.9)	71.5	(3.7)	**63.0**
	DT	**38.0**	(2.1)	**65.8**	(3.5)	74.7	(6.0)	73.4	(4.8)	**63.0**
	MT	37.5	(1.7)	60.5	(3.4)	73.5	(14.3)	72.9	(6.3)	60.4
U^2-Net-S	ST	34.9	(2.4)	62.9	(2.3)	75.4	(9.9)	68.2	(8.6)	60.4
	DT	37.0	(0.8)	63.2	(2.5)	75.8	(5.8)	69.8	(2.6)	61.4
	MT	35.6	(2.4)	51.6	(3.8)	72.2	(10.6)	70.5	(2.0)	57.1
U^2-Net-XS	ST	32.5	(1.5)	56.5	(3.9)	78.0	(8.8)	61.2	(10.7)	57.1
	DT	37.7	(0.5)	62.6	(4.1)	72.6	(9.1)	67.5	(5.9)	60.1
	MT	28.8	(2.4)	26.0	(13.2)	71.7	(8.8)	63.1	(9.0)	46.9

dependency to the computational cost (MAC) and the number of model parameters. In addition, the performance of the deep ensemble models, i.e., the U^2-Net-M†, U^2-Net-S†, and U^2-Net-XS†, deployed in single-, dual-, multi-, and mixed-task configurations are shown within the plot.

From this, the U^2-Net can be observed to provide a substantial performance gain compared to the U-Net while significantly reducing the computational cost and model parameters. Considering non-ensemble methods, the U^2-Net-S and U^2-Net-M used in dual-task mode show to provide the best trade-off between performance and computational efficiency. However, in particular, the small deep ensembles provide a significant increase

in performance and still a significant reduction of the computational cost and required number of model parameters compared to the U^2-Net-O and U-Net. As such, the findings align with the literature, demonstrating that model averaging through the use of deep ensembles effectively improves model performance by eliminating false positive and negative predictions.

mixed-task training

Overall, the U^2-Net-XS† model in mixed-task configuration is observed to provide a significant relative improvement by approximately 9.4 % w.r.t. the average AUPRC while requiring a similar computational budget as the U^2-Net-S in dual-task mode. Depending on the computational budget, by using either the U^2-Net-S† or the U^2-Net-M† a further increase in segmentation performance by 9.2 % and 10.6 % compared to the best performing single-model instance — the U^2-Net-M in dual-task mode — can be achieved at the expense of an increased computational cost. That is, given parallel execution of the individual model instances, a memory usage of 1.33 GB (U^2-Net-XS†), 2.61 GB (U^2-Net-S†), and 5.21 GB (U^2-Net-M†) is observed. As a result, the U^2-Net-M† would not be executable on an Nvidia Jetson Nano. Nonetheless, with sequential execution or larger but more expensive Nvidia Jetson variants also deploying the U^2-Net-M† could be feasible. Moreover, further optimization of the model's memory requirement and computational complexity by using soft or partial hard weight sharing across the encoders similar to the model architecture designed by [139] could facilitate model training and deployment.

4.3.4 Literature comparison

A comparison of this chapter's findings to the related literature is provided in Table 4.5 showing the performance of the proposed methods to overall be on par with the state-of-the-art. That is, despite not reaching the performance for MA segmentation, the U^2-Net ensembles are on par for the HX and the HE segmentation. Moreover, they significantly outperform the quality of the CWS segmentation compared to the literature.

image resolution

The reduced MA segmentation quality is primarily expected to be caused by the comparable low-resolution input image resulting in both a loss of essential information, as MA lesions are typically only a few pixels wide. In comparison, the L-Seg model proposed by Guo *et al.* [147] is trained and evaluated on high-resolution input images with (1440 × 960) pixels and with this significantly outperforms the U^2-Net for MA segmentation.

TABLE 4.5: Comparison of the deep ensemble-based U^2-Net's lesion specific and average median AUPRC ($\overline{\text{AUPRC}}$) [%] performance in mixed-task mode to results from the literature.

Model	MA	HE	HX	CWS	$\overline{\text{AUPRC}}$
U^2-Net-M†	42.2	69.7	86.2	**80.5**	69.7
U^2-Net-S†	42.1	68.7	85.0	79.5	68.8
U^2-Net-XS†	41.7	67.7	85.2	74.3	67.2
Zhou et al. [138]	47.3	65.8	84.5	71.6	67.3
+Semi+Adv	49.6	69.4	88.7	74.1	**70.4**
Yan et al. [145]	**52.5**	**70.3**	**88.9**	67.9	69.9
Guo et al. [147]	46.3	63.7	79.5	71.1	65.2

Similarly, Zhou *et al.* [138] deploy model optimization and inference with fundus images at (640×640) resolution. Moreover, Yan *et al.* [145] proposed a Dual-U-Net architecture fusing a global U-Net that is processing the complete, downsampled retinal images and a local U-Net that operates on high-resolution image patches, which they observe to significantly boost the segmentation performance of both the smallest lesions, i.e., the MAs and HXs. This equally aligns with that the best-performing methods for MA segmentation within the original IDRiD competition [161] either exploited very high-resolution input images, cascaded CNNs for mask-refinement, or sliding window approaches.

[161] Porwal *et al.*, "IDRiD: diabetic retinopathy – segmentation and grading challenge" (2020)

U^2-Net depth and multi-scale features

Another cause for the comparably low MA segmentation performance might be the deep network structure of U^2-Net, as the segmentation of the MAs is expected to primarily rely on local image information. Conversely, the large performance improvements for both the HE and the CWS segmentation, which are achieved by using the U^2-Net compared to the U-Net, suggest that the additional contextual information of the multi-scale features extracted by the RSU blocks is utterly important for a high-quality HE and CWS segmentations. This is in line with the observation made by Yan *et al.*, who observe the shallower architectures to yield better performance w.r.t. the MA segmentation and that the larger and more homogenous lesions, i.e., HEs and CWSs, are segmented with the highest performance using only the global U-Net.

lightweight segmentation

[162] Li *et al.*, "Diagnostic assessment of deep learning algorithms for diabetic retinopathy screening" (2019)

Comparing the proposed results to the LWENet proposed by Guo *et al.* [143], their pretrained model on the DDR [162] dataset outperforms the U^2-Net-M with regard to the DSC, i.e., DSC = 78.2 % vs. DSC = 74.3 %, which has a similar number of parameters. However, without applying pretraining as in this analysis, the LWENet performs significantly worse (DSC = 69.7 %). Moreover, the U^2-Net-XS† is more than nine times smaller, only slightly slower w.r.t. inference at the identical image resolution, i.e., 11.1 fps (Nvidia GTX 1080Ti) vs. 10.5 fps (Nvidia RTX 2080Ti)

at 1440×960 pixels, and achieves similar performance, i.e., DSC = 77.9 %, without exploiting transfer learning.

The Xception-U-Net proposed by Zhou *et al.* when being trained in a simple fully supervised manner provides a strong baseline, which is on par with the U^2-Net-XS ensemble. Regarding the computational cost, it has a higher number of parameters and a similar computational complexity w.r.t. the required MAC operations compared to the U^2-Net-M and U^2-Net-S in multi-task mode, i.e., 2.34 M parameters and 7.8 GMACs operations, respectively. Despite their reported performance being higher than that of the single-instance U^2-Nets in multi-task mode, training and evaluating the Xception-U-Net in multi-task mode within this work's experimental setup yields a significantly lower average median AUPRC performance of 56.5 %, which — as previously discussed — is likely to be primarily caused by the reduced input image resolution. This, in turn, is outperformed by the U^2-Net-S in multi-task mode and by the U^2-Net-XS ensemble, whereby the latter comprises about 73 % fewer parameters showing the lightweight U^2-Net architecture design to add a performance gain over the Xception-U-Net architecture, albeit at a moderately increased cost w.r.t. the required MAC operations.

architecture improvements

Further optimization of the U^2-Net architecture by exploiting more refined mobile building blocks could not only improve the reduction of the computational complexity but also the U^2-Net's performance. That is, simply replacing the regular convolutions with DCs as in this work decreased the computational cost but also disproportionately impaired the segmentation performance. In contrast, Zhou *et al.* observe the Xception-blocks, which exploit both DCs and residual connections with additional a (1×1)-convolutions, to improve the performance compared to the using regular convolutions. Accordingly, adopting Xception-based convolutions within the U^2-Net architecture instead of solely exploiting DCs could provide an additional benefit to the performance. Moreover, exploiting spatial attention modules such as squeeze-and-excitation (SE) [163] layers or mobile inverted bottleneck convolution (MBC) [151] blocks are expected to further improve the U^2-Net's efficiency. As a result, this would allow retaining a higher input image resolution and feature depth within the U^2-Net's bottleneck RSU blocks and be expected to significantly improve the segmentation performance of the U^2-Nets. Finally, methods such as weight quantization and model pruning [164] or ensemble distilling [165] could enable more effi-

[163] Hu *et al.*, "Squeeze-and-excitation networks" (2020)

[164] Han *et al.*, "Deep compression: compressing deep neural networks with pruning, trained quantization and huffman coding" (2016)

[165] Hinton *et al.*, "Distilling the knowledge in a neural network" (2015)

cient deployment of the U^2-Net to edge devices potentially allowing for the larger ensemble variants.

transfer- and semi-supervised learning

In addition, in future work, the ability of the U^2-Net to generalize to other data should be evaluated. Nonetheless, the availability of pixel-level annotated image data remains a major challenge for biomedical image segmentation in order to further improve robustness and generalization. One solution to overcome this limitation could be the exploitation of image-level annotated retinal fundus datasets by deploying sophisticated training strategies such as adversarial and semi-supervised training schemes as utilized by [138]. Such an approach could easily be adapted, in case of fusing the image segmentation and classification into a single, end-to-end trainable model as will be discussed in more detail in section 6.4.

4.4 Conclusion and outlook

For a real-world, clinical application, a DR screening system ideally explains the derived prediction by, e.g., highlighting the disease-related pathologic areas in the processed retinal fundus image to allow the supervisor to verify and challenge the prediction. Furthermore, to allow fast and seamless usage on mobile hardware these systems should additionally be designed in a lightweight manner. To this end, this chapter analyzed the potential of the U^2-Net — a U-Net-based architecture that exploits efficient multi-scale feature extraction modules called RSU blocks — to be used as a basic building block in a holistic DR screening pipeline that complies with these requirements while searching for a reasonable tradeoff between the computational cost of the model and retaining a high predictive performance.

The presented results suggest that (a) the U^2-Net is an overall suitable architecture for biomedical image segmentation, (b) the model architecture shows good performance that is on par with and in particular for CWS segmentation shows the potential to outperform current state-of-the-art results even when the model is trained on a small dataset from scratch, (c) exploiting a mixture of single- and dual-task training provides overall best segmentation performance, and (d) the downscaled U^2-Nets propose to allow seamless edge device implementation, as they show very promising results for DR-related lesion segmentation while having very few parameters as well as a reasonable computational cost. In particular, the U^2-Net-XS† is observed to yield a good tradeoff between model complexity

as well as computational cost and lesion segmentation performance. As discussed above, further optimization of the base architecture could improve the performance upon the promising results of this analysis.

Nevertheless, particularly improving the MA segmentation performance could play an important role in providing a highly sensitive early detection of DR onset, and further optimization of the model architecture could additionally improve the Pareto optimum of the performance-cost tradeoff. Moreover, to develop a truly holistic DR grading pipeline that is able to explain particularly more severe DR levels, the corresponding DR-related lesions, such as IRMAs and NVs, would have to be taken into account within the segmentation model.

5 | Transparency through uncertainty-awareness

As outlined in the introduction, this chapter analyzes the suitability of the DKL framework to improve NNs for their lack of knowledge by means of evidence-based uncertainty quantification within the DR screening setting. First, a short motivation summarizing the key aspects of the practical relevance of uncertainty calibration and awareness including an overview of the related work is given in section 5.1, followed by a description of the adapted methodology and the experimental setup of this analysis provided in section 5.2. Finally, the results of the experiments comprising both ID and OOD data are presented in section 5.3. A discussion and summary of this chapter are given in sections 5.4 and 5.5. A preliminary analysis of the general feasibility and performance of the stochastic variational deep kernel learning (SVDKL) framework was published in a conference paper [166] by the author of this thesis. Parts of this chapter were previously published in a peer-reviewed journal article [45], first authored by the author of this thesis. He conceived and performed the methodology, experiments, and analyses, and wrote the manuscript. The co-authors of the publication provided guidance and feedback on the methodology, experiments, and manuscript.

[166] Siebert *et al.*, "Stochastic variational deep kernel learning based diabetic retinopathy severity grading" (2022)

[45] Siebert *et al.*, "Uncertainty analysis of deep kernel learning methods on diabetic retinopathy grading" (2023)

5.1 Motivation and related work

With the state-of-the-art performance of DNNs, a promising solution to shift the DR screening and grading to primary care, i.e., using the regular appointments to check the glycemic levels of patients suffering from DM for verifying their vision-health status. However, in contrast to humans who tend to perform well under uncertainty, deterministic DNNs usually extrapolate widely and lack calibrated uncertainties [11], [33], [73]. Hence, to minimize erroneous predictions, mitigate the risk of not receiving the

[11] Petersen *et al.*, "Responsible and regulatory conform machine learning for medicine: a survey of challenges and solutions" (2022)

[33] Jospin *et al.*, "Hands-on bayesian neural networks—a tutorial for deep learning users" (2022)

[73] Guo *et al.*, "On calibration of modern neural networks" (2017)

required intervention, and lower the likelihood of missing pathologies or diseases other than DR that could have been detected by specialists, calibrated uncertainty information is essential to allow the supervisor of the model *to reject highly uncertain samples* and refer patients to specialists for further inspection.

To address these challenges, recent related research [167]–[170] compare the use of CNNs and several BNN methods, e.g., MCD [103], DEs [102], and other approximations like VI, for the task of either detecting or grading DR. The authors of these studies highlight the feasibility of Bayesian methods and their benefit on the uncertainty calibration to be informative for model failures: Band *et al.* [167] analyze uncertainty calibration for the binary classification of referable DR comparing mean field variational inference (MFVI), MCD, and DE, but do neither include DKL nor consider the DR severity grading task. Similarly, Jaskari *et al.* [168] evaluate DE, MCD, MFVI, generalized variational inference (GVI), and radial BNNs on the task of both referable DR detection and severity grading. Proposing DR|Graduate, Aruajo *et al.* [169] train a CNN on the multiclass task of DR severity grading with an additional classification head to estimate a predictive uncertainty σ that is used to place a Gaussian distribution with σ as the standard deviation on the predicted grade. As GPs typically provide well-calibrated uncertainty estimates but do lack the feature extraction abilities of CNNs, Leibig *et al.* [170] analyze the combined usage of these two, but find their MCD-based BNN to outperform a GP being trained on the latent features of a fixed CNN within the decision referral setting, i.e., the binary detection of referable (>1) and any DR (>0), omitting the severity grading.

Overall, MCD- and DE-based approaches are observed to frequently provide good calibration, whereas more sophisticated VI approaches sometimes even perform worse than regular CNNs [167], [170]. However, both of these simple baselines have shortcomings, potentially causing inconsistent uncertainty calibration under distribution shift [95], [104], [167]. Acknowledging that a GP theoretically would be a promising alternative, Leibig *et al.*, however, find MCD to outperform the GP-based approach. They assume this to be caused by the loss of full stack information when training the CNN and GP independently, which was necessary due to the bad scalability of GPs to large datasets and high-dimensional inputs.

Based on these observations, the goal of this chapter is to investigate

[167] Band *et al.*, "Benchmarking Bayesian deep learning on diabetic retinopathy detection tasks" (2021)

[168] Jaskari *et al.*, "Uncertainty-aware deep learning methods for robust diabetic retinopathy classification" (2022)

[169] Araújo *et al.*, "DR|GRADUATE: uncertainty-aware deep learning-based diabetic retinopathy grading in eye fundus images" (2020)

[170] Leibig *et al.*, "Leveraging uncertainty information from deep neural networks for disease detection" (2017)

[103] Gal and Ghahramani, "Dropout as a bayesian approximation: representing model uncertainty in deep learning" (2016)

[102] Lakshminarayanan *et al.*, "Simple and scalable predictive uncertainty estimation using deep ensembles" (2017)

[95] Ovadia *et al.*, "Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift" (2019)

[104] Wilson and Izmailov, "Bayesian deep learning and a probabilistic perspective of generalization" (2020)

whether the DKL [42] framework, i.e., a hybrid deep learning model combining these two methodologies in an end-to-end trainable architecture, can exploit the uncertainty quantification abilities of GPs within the uncertainty-informed DR classification screening. In particular, this chapter examines whether and how SVDKL [171], which allows the application of the DKL framework to be applied to classification, can be used to improve the quality of uncertainty estimates while maintaining high diagnostic performance for the DR screening and grading. This allows referring both patients with emerging or worsening DR and those with uncertain diagnoses to experts. Thereby, the SVDKL framework's performance and quality of uncertainty estimates are compared to those being provided by standard approximate Bayesian CNNs. The remainder of this chapter provides a comprehensive evaluation of SVDKL and recently proposed variants [44], [172]–[174] for the task of DR severity (sDR) grading and referable DR (rDR) detection that is compared to the most commonly applied approximative BNN methods exploiting MCD and DE on both in-distribution (ID) and out-of-distribution (OOD) data while disentangling aleatoric und epistemic uncertainty estimates.

[42] Wilson *et al.*, "Deep kernel learning" (2016)

[171] Wilson *et al.*, "Stochastic variational deep kernel learning" (2016)

[44] Tran *et al.*, "Calibrating deep convolutional gaussian processes" (2019)

[172] Ober *et al.*, "The promises and pitfalls of deep kernel learning" (2021)

[173] Liu *et al.*, "A simple approach to improve single-model deep uncertainty via distance-awareness" (2023)

[174] Amersfoort *et al.*, "On feature collapse and deep kernel learning for single forward pass uncertainty" (2021)

5.2 Methods

In the following, the methodology and experimental setup of this chapter will be introduced. First, the adopted study data and image preprocessing used for training and evaluation will be introduced in sections 5.2.1 and 5.2.2. This will be followed by a detailed introduction of GPs and the DKL framework including the adaption of this method for classification tasks and some extensions to improve the uncertainty calibration of the method given in sections 5.2.3 and 5.2.4. The subsequent sections 5.2.6 to 5.2.10 will provide a detailed description of the model implementation, training and evaluation protocol, and the statistical methods used to evaluate the results for statistically significant improvements.

5.2.1 Study data

The datasets used in this work for model training and validation are grouped into ID and OOD datasets according to

$$\mathcal{D}_{\text{ID}} = \{\text{EyePACS}, \text{IDRiD}\} \tag{5.1}$$

and

$$\mathcal{D}_{\text{OOD}} = \{\text{RFMID}, \text{SIIM-ISIC}\}\,. \tag{5.2}$$

[140] Dugas *et al.*, "Diabetic retinopathy detection" (2015)

[9]The data is publically available and was hosted by Kaggle in the context of the diabetic retinopathy detection challenge in 2015. `https://www.kaggle.com/competitions/diabetic-retinopathy-detection` (Last accessed: 11/14/2023)

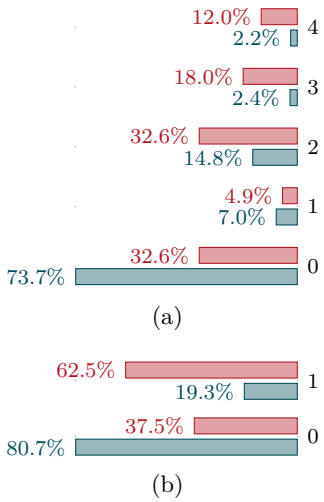


FIGURE 5.1: Distribution of (a) DR severity and (b) referable DR classes of the EyePACS (■) and IDRiD (■) dataset.

[38] Wilkinson *et al.*, "Proposed international clinical diabetic retinopathy and diabetic macular edema disease severity scales" (2003)

[34] Porwal *et al.*, "Indian diabetic retinopathy image dataset (IDRiD): a database for diabetic retinopathy screening research" (2018)

EyePACS The publicly available EyePACS [140] database[9] is used for model training and ID testing. The dataset is sponsored by the California Healthcare Foundation and the EyePACS platform. It contains high-resolution retinal fundus images captured from both direct and indirect ophthalmoscopic examinations with high variation in quality, e.g., concerning lighting, contrast, and sharpness. The image data is accompanied by DR severity (sDR) grade annotations derived from a medical expert. Aligning the ICDR grading scheme [38] introduced in section 3.2, the images are classified into the four classes no (0), mild (1), moderate (2), severe (3), and proliferative DR (4). Equivalently, referable DR (rDR) is defined as having more than mild DR, i.e., rDR class comprises the severity levels 2–4. The distributions of the sDR and rDR classes comprised in the dataset are strongly imbalanced, as displayed in Figure 5.1. In total, the dataset comprises 35 126 training images, 10 906 validation images (i.e., the *public* competition test data), and 42 670 test images (i.e., the *private* competition holdout data).

IDRiD For model validation, the IDRiD [34] dataset is deployed. This database contains 516 high-quality, high-resolution retinal fundus images captured at the Eye Clinic in Nanda, India. The dataset provides medical expert-validated grades of sDR following the ICDR scale. Although sharing the same task, the class distribution is different compared to the EyePACS dataset as displayed in Figure 5.1. Furthermore, as the data is collected in India and was graded by different experts, it is expected to impose a distribution shift. Nonetheless, due to the similarity of the IDRiD to the EyePACS dataset, it will be referred to as ID data in the remainder of this thesis. In addition to the sDR grades, annotations for DME risk are provided. The dataset is published with a predefined training/test split with ratio 413/103. The training images of the IDRiD dataset are *not* used for model training. Instead, the joint training and test data is used to estimate the generalization properties of the analyzed methods.

[175] Pachade *et al.*, "Retinal fundus multi-disease image dataset (RFMiD): a dataset for multi-disease detection research" (2021)

[10]The data is publically available and was hosted in the context of the Retinal Image Analysis for multi-Disease Detection Challenge workshop at the IEEE International Symposium on Biomedical Imaging (ISBI-2021). `https://dx.doi.org/10.21227/s3g7-st65` (Last accessed: 06/01/2024)

RFMID Similar to the IDRiD dataset, the RFMID [175] database[10] contains retinal fundus images. However, it is only loosely related to the DR

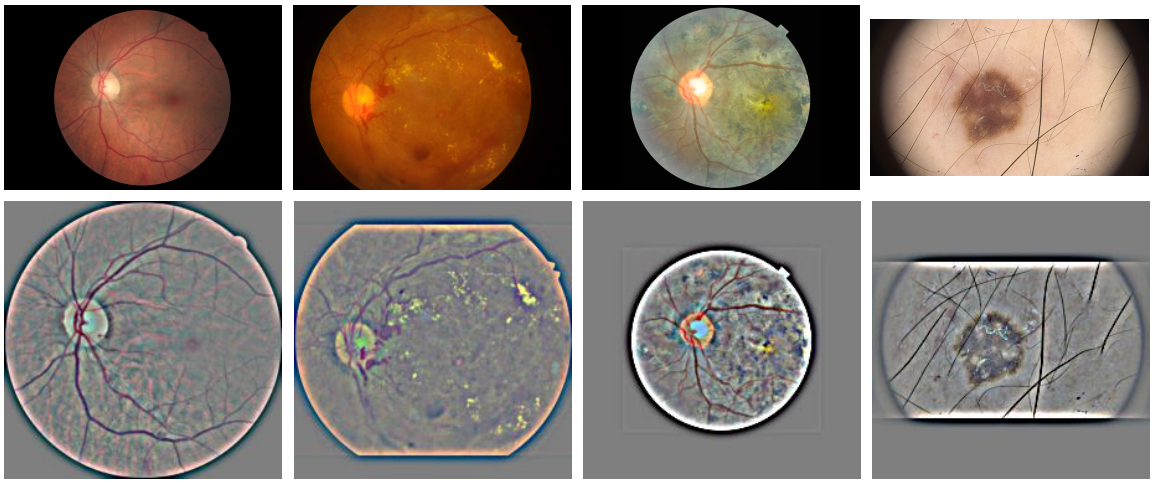


FIGURE 5.2: Original images from the EyePACS [140], IDRiD [34], RFMID [175], and SIIM-ISIC [176] dataset (top row) and the corresponding preprocessed images (bottom row).

grading task, i.e., it comprises a multitude of other eye diseases in addition to DR and does not contain annotations for the DR severity. The dataset can thus be viewed as a mixture of both ID and OOD data and will therefore be referred to as both task-related OOD and near-ID data due to sharing the same input domain and the partial task overlap. However, due to being primarily used to validate the method's uncertainty-based OOD detection abilities, the dataset is categorized into the group of OOD datasets.

SIIM-ISIC As a fully task-unrelated, OOD dataset, the SIIM-ISIC [176] dataset[11] is used. It is a medical dataset of annotated dermoscopic images for skin lesion classification and is in this work used to validate the OOD detection abilities in addition to the RFMID data.

[176] Rotemberg *et al.*, "A patient-centric dataset of images and metadata for identifying melanomas using clinical context" (2021)

[11] The data is publically available and was hosted by Kaggle in the context of the 2020 SIIM-ISIC Melanoma Classification challenge. `https://www.kaggle.com/competitions/siim-isic-melanoma-classification` (Last accessed: 06/01/2024)

5.2.2 Image preprocessing and augmentation

For preprocessing, all image samples of the datasets comprised within $\mathcal{D}_{\text{ID}}$ are cropped to a square containing the visible retinal disc, whereas out-of-distribution images are padded to a square to retain their aspect ratio. The minimum viable rectangle is obtained by using a U^2-Net-XS as introduced in the previous chapter trained for predicting the extent of the visible retinal disc. Subsequently, all images of the comprised datasets in both $\mathcal{D}_{\text{ID}}$ and $\mathcal{D}_{\text{OOD}}$ are resampled to $(I_h, I_w) = (224 \times 224)$ pixels and the local color mean is removed using a Gaussian filter per image channel, following the implementation of Ben Graham [177], the winner of the Kaggle

[177] Graham, "Kaggle diabetic retinopathy detection competition report" (2015)

DR Challenge [140]. A blur constant of $b = 30$ is used in this experiment, resulting in using a kernel with the standard deviation $\sigma = \frac{I_{h/w}}{2b} \approx 3.73$. An example image per dataset prior to and after preprocessing is given in Figure 5.2. All images are standardized to the statistics of the EyePACS training data, i.e., zero mean and unit variance, prior to feeding them to the networks.

During model training, online data augmentation is applied. In detail, this comprises random horizontal and vertical flipping with probability $p_{\text{fl}} = 0.5$, and random image rotation within $\pm 55°$ is applied, as well as random image translation and scaling within $\pm 20\,\%$ and $\pm 10\,\%$ w.r.t. the maximum image height I_h and width I_w.

5.2.3 Gaussian processes

Gaussian processes (GPs) are a popular ML tool that generalizes well due to its nonparametric nature and provides predictions with calibrated uncertainty estimates [98, p. 675]. A GP describes a multivariate Gaussian distribution over a finite subset of function values of an unknown function $g : \mathcal{X} \to \mathcal{Y}$

[98] Murphy, *Probabilistic Machine Learning: Advanced Topics* (2023)

$$g(\mathbf{x}) \sim \mathcal{GP}\left(m(\mathbf{x}), k_{\boldsymbol{\theta}_{\text{GP}}}(\mathbf{x}, \mathbf{x}')\right). \tag{5.3}$$

covariance function

It is fully specified by the mean $m(\mathbf{x})$ and covariance $k(\mathbf{x}, \mathbf{x}')$ function [178, p. 13]. The latter is defined by a number of hyperparameters $\boldsymbol{\theta}_{\text{GP}}$ and incorporates prior knowledge about the observed data by defining the dependency of the function values evaluated at two input positions $\mathbf{x}$ and $\mathbf{x}'$. A common choice for k is the squared exponential kernel, which is also known as RBF kernel and defined as

[178] Rasmussen and Williams, *Gaussian processes for machine learning* (2006)

$$k_{\text{RBF}, \boldsymbol{\theta}_{\text{GP}}}(\mathbf{x}, \mathbf{x}') = \sigma_k^2 \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\lambda^2}\right), \tag{5.4}$$

similar to the definition in (2.7). Thereby, the kernel hyperparameters $\boldsymbol{\theta}_{\text{GP}} = \{\sigma_k^2, \lambda\}$ define the overall variance to the mean and the length scale of the kernel, which determines the distance over which the function values are correlated with another [178, pp. 83–84]. Whereas not necessarily required, typically a zero mean function $m(\mathbf{x}) = 0$ is chosen. This does, however, not limit the posterior to mean to be zero [178, p. 27].

Gaussian process regression

Considering a regression task $y = g(\mathbf{x}) + \epsilon$ with measurement noise $\epsilon \sim \mathcal{N}(0, \sigma_y^2)$, i.e., assuming a Gaussian likelihood, the GP prior over

the function values $\mathbf{y} = [y_1, y_2, \ldots, y_N]$ at an arbitrary set of input points $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]$ with $\mathbf{x} \in \mathbb{R}^D$ is given by [98, p. 687]

$$\mathbf{y} \sim p(\mathbf{g} \mid \mathbf{X}) = \mathcal{N}(0, K(\mathbf{X}, \mathbf{X}) + \sigma_y^2 \mathbf{I}) \tag{5.5}$$

with the covariance matrix

$$K(\mathbf{X}, \mathbf{X}') = \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}'_1) & k(\mathbf{x}_1, \mathbf{x}'_2) & \ldots & k(\mathbf{x}_1, \mathbf{x}'_m) \\ k(\mathbf{x}_2, \mathbf{x}'_1) & k(\mathbf{x}_2, \mathbf{x}'_2) & \ldots & k(\mathbf{x}_2, \mathbf{x}'_m) \\ \vdots & \vdots & \ddots & \vdots \\ k(\mathbf{x}_n, \mathbf{x}'_1) & k(\mathbf{x}_n, \mathbf{x}'_2) & \ldots & k(\mathbf{x}_n, \mathbf{x}'_m) \end{bmatrix} \tag{5.6}$$

and the vector of latent function values $\mathbf{g} = g(\mathbf{X}) = [g(\mathbf{x}_1), \ldots, g(\mathbf{x}_N)]$. Furthermore, the joint prior distribution of the GP for the samples $\mathbf{X}$ and a set of test samples $\mathbf{X}^*$ is defined as

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{y}^* \end{bmatrix} \sim p(\mathbf{g}, \mathbf{g}^* \mid \mathbf{X}, \mathbf{X}^*) = \mathcal{N}\left(0, \begin{bmatrix} K(\mathbf{X}, \mathbf{X}) + \sigma_y^2 \mathbf{I} & K(\mathbf{X}, \mathbf{X}^*) \\ K(\mathbf{X}^*, \mathbf{X}) & K(\mathbf{X}^*, \mathbf{X}^*) \end{bmatrix}\right). \tag{5.7}$$

By observing the dataset $\mathcal{D} = \{\mathbf{X}, \mathbf{y}\}$ with measurements $\mathbf{y}$ at locations $\mathbf{X}$ and conditioning the joint prior on the observed data, the posterior predictive distribution

$$\mathbf{y}^* \sim p(\mathbf{g}^* \mid \mathbf{X}^*, \mathbf{X}, \mathbf{y}) = \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*) \tag{5.8}$$

can be derived, which has the closed form solution with mean $\boldsymbol{\mu}^*$ and variance $\boldsymbol{\Sigma}^*$ [98, p. 687]

$$\boldsymbol{\mu}^* = K(\mathbf{X}^*, \mathbf{X})[K(\mathbf{X}, \mathbf{X}) + \sigma_y^2 \mathbf{I}]^{-1} \mathbf{y} \tag{5.9}$$

$$\boldsymbol{\Sigma}^* = K(\mathbf{X}^*, \mathbf{X}^*) - K(\mathbf{X}^*, \mathbf{X})[K(\mathbf{X}, \mathbf{X}) + \sigma_y^2 \mathbf{I}]^{-1} K(\mathbf{X}, \mathbf{X}^*). \tag{5.10}$$

That is, predicting with a GP does not only provide a mean estimate, but a full distribution over the function values at the test points based on the evidence from the training samples and the belief in the covariance of the function values expressed by the kernel function. A visual example of the GP's prior and posterior for a 1-dimensional regression task is given in Figure 5.3.

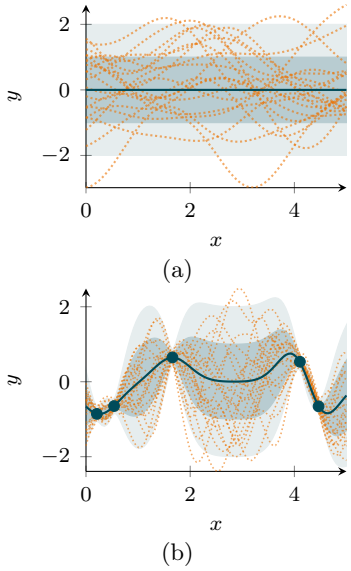


FIGURE 5.3: Samples from $g(x)$ (···) from (a) the GP prior and (b) the posterior after observing the data $\mathcal{D}$ (●) according to (5.5) and (5.8) with using an RBF kernel for a 1-dimensional regression task. The solid line (—) and the shaded area show the prior and posterior mean as well as the single and twofold standard deviation.

advantages of GPs

The major advantages of GPs are both the well-calibrated predictive uncertainty estimates and the *nonparametric* nature of the method. In

contrast to parametric models such as a linear regressor whose flexibility is dependent on the type and amount of selected basis functions, a GP is only sensitive to the selected kernel function with its usually small set of hyperparameters, which can automatically be optimized through the log marginal likelihood of the model on the training data $\mathcal{D}$ [178, p. 19]. This, on the one hand, maximizes the data fit and, on the other hand, penalizes the covariance matrix' complexity favoring smoother functions and, hence, automatically adapts the GP's flexibility to the amount of available data [98, pp. 673, 689]. As a result, GPs typically generalize well.

disadvantages of GPs

The main drawback of GPs is their computational cost. The inversion of the data-dependent covariance matrix using Cholesky decomposition scales with $\mathcal{O}(N^3)$ and — after retrieving K^{-1} once — predicting still has a complexity of $\mathcal{O}(N^2)$, which is considerably limiting the scalability of GPs to large data [42], [178, p. 114]. Another disadvantage of GPs are their limited feature extraction capabilities compared to DNNs: They natively do not work as well on high-dimensional, structured inputs like images [172].

sparse variational GP

To solve the former, i.e., to make GPs scalable to large datasets, methods such as sparse variational GPs (SVGPs) can be used. The main idea of SVGP is to approximate the true GP's posterior by a simpler distribution that is dependent on a smaller number of inducing points rather than the entire dataset in order to reduce the computational cost. Note that the terms indicating the dependency on the inputs such as $\mathbf{X}$ and $\mathbf{X}^*$ are omitted in the following for clarity. In detail, SVGP aims to find a variational posterior distribution

$$q(\mathbf{g}, \mathbf{g}^*, \mathbf{u}) = p(\mathbf{g}, \mathbf{g}^* \mid \mathbf{u})q(\mathbf{u}) \tag{5.11}$$

in order to approximate the true posterior

$$p(\mathbf{g}, \mathbf{g}^*, \mathbf{u} \mid \mathbf{y}) \approx q(\mathbf{g}, \mathbf{g}^*, \mathbf{u}) \tag{5.12}$$

of the joint prior over the latent variables $\mathbf{g}$, $\mathbf{g}^*$, and the inducing variables $\mathbf{u} \in \mathbb{R}^{N'}$, which are indexed by $N' \ll N$ inducing points [98, pp. 702–704]. Thereby, the variational distribution $q(\mathbf{u})$ can be chosen arbitrarily but for simplicity is commonly defined as a normal distribution

$$q(\mathbf{u}) = \mathcal{N}(\boldsymbol{\mu}, \mathbf{C}) \tag{5.13}$$

with mean $\boldsymbol{\mu}$ and covariance $\mathbf{C}$ and $p(\mathbf{g}, \mathbf{g}^* \mid \mathbf{u})$ can be derived in closed form from the GP prior [98, pp. 702–704]. By minimizing the KL divergence

$$\mathcal{L}_{\text{SVGP}}(q) = \text{KL}\left[q(\mathbf{g}, \mathbf{g}^*, \mathbf{u}) \parallel p(\mathbf{g}, \mathbf{g}^*, \mathbf{u} \mid \mathbf{y})\right], \tag{5.14}$$

the variational posterior distribution $q(\mathbf{g}, \mathbf{g}^*, \mathbf{u})$ can be optimized to capture the relevant information of the true posterior [98, p. 703]. As a result, predictions can be derived based on their dependency w.r.t. $\mathbf{u}$ instead of relying on $\mathbf{g}$, which significantly reduces the computational cost. As for the variational inference procedure introduced in (2.55) and (2.57), the above KL divergence is intractable but can be approximated using the ELBO objective

$$\mathcal{L}_{\text{ELBO}}(q) = \int p(\mathbf{g} \mid \mathbf{u}) q(\mathbf{u}) \log p(\mathbf{y} \mid \mathbf{g}) \, d\mathbf{g} \, d\mathbf{u} - \text{KL}\left[q(\mathbf{u}) \parallel p(\mathbf{u})\right]. \tag{5.15}$$

Gaussian process classification

In contrast to GP-regression, the GP likelihood becomes non-Gaussian for classification tasks, which involves squashing the model output through a nonlinear activation function, such as the softmax or sigmoid, to obtain the individual class probabilities [98, pp. 693–695]. As a result, the posterior becomes analytically intractable and an approximation to the non-Gaussian posterior is required. This could for instance be accomplished by using VI approaches such as the SVGP method introduced above, which automatically allows handling non-Gaussian likelihoods [98, p. 702].

5.2.4 Deep kernel learning

Due to the limited feature extraction capabilities of GPs, and as obtaining well-calibrated uncertainty estimates by using BNNs remains difficult, both Wilson *et al.* [42] and Calandra *et al.* [179] propose deep kernel learning (DKL). This is a promising approach to compensate for these disadvantages by combining the feature extraction capabilities of DNNs with the reliable uncertainty representation of GPs. To this end, a GP regressor — being applied to the latent space of a DNN's feature extractor — and the model itself are jointly optimized through the marginal likelihood of the GP by interpreting the DNN's parameters and learned transformation as kernel hyperparameters and function of the GP, respectively.

[179] Calandra *et al.*, "Manifold Gaussian processes for regression" (2016)

stochastic variational deep kernel learning

Later on, Wilson *et al.* [171] propose stochastic variational deep kernel learning (SVDKL) as an extension to the DKL approach for classification

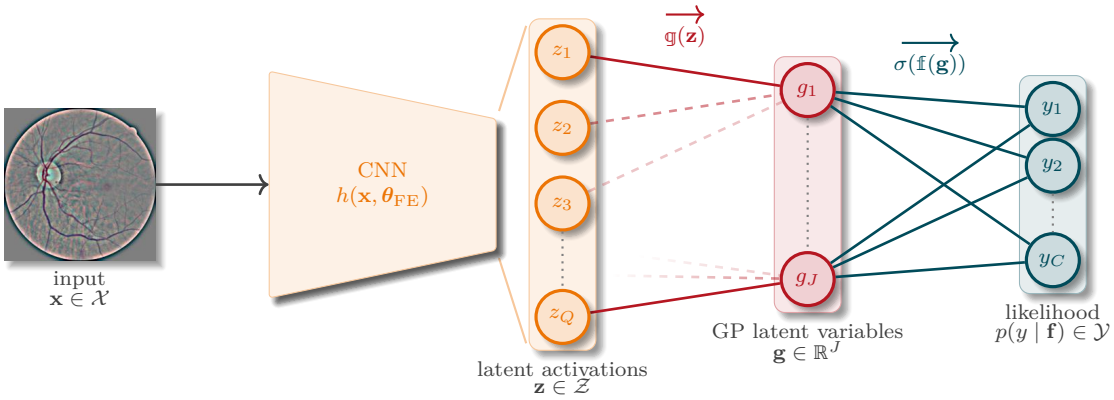


FIGURE 5.4: Schematic diagram of the SVDKL architecture. Note that for $Q = J$, the input to each GP in the additive GP layer is 1-dimensional. Accordingly, for $Q > J$ the input dimension for each GP increases, which is indicated by the dashed red lines. Figure derived upon [47].

tasks. In detail, the hybrid model architecture is designed to apply an additive GP layer

$$\mathbf{f} = \mathbb{f}(\mathbf{z}) = \mathbf{A}\mathbb{g}(\mathbf{z}) \in \mathbb{R}^C \tag{5.16}$$

to the DNN's latent activations $\mathbf{z} = h(\mathbf{x}, \boldsymbol{\theta}_{\text{FE}}) \in \mathbb{R}^Q$. This layer comprises J GPs

$$\mathbf{g} = \mathbb{g}(\mathbf{z}) = [g_1(\mathbf{z}_1), \dots, g_J(\mathbf{z}_J)] \in \mathbb{R}^J \tag{5.17}$$

$$g_j(\mathbf{z}) \sim \mathcal{GP}(0, k_j), \ \forall j = 1, \dots, J, \tag{5.18}$$

which are correlated through the linear layer $\mathbf{A} \in \mathbb{R}^{C \times J}$. Thereby, each kernel function $k_j = k_{\boldsymbol{\theta}_{\text{GP},j}}(\mathbf{z}_j, \mathbf{z}'_j)$ parameterized by the hyperparameters $\boldsymbol{\theta}_{\text{GP}}$ takes a subset $\mathbf{z}_j \in \mathbf{z}$ of the latent activations as input. By applying nonlinear softmax activation $\sigma(\cdot)$ as introduced in (2.14), the SVDKL's likelihood $p(y \mid \mathbf{f}) = \sigma(\mathbb{f}(\mathbf{z}))$ is derived. A schematic diagram of a SVDKL model comprising a convolutional feature extractor is displayed in Figure 5.4.

By exploiting the above introduced SVGP approach with assuming independence of individual GPs, i.e., defining the set of independent inducing variables for all GPs $q(\mathbf{u}) = \prod_{j=1}^{J} q(\mathbf{u}_j)$, and using SVI under the assumption that the model's likelihood factorizes over individual samples, i.e., $p(\mathbf{y} \mid \bar{\mathbf{f}}) = \prod_{n=1}^{N} p(\mathbf{y} \mid \bar{\mathbf{f}}_n)$ with $\bar{\mathbf{f}} = \mathbb{f}(\mathbf{Z})$, Wilson *et al.* [171] derive the following minibatch sampling scheme

$$\mathcal{L}_{\text{SVDKL}} \simeq -\frac{N}{TB} \sum_{t=1}^{T} \sum_{b=1}^{B} \log p(y \mid \mathbf{f}^{(b,t)}) - \text{KL}\left[q(\mathbf{u}) \parallel p(\mathbf{u})\right] \tag{5.19}$$

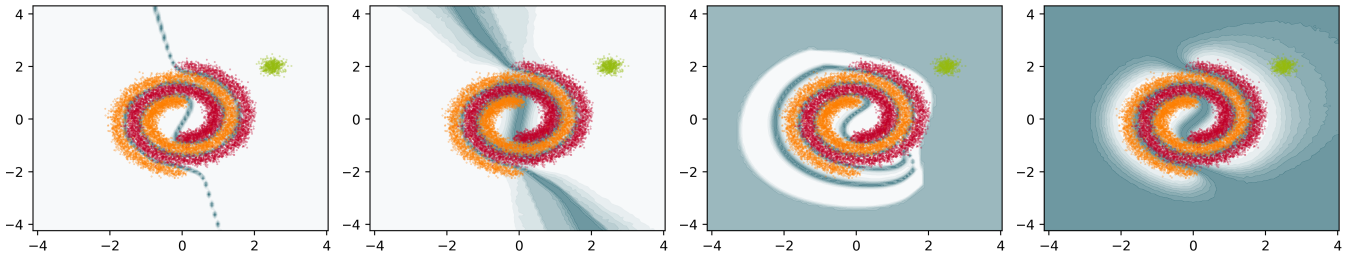
as an approximation to the ELBO of the SVDKL model. Thereby, $\mathbf{f}^{(b,t)}$ denotes the latent variable of the additive GP layer for the t-th Monte Carlo sample of the b-th input of the current minibatch, respectively, and the KL divergence between q and p has a closed-form solution. By additionally restricting the inducing points to a grid as well as exploiting kernel interpolation and structure in the covariance matrix, they further improve the efficiency and scalability of the proposed approach.[12] This sampling scheme allows for joint optimization of the DNN's parameters $\boldsymbol{\theta}_{\text{FE}}$, variational parameters $\boldsymbol{\theta}_{\mathbf{u}} = \{\boldsymbol{\mu}_j, \mathbf{C}_j \mid \forall j = 1, \ldots, J\}$, and the GP's kernel hyperparameters $\boldsymbol{\theta}_{\text{GP}}$ through automatic backpropagation and stochastic gradient descent optimization. As this scheme only requires drawing T samples from the variational distribution for a minibatch of size B, it only adds a small overhead to the regular NN training and inference procedure [171]. With their approach, Wilson *et al.* observed improved predictive performance over standard DNNs as well as independently training a GP on top of pre-extracted features of a CNN for several benchmark tasks. While not explicitly analyzing the quality of the uncertainty estimates, they reason that the deep kernel learning approach would automatically yield well-calibrated uncertainties.

[12] The interested reader is referred to [171] for details on the derivations of this approximation scheme.

Figures 5.5a to 5.5c illustrate the uncertainty estimates obtained from a deterministic NN, a frequently used BNN approximation exploiting MCD and DE, as well as an SVDKL model applied to a simple toy-classification task. The illustration highlights the promising uncertainty awareness of the deep kernel learning framework, i.e., the certainty of the model predictions significantly diminishes in regions with no training data, whereas both the deterministic and approximate BNNs widely extrapolate with high confidence and without communicating the lack of evidence from the training data.

5.2.5 Pathological behaviour of deep kernel learning

Recent studies showed that deep kernel learning can be prone to overfitting and, hence, produces unreliable uncertainty estimates despite the Bayesian modeling by means of the GP layer [44], [172], [174]. This was observed to be caused by increased model complexity [44], i.e., an overparameterized kernel function, and by the optimization of the ELBO, which can drive the GPs' kernels to overly correlate the training data leading to



(a) NN (b) DE-MCD-NN (c) SVDKL (d) DE-MCD-SVDKL

FIGURE 5.5: Predictive model uncertainty (■, a darker color equals a higher uncertainty) for a 2-dimensional classification task and the swiss-roll dataset comparing (a) a deterministic NN, (b) a BNN exploiting MCD and deep ensembling, (c) a vanilla SVDKL model, and (d) an SVDKL model equally extended with MCD and using deep ensembling. The training samples are given in red (●) and orange (●). Potential test samples are visualized by green dots (●).

feature collapse

feature collapse [172], [174]. That is, the NN-based kernel function learns a mapping that is insensitive to variations in the input space as both ID and OOD samples are mapped to similar latent features [174]. Ober *et al.* reason that this would be caused by minimizing the KL divergence of the ELBO, which can lead to covariance matrices that overly correlate all input samples. This forces the NN to be less sensitive to variations in the input images and suppress features that might not *deemed* necessary to the network for the classification task based on the training data but are exceptionally important to detect out-of-distribution samples.

prevention of the feature collapse

Whereas Ober *et al.* observe that this strong overfitting could *implicitly* be prevented to some extent by the regularization effect of stochastic mini-batch training [180], they conclude that a fully Bayesian treatment, i.e., additionally making the feature extractor Bayesian, would be preferable and *explicitly* fixes this issue [44], [172]–[174]. Ober *et al.* verify their observations by training a simple DKL model using a HMC procedure for the task of a 1-d toy example sampling both the model weight and GP hyperparameters, which they find to resolve the pathologic behavior completely. This is nicely illustrated in [172, Fig. 1 (b) and 5 (a)]. However, due to the insufficient scalability of HMC to more complex models and larger data, they also show that using stochastic gradient Langevin dynamics as an approximate Bayesian method can yield sufficiently good results.

[180] Keskar *et al.*, "On large-batch training for deep learning: generalization gap and sharp minima" (2017)

MCD-SVDKL

Similarly, exploiting the close connection of MCD and VI, Tran *et al.* [44] apply the former to the feature extractor of the DKL model to induce an approximate Bayesian posterior over the latent variables and observe this to mitigate the issue. Due to the simplicity of this method, it is adopted

for this work to integrate an approximative Bayesian feature extractor to the SVDKL model. In the remainder of this work, this architecture will be referred to as MCD-SVDKL.

SN-SVDKL

More recently, Liu *et al.* [173] as well as van Amersfoort *et al.* [174] propose to tackle the feature collapse by enforcing the CNN's feature extractor to be approximately bi-Lipschitz. To this end, they constrain the spectral norm (SN) of residual blocks within their feature extractors to be smaller than a predefined constant. Van Amersfoort *et al.* reason that this would drive the latent representation to be input distance aware, i.e., force the mapping from the input to the latent space to be both sensitive and smooth to changes in the input. This translates to that changes in the input space are required to induce a change in the latent space and that small changes in the input should equally cause small shifts in the output. They observe that using spectral norm could prevent feature collapse and hence, also improve the uncertainty calibration of the DKL model. Adding to this, van Amersfoort *et al.* also apply spectral norm to batch norm layers within their model.

In contrast to the MCD-extended DKL model proposed by Tran *et al.*, a big advantage of the spectral-normalized DKL approach is that it does not require additional sampling during inference. To approximate the spectral norm of both the fully connected and convolutional layer weights, the power iteration method can be used, which adds only a small runtime overhead [173], [174]. The model architecture that integrates a spectral-normed feature extractor into the SVDKL model will be referred to as SN-SVDKL in the remainder of this work.

Whereas the feature collapse not being a highly prominent issue for the toy example presented in Figure 5.5c, extending the SVDKL model by using MCD and building a DE again can significantly improve the uncertainty calibration as visible in Figure 5.5d.

5.2.6 Baseline setup

For the following analysis, all models are implemented and trained using the PyTorch [149] framework. The EfficientNet-B0 [151] is used as the baseline model, as the authors who proposed this architecture show it to have similar performance compared to the ResNet-50, which is used in the analysis of Band *et al.* [167], whereas being more efficient. This is achieved by

[149] Paszke *et al.*, "PyTorch: an imperative style, high-performance deep learning library" (2019)

(CNN)

[151] Tan and Le, "EfficientNet: rethinking model scaling for convolutional neural networks" (2019)

introducing mobile inverted bottleneck convolution (MBC) blocks, which are based on the inverted residual block used in the MobileNetV2 [181] additionally integrating squeeze-and-excitation (SE) [163] modules. That is, first, the feature depth of the input activation is expanded by a single 1×1 convolution block including BN and nonlinear activation, followed by a depthwise convolution block — equally including BN and nonlinear activation. Second, a SE attention module is applied to recalibrate channel importance [163]. Finally, the resulting activations are projected back to the original low-dimensional feature representation using another 1×1 convolution block with sequential application of BN (without nonlinear activation), which are added to the input activation of the MBC block, i.e., a residual mapping is applied. Both the inverted residuals and SE layers were found to increase the model performance over standard convolutions while the former improves memory efficiency [181] and the latter only adds marginal computational cost [163].

[181] Sandler *et al.*, "MobileNetV2: inverted residuals and linear bottlenecks" (2018)

[163] Hu *et al.*, "Squeeze-and-excitation networks" (2020)

The EfficientNet-B0 is set up by interchanging the original fully connected layer with a new one that outputs the desired number of target classes ($C = 5$). The model is trained using L2-norm regularization and the NLL objective function as introduced in (2.11)

$$\mathcal{L}_{\text{CNN}} = -\sum_{b=1}^{B} \log p(y_b \mid \mathbf{X}_b, \boldsymbol{\theta}), \tag{5.20}$$

with B denoting the number of data samples of the current minibatch.

(MCD-CNN)

As approximate Bayesian baselines, MCD- and DE-based CNNs are deployed. To implement the former, regular elementwise Bernoulli dropout is applied to the latent features of the baseline CNN prior to the final fully connected layer. As regular dropout was shown to primarily slow down training when used in conjunction with convolutional layers, spatial dropout [182] is added after each of the building blocks within the convolutional backbone of the EfficientNet-B0, which in contrast drops entire activation maps instead of single pixels of the intermediate actions. Furthermore, stochastic depth, i.e., bypassing residual blocks with identical input-output dimensions while decreasing the survival probability of each block's activations with model depth is applied during both model training and prediction. This can be interpreted as randomly sampling models with differing depths to improve the Monte Carlo samples [183], [184].

[182] Tompson *et al.*, "Efficient object localization using convolutional networks" (2015)

[183] Huang *et al.*, "Deep networks with stochastic depth" (2016)

[184] Antoran *et al.*, "Depth uncertainty in neural networks" (2020)

Following the implementation of Band *et al.* [167], the DEs are built from the deterministic baseline model by sampling the individual members from the pool of trials conducted to train the baseline CNN without replacement to reduce computational costs. Analogously, a third Bayesian baseline model is derived by combining MCD and deep ensembling. All DE models are set up to comprise five model instances.

(DE-CNN)

(DE-MCD-CNN)

5.2.7 Deep kernel learning implementation

To implement the SVDKL models, the GPyTorch [185] toolbox is adopted following the proposed setup by Wilson *et al.* [171]. The backbone of one of the pretrained baseline CNNs is used as the feature extractor transforming the input to the latent space $\mathbf{z} = h(\mathbf{x}, \boldsymbol{\theta}_{\mathrm{FE}}) \in \mathbb{R}^Q$ upon which a layer with $J = Q$ GPs is placed. Each deep kernel learning model is trained by minimizing the negative approximate ELBO according to (5.19): first fine-tuning only the variational and kernel hyperparameters upon a pretrained CNN backbone and then unfreezing all parameters for full model end-to-end training. An RBF kernel with 64 inducing points is used. The grid bounds for each kernel are set to $(-10, 10)$. Except for the elementwise dropout applied to the latent features, the same dropout scheme is used for the MCD-based SVDKL models as for the MCD-CNN model.

[185] Gardner *et al.*, "GPyTorch: blackbox matrix-matrix Gaussian process inference with GPU acceleration" (2018)

(SVDKL)

(MCD-SVDKL)

The spectral normed SVDKL models are adapted from the implementation of van Amersfoort *et al.* [174]. To handle the grouped convolutions that are extensively used within the EfficientNet-B0, the SN is computed for each group individually. In contrast to the ResNet-18 adopted by van Amersfoort *et al.* that uses ReLU nonlinearity, SiLU activations are utilized within the EfficientNet. In this work, these are replaced with their Lipschitz equivalent [186]

(SN-SVDKL)

[186] Chen *et al.*, "Residual flows for invertible generative modeling" (2019)

$$\mathrm{LipSiLU}(x) = \frac{\mathrm{SiLU}(x)}{1.1} = \frac{x\sigma(x)}{1.1}, \tag{5.21}$$

where $\sigma(x)$ denotes the sigmoid function to restrict the Lipschitz constant to fall below one and, hence, minimize the influence of the nonlinear activation on the overall SN of the residual model.

Equivalently to the process described above, ensembles of SVDKL models are built by model averaging over five model instances.

(DE-SVDKL)

5.2.8 Training protocol

All model instances are *solely* trained on the EyePACS training split using the Adam optimizer for 200 epochs and from ten randomly drawn seeds, i.e., random model initializations. PyTorch's automatic mixed precision package is exploited, i.e., native PyTorch layers use half precision whenever possible to speed up model training. Accordingly, ten deep ensembles are created per model configuration as described above. Model convergence at training time is tracked by measuring the macro AUROC on the *public* EyePACS data split, reserving the *private* holdout split for model testing. For the final model selection, the weights of the epoch with the best validation AUROC score were restored.

A plateau scheduler is implemented to halve the learning rate after ten epochs without improvement. Additionally, early stopping is used to terminate the training of individual trials after 20 epochs without improvement after a burn-in phase of 30 epochs. For MCD-based models, the number of Monte Carlo samples drawn was set to $T = 1$ during training, $T = 3$ during validation, and $T = 32$ at test time. Accordingly, $T = 32$ samples are drawn from the variational distribution of the SVDKL models for prediction.

Similar to the class weighting applied by [167], [169], [170], the strong imbalance of the DR severity in the EyePACS data is countered by randomly sub- and oversampling the majority and minority classes, respectively, with a probability corresponding to their inverse relative class frequency to ensure that rare classes are sufficiently represented in the training data. The oversampling strategy is paired with online data augmentation according to section 5.2.2 to additionally improve model generalization.

Model and optimization hyperparameters, i.e., learning rate, batch size, L2-norm regularization, droprates, and SN coefficients, were optimized using the Ax-platform [187] toolbox. A separate optimization loop with a maximum of 300 trials per method while optimizing for the macro AUROC score was run. Each hyperparameter optimization loop was manually pruned if no significant improvement of the performance or reduction of the spectral norm coefficients was observed. Whereas hyperparameter optimization for non-spectral normed models used ten initial Sobol steps followed by parameter selection based on Gaussian Process Expected Improvement, a multi-objective optimization was applied for spectral normed

[187] Bakshy *et al.*, "AE: a domain-agnostic platform for adaptive experimentation" (2018)

models to concurrently reduce the coefficients used to constrain the SN while maximizing the AUROC score. Image augmentation parameters were determined in an initial optimization loop using the baseline CNN.

5.2.9 Evaluation protocol

To verify the working hypothesis, i.e., that the deep kernel learning framework can provide a benefit for uncertainty calibration while retaining high diagnostic performance, each method is evaluated for the task of predicting the five DR severity stages (sDR). As the calibration of uncertainty was observed to be task-dependent [167], [168] and, in particular, the detection of rDR is of high importance within the DR screening setting, the severity grading is additionally generalized to this binary detection task. To this end, the predictive probabilities for the sDR stages 2–4 are aggregated to the rDR class (1) and stages 0–1 to the no rDR class (0), resulting in the set of tasks

$$\mathcal{Q} = \{\text{sDR}, \text{rDR}\}\,. \tag{5.22}$$

In-distribution diagnostic performance

To analyze each method's in-distribution predictive performance, the macro AUROC, accuracy (ACC), NLL, and quadratically weighted Cohen's kappa (QWK) [188] are used, i.e., the set of ID diagnostic performance metrics is defined as

$$\mathcal{M}_{\text{ID,P}} = \{\text{AUROC},\ \text{ACC},\ \text{QWK},\ \text{NLL}\}\,. \tag{5.23}$$

[188] Cohen, "Weighted kappa: nominal scale agreement provision for scaled disagreement or partial credit." (1968)

In contrast to other standard metrics that are agnostic to the severity of misclassifications, the QWK score is a *measure of reliability*. It is computed based on Cohen's kappa

Cohen's kappa

$$\kappa = \frac{p_o - p_c}{1 - p_c} = 1 - \frac{q_o}{q_c}, \tag{5.24}$$

which measures the inter-rater agreement by correcting the observed agreement p_o of predicted and target class labels by the agreement expected by chance p_c or the observed disagreement $q_o = 1 - p_o$ and disagreement by

chance $q_c = 1 - p_c$, respectively [188]. By replacing q_o and q_c with a weighted disagreement q'_o and q'_c, the weighted Cohen's kappa score

weighted Cohen's kappa

$$\kappa_w = 1 - \frac{q'_o}{q'_c} = 1 - \frac{\sum v_{ij} p_{o,ij}}{\sum v_{ij} p_{c,ij}}, \tag{5.25}$$

can be derived with weights v_{ij}, the observed proportion of judgements $p_{o,ij}$, and the expected proportion of judgments by chance $p_{c,ij}$ for the i,j-th confusion matrix entry [188]. With applying quadratically increasing weights based on the distance between observed and target DR severity grade, severe misclassifications are increasingly penalized over small deviations by using the QWK, i.e., it measures how well the predictions are aligned with the true targets. While a perfect fit would yield a score of $\kappa_w = 1$, predicting by chance results in $\kappa_w = 0$ [188]. In general, the kappa score can also be negative, i.e., $\kappa_w < 0$, which relates to worse than random guessing but is rarely encountered in practice [189]. Note that, in the remainder of this work the QWK score will be expressed as percentage w.r.t. to perfect agreement. When being applied to binary classification tasks it reverts to computing the regular kappa score.

[189] Cohen, "A coefficient of agreement for nominal scales" (1960)

In-distribution uncertainty

Each method's in-distribution uncertainty calibration is analyzed according to the expected calibration error (ECE) [73], [190] and area under the accuracy-rejection curve (AUARC) [191], for which the set of in-distribution uncertainty metrics is defined as

[190] Naeini *et al.*, "Obtaining well calibrated probabilities using Bayesian binning" (2015)

[191] Nadeem *et al.*, "Accuracy-rejection curves (arcs) for comparing classification methods with a reject option" (2009)

$$\mathcal{M}_{\mathrm{ID,U}} = \{\mathrm{ECE},\ \mathrm{AUARCt},\ \mathrm{AUARCa}\}\,. \tag{5.26}$$

expected calibration error

The expected calibration error (ECE) summarizes the deviation of the observed model confidence to a perfectly calibrated model in a single scalar value. The score is based on the assumption that the observed accuracy within a sample population matches the mean model confidence: For a set of samples that are predicted with average confidence χ the accuracy within this subset is expected to be equal or at least very close to χ. To compute the score, test samples are sorted into $b = 1, \ldots, B$ equally spaced bins $\mathcal{B}_b = \left\{\tilde{\mathbf{X}}, \tilde{\mathbf{Y}}\right\} \subset \mathcal{D}$ based on the model's confidence for each sample.

By computing the average confidence

$$\mathrm{conf}(\mathcal{B}_b) = \frac{1}{|\mathcal{B}_b|} \sum_{i \in \mathcal{B}_b} \max_{c \in C} p(y \mid \mathbf{x}_i, \boldsymbol{\theta}) \tag{5.27}$$

and the accuracy

$$\mathrm{acc}(\mathcal{B}_b) = \frac{1}{|\mathcal{B}_b|} \sum_{i \in \mathcal{B}_b} \mathbb{1} \left(\arg\max_{c \in C} \mathbf{y}_i = \arg\max_{c \in C} p(y \mid \mathbf{x}_i, \boldsymbol{\theta}) \right) \tag{5.28}$$

for each bin, the ECE is calculated as

$$\mathrm{ECE} = \sum_{b=1}^{B} \frac{|\mathcal{B}_b|}{N} \left| \mathrm{acc}(\mathcal{B}_b) - \mathrm{conf}(\mathcal{B}_b) \right|, \tag{5.29}$$

with $|\mathcal{B}_b|$ being the number of samples in the b-th bin. The metric resembles a quality measure for aleatoric uncertainty [192].

[192] Ries *et al.*, "Comparing the quality of neural network uncertainty estimates for classification problems" (2022)

area under the accuracy-rejection curve

The AUARC metric, in contrast, mimics the screening setting that allows referring patients to specialists for further inspection. In particular, the model may refer samples with high uncertainty, i.e., samples that are either difficult to distinguish from another class or novel to the model. To compute the metric, the accuracy rejection curve is constructed by varying the fraction of referred samples, whereby the most uncertain samples are rejected and the accuracy is computed on the remaining test samples. The score is then given by integrating the area under the obtained curve. An ideal model would exhibit highly correlated uncertainty estimates to erroneous predictions [167], i.e., the score measures how well the uncertainty estimates align with the model's failures. Thus, the AUARC is expected to increase with better uncertainty calibration. Since the epistemic uncertainty would ideally only provide insights on the predictive uncertainty in low-density or unpopulated regions of the data space, the total and aleatoric predictive uncertainties are particularly relevant for the in-distribution task setting. Therefore, two independent AUARC scores based on the total (AUARCt) and aleatoric uncertainty (AUARCa), are computed to estimate the quality of the in-distribution uncertainty estimates. Following Smith and Gal [91], as well as Band *et al.* [167], the total predictive uncertainty from a given set of $t = 1, \ldots, T$ predictive samples $p_t(y|\mathbf{x}, \boldsymbol{\theta})$ is

total uncertainty

[91] Smith and Gal, "Understanding measures of uncertainty for adversarial example detection" (2018)

computed as

$$\mathcal{U}_{\text{tot}} = \mathcal{H}\Big(\mathop{\mathbb{E}}_{t}\left[p_t(y \mid \mathbf{x}, \boldsymbol{\theta})\right]\Big) = \mathcal{H}\Big(\frac{1}{T}\sum_{t=1}^{T} p_t(y \mid \mathbf{x}, \boldsymbol{\theta})\Big) \tag{5.30}$$

which equals computing the entropy $\mathcal{H}$ of the expected predictive distribution, i.e., the average over all samples obtained from the model. Analogously, the aleatoric predictive uncertainty is defined as

aleatoric uncertainty

$$\mathcal{U}_{\text{aleat}} = \mathop{\mathbb{E}}_{t}\Big[\mathcal{H}\left(p_t(y \mid (\mathbf{x}, \boldsymbol{\theta}))\right)\Big] = \frac{1}{T}\sum_{t=1}^{T} \mathcal{H}\Big(p_t(y \mid \mathbf{x}, \boldsymbol{\theta})\Big), \tag{5.31}$$

i.e., the average entropy computed over all samples from the posterior.

Out-of-distribution analysis

Similar to the severity shift analysis conducted by Band *et al.* [167], this chapter analyzes each method's performance of how well OOD samples can be detected in order to refer these to a specialist. Following the assumption that a well-calibrated model is expected to exhibit high uncertainty on unfamiliar samples that have not been part of the training data, each model's predictive uncertainty estimates for the OOD datasets are collected. Subsequently, the performance for separating the OOD data from the EyePACS data *solely* based on both the model's predictive total (AUROCt) and epistemic uncertainty estimates (AUROCe) measured through the AUROC score is analyzed. To create a balanced classification setting, the number of samples of the EyePACS and the OOD dataset are equalized using random subsampling. Thereby, according to [91], [167], the epistemic predictive uncertainty can be computed as

epistemic uncertainty

$$\begin{aligned} \mathcal{U}_{\text{epist}} &= \mathcal{U}_{\text{tot}} - \mathcal{U}_{\text{aleat}} \\ &= \mathcal{H}\left(\mathop{\mathbb{E}}_{t}\left[p_t(y \mid \mathbf{x}, \boldsymbol{\theta})\right]\right) - \mathop{\mathbb{E}}_{t}\left[\mathcal{H}\left(p_t(y \mid (\mathbf{x}, \boldsymbol{\theta}))\right)\right]. \end{aligned} \tag{5.32}$$

With this, the set of metrics to measure the OOD uncertainty calibration is defined as

$$\mathcal{M}_{\text{OOD}} = \{\text{AUROCt},\ \text{AUROCe}\}. \tag{5.33}$$

5.2.10 Statistics

To assess if the observed in-distribution results in Table 5.1 and out-of-distribution outcomes in Figure 5.7 show statistically significant differences, multiple analysis of variances (ANOVAs) are applied for each metric individually. If statistically significant differences are found, post-hoc two-sample t-tests are applied to compare any of the selected SVDKL-based models to *all* baseline models. To estimate statistical significance of each of the factor's (SVDKL, SN, MCD, and DE) impact on the observed model performance and uncertainty calibration in the ablation study, the Wilcoxon signed-rank test is applied to the distribution of the performance differences in Figure 5.8, independently for each factor and metric. In order to correct for the occurrence of Type-I errors, the Bonferroni-Holm correction is applied for both test settings. A significance level of $\alpha = 0.05$ is used to determine statistical significance. For all tests, missing samples, i.e., scores of the CNN and SN-CNN for the AUARCa, were dropped to compute the respective tests.

5.3 Results

The outline of the subsequent results section for the conducted analysis is structured as follows: In section 5.3.1 the effect of the individual extensions applied to the SVDKL framework to the quality of the uncertainty estimates for the sDR grading and rDR detection task are analyzed. Subsequently, a comparison of the SVDKL framework to the baseline models is presented in section 5.3.2 analyzing both the predictive performance as well as the ID- and OOD uncertainty calibration. Finally, in section 5.3.3 the results of an ablation study are provided, which examines the impact of the individual extensions and the deployed base model on the quality of the uncertainty and the predictive performance.

5.3.1 Benefit of the deep kernel learning extensions

Firstly, the occurrence and mitigation of the pathological behavior, which recent studies found to cause uncalibrated model uncertainty when using the vanilla SVDKL model [44], [172]–[174], is explored. That is, the benefit of the extensions proposed in the literature, i.e., combining SVDKL with a spectral normed feature extractor, or a fully Bayesian approach exploiting

MCD or DEs, on the aleatoric and epistemic uncertainty calibration is analyzed on both ID and OOD data. To this end, the set

$$\begin{aligned}\mathcal{A}_{\mathrm{I}} = \{&\text{SVDKL}, \text{SN-SVDKL}, \text{MCD-SVDKL},\\ &\text{SN-MCD-SVDKL}, \text{DE-SVDKL},\\ &\text{DE-SN-SVDKL}, \text{DE-MCD-SVDKL},\\ &\text{DE-SN-MCD-SVDKL}\}\end{aligned} \tag{5.34}$$

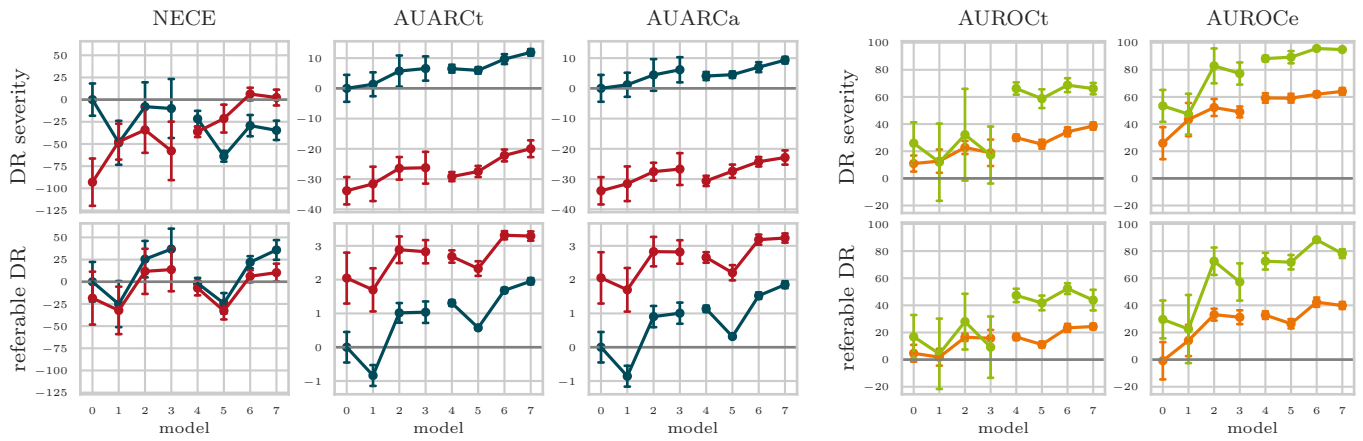
of all combinations of SVDKL and the proposed extensions is introduced.

In-distribution uncertainty analysis To analyze the ID uncertainty calibration, the average performance over all randomly seeded trials is computed for each deep kernel learning model in $\mathcal{A}_{\mathrm{I}}$ evaluated for all configurations contained in the set

$$\mathcal{D}_{\mathrm{ID}} \times \mathcal{M}_{\mathrm{ID,\,U}} \times \mathcal{Q}. \tag{5.35}$$

The results of this analysis are displayed in Figure 5.6a.

From these, both the AUARCt and AUARCa can be observed to be increasingly improved overall by the proposed extensions for the sDR grading task throughout both ID datasets. As a result, the findings show the DE-based SVDKL models extended with both SN and MCD (DE-SN-MCD-SVDKL) to provide superior performance, i.e., a relative performance gain of the AUARCt by about 12 % can be observed on the EyePACS test set from the basic SVDKL model to using the fully extended model architecture. Despite being less pronounced, this also applies to the binary rDR detection task for which a relative performance gain of about 2 % is observable for the same example as above. In contrast to this clear trend, the measured ECE exhibits high noise making the interpretation difficult. However, the sDR grading on the EyePACS dataset, a general trend of the ECE score to perform worse with adding the extensions to the vanilla SVDKL model can be observed. In contrast, for the rDR detection and on the IDRiD data in general the ECE improves analogously to the AUARC, except for the SN-SVDKL for the binary and the DE-SN-MCD-SVDKL for the multiclass task.



(a) In-distribution analysis on the EyePACS (■), and IDRiD (■) datasets.

(b) Out-of-distribution analysis on the RFMID (■) and SIIM-ISIC (■) datasets.

FIGURE 5.6: Relative performance [%] of the applied metrics comparing the SVDKL model variants, i.e., SVDKL (0), SN-SVDKL (1), MCD-SVDKL (2), SN-MCD-SVDKL (3), DE-SVDKL (4), DE-SN-SVDKL (5), DE-MCD-SVDKL (6), and DE-SN-MCD-SVDKL (7) for both the sDR grading (top row) and rDR detection (bottom row) task. For better visual clarity, the non-DE (left) and DE-based models (right) are grouped within each subplot. For the ID analysis in (a), the displayed performance is normalized to the basic SVDKL's (model 0) average performance on the EyePACS data. Note that the negative ECE (NECE) is displayed and pairwise identical scales are applied to the AUARC plots of the sDR and rDR task, as well as for the NECE for better comparison and to retain the qualitative differences of the relative performance increases. For the OOD analysis in (b), the displayed OOD detection performance is normalized to the AUROC performance corresponding to a random referral.

▷ Exploiting MCD within the SVDKL's feature extractor and deep ensembling overall improves the ID uncertainty estimates for both the sDR grading and rDR detection.

Out-of-distribution analysis To validate the alignment of uncertainty to the occurrence of unfamiliar samples, the AUROC performance according to section 5.2.9 is computed for discriminating the ID EyePACS data from samples of the OOD datasets solely based on the captured uncertainty estimates. To this end, analogously to the above analysis, the average performance over all independent, randomly initialized trials is computed for each SVDKL model in $\mathcal{A}_{\mathrm{I}}$ and evaluated for all configurations in the set

$$\mathcal{D}_{\mathrm{OOD}} \times \mathcal{M}_{\mathrm{OOD}} \times \mathcal{Q}, \tag{5.36}$$

for which the results are displayed in Figure 5.6b.

A bad, near-random separation performance can be observed on average for using the vanilla SVDKL model to detect the RFMID samples based on the *total* predictive uncertainty, i.e., the relative performance gain over a random baseline falls below 11 %. This matches the findings of van Amersfoort *et al.* [174], who equally observe random AUROC performance for discriminating between ID (CIFAR10 [193]) and OOD (SVHN [194]) samples. Similarly, the results show a low detection performance for the task-unrelated OOD samples of the SIIM-ISIC dataset, i.e., the relative performance gain over a random baseline falls below 26 %. However, the *epistemic* uncertainty-informed OOD detection performs significantly better. In particular, for the sDR grading, an average relative performance gain of approximately 53 % is observable for the detection of the SIIM-ISIC data using the vanilla SVDKL model that corresponds to a detection performance of AUROCe $\approx$ 77 %, which is typically considered fairly good. Nevertheless, when contrasting the ID samples with the task-related, near-ID samples of the RFMID dataset, the model still does not detect the latter with sufficient accuracy, which is particularly striking for the rDR detection task setting.

[193] Krizhevsky, "Learning multiple layers of features from tiny images" (2009)

[194] Netzer *et al.*, "Reading digits in natural images with unsupervised feature learning" (2011)

▷ The vanilla SVDKL model can only separate the ID from the task-unrelated OOD samples of the SIIM-ISIC dataset and by using the epistemic uncertainty estimates.

By adding the proposed extensions, a striking improvement of the OOD detection can be observed for both the near-ID (RFMID) and OOD (SIIM-ISIC) datasets, showing a better alignment of the observed uncertainty to the occurrence of unfamiliar, unknown samples. Thereby the epistemic uncertainty particularly benefits from the extended model architectures and sampling schemes. With this, the observed relative performance gain over a random referral for using the fully extended deep kernel learning model (DE-SN-MCD-SVDKL) in the sDR grading setting is on average at 64 % and 95 %, which corresponds to an AUROCe detection performance of the RFMID and SIIM-ISIC samples of about 82 % and 98 %, respectively — a very high, near-perfect detection rate. Similar to the findings from the ID performance analysis, the OOD sample detection is impaired for using spectral normed SVDKL models. Following this ambiguity, the best de-

tection performance is achieved by extending the SVDKL model by using MCD in combination with building deep ensembles (DE-MCD-SVDKL).

▷ Extending the SVDKL framework with MCD and additionally using DEs strikingly improves the detection performance of both near-ID and OOD samples from the RFMID and the SIIM-ISIC database, respectively.

5.3.2 Baseline comparison

The following experiment analyzes whether the deep kernel learning framework can add value to the ID predictive performance and uncertainty calibration as well as to the quality of the uncertainty estimates on the OOD data in comparison to the selected baseline models as introduced in section 5.2.6. As the vanilla SVDKL framework proved to lack a sufficient quality of uncertainty calibration, the overall best-performing models using the introduced extensions to prevent the feature collapse pathology, i.e., the DE-SN-MCD-SVDKL and DE-MCD-SVDKL, are selected for this analysis. Hence, the set of models that are subject to this comparison is defined as

$$\begin{aligned}\mathcal{A}_{\text{II}} = \{&\text{CNN}, \text{MCD-CNN}, \text{DE-CNN}, \text{DE-MCD-CNN},\\ &\text{DE-MCD-SVDKL}, \text{DE-SN-MCD-SVDKL}\}.\end{aligned} \tag{5.37}$$

In-distribution analysis The results for the ID baseline comparison are presented in Table 5.1. These show the observed average performance over all randomly seeded trials for each model in $\mathcal{A}_{\text{II}}$ evaluated for each task setting configuration in

$$\mathcal{D}_{\text{ID}} \times (\mathcal{M}_{\text{ID, P}} \cup \mathcal{M}_{\text{ID, U}}) \times \mathcal{Q} \tag{5.38}$$

with the standard deviation enclosed in brackets. As the deterministic baseline CNN only provides a MAP estimate, the aleatoric and total uncertainty computed according to (5.30) and (5.31) are identical. Therefore, only the AUARCt performance is reported for the vanilla CNN.

The presented results show an overall higher mean performance for the SVDKL-based models, which partially — most frequently for the AUROC,

TABLE 5.1: Comparison of the ID test performance for the EyePACS and IDRiD dataset and both the DR severity grading and the referable DR detection task. The highest mean values per metric are displayed in bold. Statistical significance according to section 5.2.10 is indicated with an asterisk.

(a) Performance for the DR severity grading

		SN	MCD	DE	AUROC [%] ↑	ACC [%] ↑	QWK [%] ↑	NLL ↓	ECE ↓	AUARC [%] ↑ total	AUARC [%] ↑ aleatoric
EyePACS	CNN	-	-	-	82.5 (0.3)	63.7 (5.4)	61.9 (1.5)	0.912 (0.036)	**0.105 (0.045)**	75.5 (4.5)	-
	BCNN	-	✓	-	84.0 (0.3)	63.5 (3.5)	64.0 (1.6)	0.892 (0.030)	0.120 (0.023)	76.3 (2.8)	75.3 (2.8)
		-	-	✓	84.9 (0.1)	68.1 (2.1)	66.6 (0.8)	0.843 (0.013)	0.144 (0.019)	80.7 (1.6)	78.8 (1.5)
		-	✓	✓	85.3 (0.1)	65.8 (1.4)	66.9 (0.4)	0.855 (0.010)	0.140 (0.009)	79.3 (1.1)	77.2 (1.0)
	SVDKL	-	✓	✓	85.4 (0.1)*	67.8 (2.1)	67.6 (0.7)	0.855 (0.012)	0.167 (0.015)	80.9 (1.2)	78.8 (1.2)
		✓	✓	✓	**86.2 (0.2)***	**70.3 (1.7)**	**70.0 (0.6)***	**0.826 (0.017)**	0.174 (0.014)	**82.5 (0.8)***	**80.5 (0.8)***
IDRiD	CNN	-	-	-	81.8 (1.6)	46.3 (5.0)	72.5 (3.5)	1.345 (0.165)	0.197 (0.045)	52.9 (3.8)	-
	BCNN	-	✓	-	81.9 (1.5)	43.7 (4.7)	72.5 (2.4)	1.275 (0.092)	0.190 (0.050)	52.1 (3.6)	51.4 (3.6)
		-	-	✓	84.3 (0.5)	48.2 (2.1)	74.7 (1.3)	1.093 (0.037)	**0.105 (0.018)**	58.7 (1.7)	55.9 (1.3)
		-	✓	✓	83.5 (0.5)	45.9 (2.1)	74.8 (1.2)	1.156 (0.028)	0.139 (0.024)	55.6 (1.5)	53.8 (1.3)
	SVDKL	-	✓	✓	84.9 (0.3)*	46.9 (1.3)	**76.9 (1.3)***	1.093 (0.015)	0.121 (0.009)	57.4 (1.4)	55.7 (1.2)
		✓	✓	✓	**85.0 (0.7)**	**49.2 (1.8)**	**76.9 (1.1)***	**1.091 (0.027)**	0.126 (0.011)	**59.0 (2.0)**	**56.8 (1.7)**

(b) Performance for the referable DR detection task

		SN	MCD	DE	AUROC [%] ↑	ACC [%] ↑	QWK [%] ↑	NLL ↓	ECE ↓	AUARC [%] ↑ total	AUARC [%] ↑ aleatoric
EyePACS	CNN	-	-	-	88.8 (0.8)	89.0 (0.6)	62.5 (1.9)	0.298 (0.013)	0.040 (0.013)	95.4 (0.4)	-
	BCNN	-	✓	-	90.0 (0.6)	89.7 (0.5)	64.9 (1.6)	0.285 (0.012)	0.054 (0.012)	96.1 (0.4)	96.0 (0.4)
		-	-	✓	90.6 (0.2)	90.2 (0.2)	65.8 (0.5)	0.271 (0.004)	**0.038 (0.004)**	96.7 (0.1)	96.5 (0.1)
		-	✓	✓	91.2 (0.1)	90.5 (0.1)	67.0 (0.5)	0.270 (0.002)	0.052 (0.004)	96.9 (0.1)	96.7 (0.1)
	SVDKL	-	✓	✓	91.1 (0.1)	90.4 (0.1)	66.1 (0.3)	0.272 (0.003)	0.047 (0.004)	96.8 (0.1)	96.6 (0.1)
		✓	✓	✓	**91.7 (0.2)***	**90.8 (0.2)***	**67.4 (0.8)**	**0.260 (0.005)***	**0.038 (0.007)**	**97.0 (0.1)***	**96.9 (0.1)***
IDRiD	CNN	-	-	-	96.0 (0.6)	89.7 (0.9)	77.7 (2.2)	0.256 (0.015)	**0.051 (0.010)**	97.6 (0.3)	-
	BCNN	-	✓	-	96.7 (0.5)	89.5 (1.0)	76.9 (2.3)	0.254 (0.023)	0.071 (0.011)	97.6 (0.5)	97.6 (0.5)
		-	-	✓	96.8 (0.1)	90.0 (0.6)	78.4 (1.3)	0.227 (0.004)	0.052 (0.009)	98.2 (0.1)	98.1 (0.1)
		-	✓	✓	**97.2 (0.1)**	89.5 (0.3)	77.0 (0.8)	0.236 (0.005)	0.072 (0.003)	98.0 (0.1)	97.9 (0.1)
	SVDKL	-	✓	✓	**97.2 (0.1)**	**91.1 (0.7)***	**80.7 (1.6)***	**0.219 (0.007)***	0.056 (0.004)	**98.3 (0.1)***	**98.2 (0.1)**
		✓	✓	✓	97.1 (0.1)	90.4 (0.5)	79.3 (1.1)	**0.219 (0.007)***	0.054 (0.006)	**98.3 (0.1)**	**98.2 (0.1)**

QWK, and AUARCt score — exhibits a statistically significant improvement over *all* the compared baseline models. The largest gain is visible for the QWK, which is exemplarily improved by 2.1 % for the sDR grading task on the IDRiD data comparing the DE-SN-MCD-SVDKL to the DE-MCD-CNN, suggesting that the predictions are more reliable. That is, they are better aligned with the target severity grades, and severe misclassifications are observed less frequently compared to the baselines. Similarly, the findings show the AUARC performance for the sDR grading task to increase by 1.8–1.9 % comparing the DE-SN-MCD-SVDKL to the best-performing non-SVDKL model, i.e., the DE-CNN, on the EyePACS data. Thereby, the AUARC computed based on the aleatoric uncertainty estimate can be

observed to be slightly outperformed by the score based on the total model uncertainty. This indicates that the additional epistemic uncertainty information inherent to the total uncertainty allows to more reliably identify samples within the ID test data that are unfamiliar from model training.

As for the previous analysis, the ECE score shows an exception to this trend: The lowest average calibration error is observed for using the vanilla CNN and the error increases for the Bayesian baselines as well as the DKL-based models for the sDR grading task on the EyePACS dataset. In contrast, the score decreases throughout the Bayesian baseline and SVDKL models and with this is approximately on par or better with the vanilla CNN for the rDR detection task and on the IDRiD dataset.

Interestingly, the findings show a highly reduced accuracy — and also a significantly reduced AUARC performance — for the sDR grading on the IDRiD data throughout the methods. Conversely, the QWK increases, showing that all methods are more reliable and make less severe errors. Both the baselines and SVDKLdkl-based models overall are observed to generalize well, i.e., they are mostly agnostic to the imposed country and target shifts by the IDRiD data particularly according to the AUROC and QWK, and for the task generalization to the rDR detection. Furthermore, the presented results outline that using either of the fully extended SVDKL-based architectures can improve over the standard approximate Bayesian CNNs, concerning both the diagnostic predictive performance and the uncertainty calibration measured through the AUARC.

▷ The SVDKL-based models overall improve both the predictive performance and uncertainty calibration w.r.t. the AUARC for the sDR grading and rDR detection over all baselines.

Out-of-distribution analysis Analogously to the OOD analysis of the previous section, in the following, each method's ability to detect unfamiliar image samples by using the total and epistemic predictive uncertainty estimates is evaluated. However, to get a more detailed measure of the uncertainty calibration for the task-related OOD samples, i.e., if higher uncertainty estimates are indeed related to samples containing unfamiliar pathologies and diseases, two new subsets of the RFMID data are introduced: The first set, the RFMID-O subset, only contains samples being

affected by *any other disease* than DR. In contrast, the second subset, which is referred to as RFMID-DR, only comprises subjects being *healthy* or suffering from *any DR* but no other disease to closely mimic the ID task data. In addition to the latter, also the IDRiD data is used within this analysis. Both these sets serve as a baseline due to the data and task similarity of the RFMID-DR and IDRiD to the EyePACS training data, for which, followingly, a lower performance compared to the semantically more distant OOD datasets should be observed. The results of this analysis are displayed in Figure 5.7 showing the average performance over all randomly seeded trials $t \in T$ for each model in $\mathcal{A}_{\text{II}}$ along with its standard deviation evaluated for all possible configurations in

$$\tilde{\mathcal{D}}_{\text{OOD}} \times \mathcal{M}_{\text{OOD}} \times \mathcal{Q} \tag{5.39}$$

where

$$\tilde{\mathcal{D}}_{\text{OOD}} = \mathcal{D}_{\text{OOD}} \cup \{\text{RFMID-O}, \text{RFMID-DR}, \text{IDRiD}\}\,. \tag{5.40}$$

results on the SIIM-ISIC dataset

As expected, both the approximate Bayesian CNNs as well as the selected SVDKL models outperform the baseline CNN on the detection of samples originating from the OOD SIIM-ISIC dataset within both the rDR detection and sDR grading for using the total uncertainty estimate. Thereby, the approximate Bayesian CNNs provide a strong baseline and partly outperform the SVDKL models. However, this is less pronounced for the epistemic uncertainty-based OOD detection, for which a near-perfect detection rate within the sDR grading for both the baselines and SVDKL models is observed with an average AUROC $> 94\%$. Referring to the more difficult task of distinguishing between the EyePACS and RFMID data, both SVDKL models show a statistically significantly higher mean detection performance based on the epistemic uncertainty estimate (AUROCe), i.e., an improvement of +2.0–3.0 % for the sDR grading and +1.8–3.1 % for the rDR detection task is observed.

results on the RFMID/IDRiD datasets

▷ The extended SVDKL models are observed to improve the near-ID detection w.r.t. the AUROCe score compared to the baselines for both task settings.

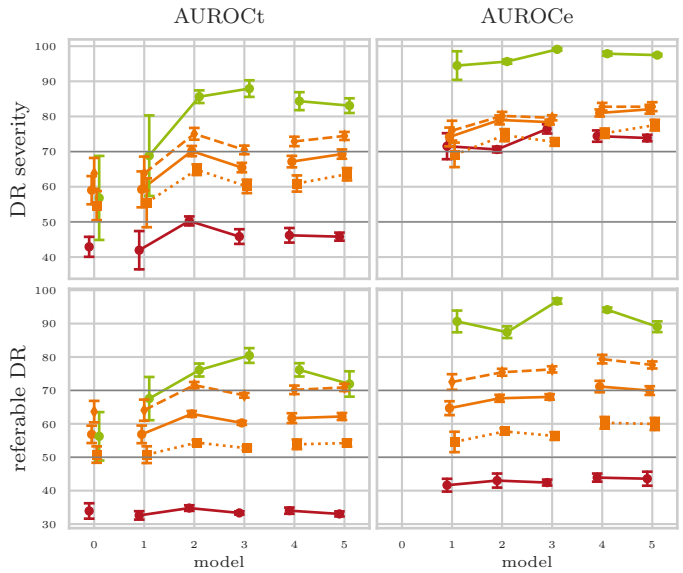


FIGURE 5.7: AUROC performance on distinguishing samples from the IDRiD (■), RFMID (■), and SIIM-ISIC (■) to those of the EyePACS dataset based on the total (left column) and epistemic uncertainty estimates (right column) for the CNN (0), MCD-CNN (1), DE-CNN (2), DE-MCD-CNN (3), DE-MCD-SVDKL (4), and DE-SN-MCD-SVDKL (5) models and both the DR severity grading (top row) and referable DR detection task (bottom row). The orange dashed and dotted lines indicate the RFMID-O and RFMID-DR subsets, respectively. The darker gray, horizontal lines indicate a usually considered fair performance (at AUROC=70 %) and a random classification as being out-of-distribution or not (at AUROC=50 %). For better visual clarity, the vanilla CNN (left), BNNs (middle), and SVDKL models (right) are grouped within each subplot. Please note that these results are provided in Table 1 in the appendix in more detail.

Overall, also the performance to detect the RFMID-O samples is observed to be higher than for the full RFMID data, outlining that the models are indeed more uncertain on the samples containing unfamiliar pathologies related to other eye diseases than DR. Accordingly, a lower detection performance for separating the EyePACS data from the RFMID-DR subset and IDRiD data is observable, which both entail only healthy subjects and patients with DR and, hence, resemble the original EyePACS task setting. That is, the Bayesian models yield well-calibrated uncertainty estimates that correlate well with the semantic distance to the training dataset.

Furthermore, comparing the two task settings in general, a higher detection performance is observable for the sDR grading than for the rDR detection task, which indicates that relevant information about the predictive uncertainty to separate ID and OOD samples from one another is lost due to the binary aggregation as introduced in section 5.2.9. This is becomes clear considering for instance the case that the model is uncertain to which of the DR severity classes 2–4 a subject belongs, but is certain that the subject does not belong to class 0 or 1. This information is lost by the aggregation to the alternative hypothesis space for the rDR detection.

▷ The aggregation to the binary task setting causes a loss of uncertainty information highlighting the task-dependency of the predictive uncertainty.

Interestingly, a fairly good AUROCe performance (AUROCe $> 70\,\%$) is observed for the sDR grading on the IDRiD data across all methods, i.e., the IDRiD samples can be separated quite well from the EyePACS data. This indicates that higher uncertainty estimates are observed for the IDRiD than for EyePACS data within the multiclass prediction setting. Moreover, the findings for the IDRiD data show a worse-than-random detection performance across all compared methods within the rDR detection setting, i.e., all models equally are more confident on the IDRiD than for the EyePACS data within the binary task setting. This indicates a strong distribution shift between both datasets.

5.3.3 Ablation study

Following the previous analysis, the extended SVDKL framework proved to add a benefit for both the diagnostic predictive performance and the quality of the uncertainty estimates. In particular, the *fully* extended SVDKL models (DE-SN-MCD-SVDKL and DE-MCD-SVDKL) were observed to provide valuable uncertainty estimates for detecting the near-ID samples of the RFMID dataset. Nevertheless, the standard approximate Bayesian CNNs yield a strong baseline performance. Hence, the following experiment aims to disentangle the individual effects of the applied extensions on the predictive performance and uncertainty calibration in comparison to the performance improvement that can be achieved by using a deep kernel learning-based BNN. To this end, every possible combination of SVDKL- and CNN-based models with all the examined extensions are trained and evaluated regarding their impact on the predictive performance and uncertainty calibration following both the ID and OOD analysis of the previous sections. In detail, first, the set of every possible combination of base model and extension used in this analysis is defined as

$$\begin{aligned}\mathcal{A}_{\text{III}} = \{&\text{CNN},\ \text{SVDKL},\\ &\text{SN-CNN},\ \text{SN-SVDKL},\\ &\vdots\\ &\text{DE-SN-MCD-CNN},\ \text{DE-SN-MCD-SVDKL}\}.\end{aligned} \tag{5.41}$$

Analogously to the preceding analyses, the average performance scores of all randomly seeded trials for each model $a \in \mathcal{A}_{\text{III}}$ denoted as $\bar{P}_z$ are computed

for every evaluation configuration

$$\mathcal{Z} = \mathcal{Q} \times (\mathcal{Z}_{\text{ID}} \cup \mathcal{Z}_{\text{OOD}}), \tag{5.42}$$

where

$$\mathcal{Z}_{\text{ID}} = \mathcal{D}_{\text{ID}} \times (\mathcal{M}_{\text{ID, P}} \cup \mathcal{M}_{\text{ID, U}}) \tag{5.43}$$

and

$$\mathcal{Z}_{\text{OOD}} = \mathcal{D}_{\text{OOD}} \times \mathcal{M}_{\text{OOD}}. \tag{5.44}$$

Subsequently, the average performance differences

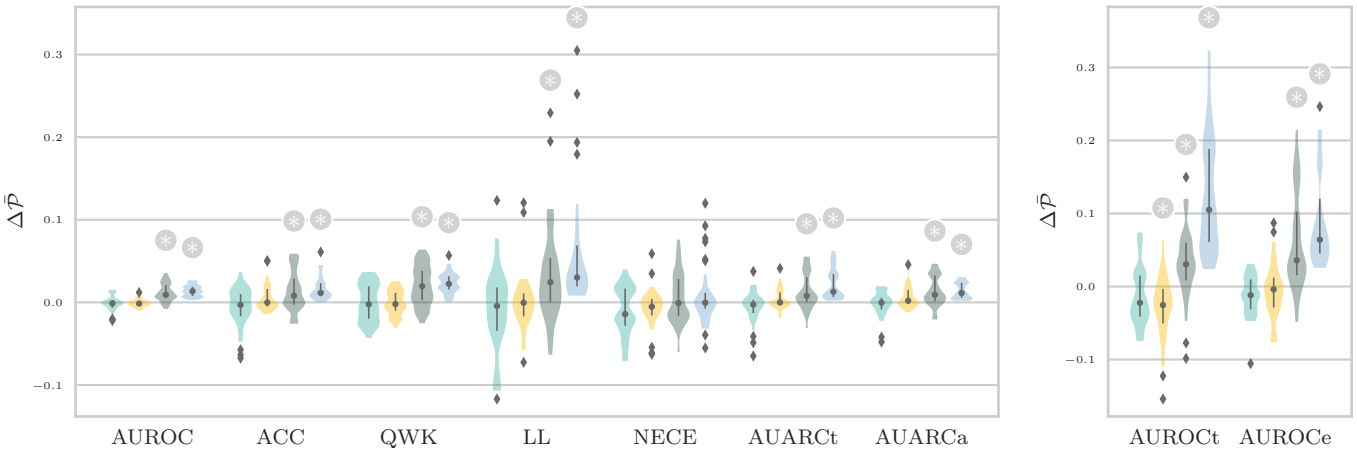
$$\Delta\bar{P}_{z,a,a'} = \bar{P}_{z,a} - \bar{P}_{z,a'} \tag{5.45}$$

are computed for paired trials that share the same evaluation configuration $z \in \mathcal{Z}$ and opposing model configurations a and a'. Thereby, opposing model configurations correspond to either using a CNN or SVDKL-based model and dis- or enabling a single extension (SN, DE, MCD) while keeping the other settings fixed. For instance, the differences according to (5.45) are computed for the configurations (a, a') = (SN-MCD-SVDKL, SN-MCD-CNN) and (SN-MCD-SVDKL, SN-SVDKL). Finally, the differences $\Delta\bar{P}_{z,a,a'}$ are aggregated over the task settings $\mathcal{Q}$ and test datasets $\mathcal{D}_{\text{ID/OOD}}$ to a single, independent set of observed differences $\Delta\bar{\mathcal{P}}$ per metric in $\mathcal{M}_{\text{ID/OOD}}$ and extension or base model, respectively. The observed results are displayed in the box-violin plots in Figure 5.8.

effect of DEs and MCD

The results of the ablation analysis show that building deep ensembles has a high, significant impact that mostly exceeds the effect of every other extension. The most striking impact can be observed within the OOD detection task w.r.t. either the total or epistemic predictive uncertainty estimates (AUROCt and AUROCe). Nonetheless, also the ID predictive performance and model calibration measured through the AUARC significantly benefit from using deep ensembles. Similarly, but less pronounced, adding MCD improves both the quality of uncertainty and predictive performance throughout the metrics.

▷ Adding MCD and exploiting DEs provide the strongest benefits w.r.t. the predictive performance and the quality of both the ID and OOD uncertainty estimates.



(a) ID analysis (b) OOD analysis

FIGURE 5.8: Distribution of the performance differences between using a CNN or SVDKL base model and any model with or without each main factor, i.e., SVDKL (■), SN (■), MCD (■), and DE (■) for (a) the ID and (b) OOD performance metrics. The results are displayed as box-violin plots, which show the median and IQR accompanied by the violins kernel density estimation (KDE) covering the 1.5-IQR range. More extreme points are considered outliers and are marked by a diamond (♦). Statistical significance estimated according to section 5.2.10 is indicated with a star above the respective violin. A higher delta indicates that adding the base model or extension improved the respective metric. Note that the log-likelihood (LL) and negative ECE (NECE) are displayed in (a) for better comparison, and that the H1 hypothesis of the reported statistical test corresponds to whether a statistically significant majority of paired differences improves or decreases the individual performance metric w.r.t. the selected base model or extension.

Again, as already observed within the previous analysis, the ECE imposes an exception to this trend, i.e., only some trials that use deep ensembles and MCD are observed to improve the model calibration according to the ECE.

effect of SN

Similarly, a high proportion of the trials for which the feature extractor is constrained to be approximately bi-Lipschitz by applying spectral normalization shows a negative impact on the observed model performance. Nonetheless, particularly for the AUROCe, the performance could be improved within some trials by leveraging spectral normalization to enforce an approximately bi-Lipschitz mapping in the feature extractor. This matches the observations that, e.g., the DE-SN-MCD-SVDKL model outperforms the DE-MCD-SVDKL model, but the SN-SVDKL model is observed to perform overall worse than the vanilla SVDKL model.

effect of SVDKL

Considering the impact of using SVDKL, overall similar proportions of the conducted trials are observed to either decrease or improve the model performance and uncertainty calibration. This suggests that — under specific circumstances — using the SVDKL approach seems to hurt the model

performance compared to not using deep kernel learning at all. This can be attributed to the fact that the vanilla SVDKL model performs at most on par or worse compared to the baseline CNN. That is, examining the results in more detail, the performance of the SVDKL model is decreased by 0.4–22.9 % w.r.t the vanilla CNN for all ID metrics ($\mathcal{M}_{\text{ID, P}} \cup \mathcal{M}_{\text{ID, U}}$) within the sDR grading on the EyePACS dataset. Nevertheless, the results also show a substantial proportion of trials for which adding SVDKL is observed to indeed induce a positive effect, e.g., for the ACC, QWK, NLL, ECE, AUROCt, and AUROCe, which is in line with the findings of the baseline comparison.

▷ Despite being less pronounced, SVDKL is observed to provide an additional benefit for both predictive performance and uncertainty calibration when mitigating the feature collapse by the analyzed extensions.

5.4 Discussion

In the following, the results and implications of the conducted experiments and the observed results will be discussed, closely following the structure of the results section. That is, first the findings of the analysis and curation of the feature collapse will be reviewed in section 5.4.1, followed by a discussion on the practical benefits of the SVDKL framework on the ID diagnostic performance as well as the ID and OOD uncertain calibration in sections 5.4.2 and 5.4.3, respectively. Subsequently, the results of the ablation study will be discussed in section 5.4.4, followed by a comment on the distribution shift introduced by the resampling applied during model training and a comparison of the achieved results to the related literature in sections 5.4.5 and 5.4.6.

5.4.1 Analysis of the feature collapse pathology

The findings confirm the occurrence of the observed pathologies caused by the vanilla SVDKL framework as observed in the literature [44], [172]–[174]. This highlights the necessity of regularizing the CNN being used as feature extractor within the SVDKL model to learn a distance-aware mapping, either by enforcing the mapping to be approximate bi-Lipschitz or

by full Bayesian modeling, i.e., using a non-deterministic, Bayesian feature mapping prior to the additive GP layer, which is utilized as the final layer in the deep kernel learning model. Particularly, the latter was observed to improve both the predictive performance as well as the ID and OOD uncertainty calibration, i.e., a better alignment of the uncertainty estimates with the model's failures or the occurrence of OOD samples. Thereby, the improvement is, expectedly, observed to be greater for the epistemic than for the aleatoric uncertainty estimates, because both building DEs and applying MCD aim to improve model diversity — the former by exploiting multi-modality of the complex posterior, the latter by capturing local model uncertainty. Similarly, minimizing the spectral norm guides the model towardss learning a distance-aware latent space representation. That is, all three methods are designed to prevent the mapping of OOD samples within the same region as ID samples. Nevertheless, by using the extended architectures over the vanilla SVDKL model also the aleatoric uncertainty estimates could be improved within the conducted experiments.

practical benefits of spectral norm

In contrast to the consistent performance improvement observed from using MCD and DE, the findings revealed the approximate bi-Lipschitz models by means of constraining the spectral norm to exhibit instabilities, and only sporadic improvements in the quality of uncertainty could be achieved. With this, the results — unexpectedly — do not fully support the overall positive findings of Liu *et al.* [173]. This, on the one hand, might be caused by the method's lack of guarantees for a true bi-Lipschitz mapping [174], i.e., too loose bounds on the spectral norm might not sufficiently prevent the feature collapse and, followingly, could cause arbitrary wrong uncertainty estimates. On the other hand, bounds too tight could cause unreliable convergence by preventing the feature extractor from learning a meaningful feature mapping. In accordance with this, the bound on the spectral norm emerged to be difficult to optimize potentially leading to the unreliable prevention of the feature collapse pathologies.

5.4.2 In-distribution performance and uncertainty calibration

ID predictive performance

Regarding the findings of the in-distribution analysis in section 5.3.2, the fully extended deep kernel learning-based architectures, i.e., the DE-SN-MCD-SVDKL and DE-MCD-SVDKL, are observed to mostly outperform

or be at least on par with both the deterministic and the Bayesian baseline models across all settings and metrics analyzed. Furthermore, the findings highlight the model's ability to generalize well for distribution shifts and different task settings. As this work focusses on evaluating the feasibility and benefit of deep kernel learning for the DR grading task over standard BNNs and vanilla CNNs, competing with the state-of-the-art diagnostic performance is beyond the scope of this work. As a result, both the baseline and SVDKL models do not achieve the performance reported in, e.g., [168]. Nonetheless, the observed diagnostic performance for the detection of rDR is close to par with the results presented in [167], [170]. Moreover, the findings show the SVDKL framework to improve the diagnostic performance over the presented baselines, and the performance of the former is expected to improve accordingly by applying standard scaling methods for deep learning such as using high-resolution images, more capacitive base models [151], and exploiting more sophisticated training schemes, e.g., exploiting binocular imaging [195]. Thereby, particularly the former could be expected to significantly improve the diagnostic performance according to [151], [161], [167]–[170], [196]. Additionally, although more capacitive backbones significantly increase the computational complexity and are more likely to overfit, the observed improvements of the extended SVDKL approach are expected to transfer to using models with higher depth and width, e.g., the EfficientNet-B7 [151], [197] or EfficientNetV2-L [159] given well-tuned hyperparameters.

[195] Qian *et al.*, "Two eyes are better than one: exploiting binocular correlation for diabetic retinopathy severity grading" (2021)

[161] Porwal *et al.*, "IDRiD: diabetic retinopathy – segmentation and grading challenge" (2020)

[196] Sahlsten *et al.*, "Deep learning fundus image analysis for diabetic retinopathy and macular edema grading" (2019)

[197] Chetoui and Akhloufi, "Explainable diabetic retinopathy using EfficientNET" (2020)

[159] Tan and Le, "EfficientNetV2: smaller models and faster training" (2021)

ID uncertainty calibration

The in-distribution uncertainty calibration, i.e., whether the predictive uncertainty is a reliable indicator for potential model failures, is considered an important model property that can improve usability and applicability in clinical practice by improving trust in the model's output. There is no single metric to comprehensively quantify the quality of the uncertainty calibration [105], which is additionally hampered by the common lack of a proper gold standard to compare with. The ECE is a frequently used metric for quantifying uncertainty calibration, which approximates the uncertainty of the model by the maximum confidence of the predictive distribution.

[105] Ashukha *et al.*, "Pitfalls of in-domain uncertainty estimation and ensembling in deep learning" (2020)

However, the ambiguous results obtained for the ECE in this analysis limit a profound and reliable conclusion on the model calibration based on this specific score. A potential explanation for these ambiguities may be the sensitivity to the model temperature, static binning scheme, and the class agnosticism of the metric, which in previous studies were found to

result in an unreliable approximation to the calibration error [95], [105], [198]. Whereas more sophisticated variants of the ECE, which exemplarily use adaptive binning and class-specific approaches [198], promise to mitigate the latter issues, other literature suggests that the metric can still yield a biased estimate [105]. To mitigate the metric's sensitivity to the model temperature, Ashukha *et al.* [105] propose to compare the ECE after applying temperature scaling [73]. However, the optimal model temperature obtained for a specific dataset was shown to unreliably generalize to distribution shifted data [95] suggesting that the practical utility of the temperature scaling procedure is limited to ID data. Moreover, as the ECE does not account for the model's full uncertainty of the predictive distribution, the metric might underestimate model uncertainty. In contrast, the AUARC is less sensitive to the model temperature [105] when being computed w.r.t. the fraction of referred samples instead of rejecting samples based on a fixed threshold as conducted in this work, and taking the full predictive distribution into account. Hence, the AUARC metric might be a more honest estimate of the model's uncertainty calibration. Consequently, it would be of great interest to evaluate the model calibration by means of more sophisticated metrics, e.g., the expected uncertainty calibration (UCE) [199], in addition to the AUARC metric in future work.

[198] Nixon *et al.*, "Measuring calibration in deep learning" (2019)

[199] Laves *et al.*, "Well-calibrated model uncertainty with temperature scaling for dropout variational inference" (2019)

Nonetheless, some general trends could be observed regarding the uncertainty calibration according to the ECE: At first sight, considering the sDR grading setting on the EyePACS dataset, the findings imply the predictive confidence to align worse with the actual expected diagnostic accuracy throughout the Bayesian baseline and SVDKL models compared to the vanilla, baseline CNN, particularly, for combining several of the analyzed extensions. At second sight, the calibration error of all Bayesian models observed under the task generalization to the rDR detection and on the IDRiD data set decreases overall with adding the extensions. This finally leads to a calibration error of the deep kernel learning-based models that is on par compared to the deterministic baseline CNN and the approximate BNNs, respectively, when being applied to shifted data and task settings.

Moreover, the results of this work, indeed, show an improvement of the ID uncertainty calibration measured through the AUARC, particularly, for using the extended SVDKL models, showing that a higher model uncertainty is positively associated with an increased probability of misclassification. Through the similarity of the AUARC metric and the AvUC loss proposed

in [200], analyzing the benefit of combining the latter with the SVDKL framework on the uncertainty calibration would be of great interest for future work.

[200] Krishnan and Tickoo, "Improving model calibration with accuracy versus uncertainty optimization" (2020)

5.4.3 Out-of-distribution uncertainty calibration

Moreover, the results show a significant improvement in the near-ID detection performance, i.e., on the RFMID dataset, by using the extended SVDKL models. This demonstrates the deep kernel learning framework to be more capable of detecting minor distribution shifts compared to the approximate Bayesian CNNs and highlights the methods's high sensitivity to subtle alterations in the input space that can be measured through the model's epistemic uncertainty estimates. With this, the findings propose the extended deep kernel learning-based models to be more likely to recognize their lack of knowledge when being applied to the task of DR grading, which, as a result, could minimize the risk of missing pathologies the network was not trained to detect. This is of particular importance in a setting where grading is conducted in primary care, as specialists may only be involved upon request — as indicated either by the model's uncertainty or when any referable DR is detected. Otherwise, patients would be prevented from receiving necessary therapeutic interventions to avert other potentially sight-threatening diseases from progressing. Similar to the findings by Hüllermeier *et al.* [92], the epistemic uncertainty in the results presented in section 5.3.2 was observed to be task-dependent, i.e., it partly collapses due to the aggregation from the sDR grading to the rDR detection task setting, to some extent impairing the model's ability to separate between ID and OOD samples. This emphasizes the importance of expanding the scope of the DR screening to additionally encompass the task of sDR grading, which not only facilitates disease monitoring and triaging patients for referral but also allows more informed decision-making and enhances clinical usability [168], [201].

[92] Hüllermeier and Waegeman, "Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods" (2021)

5.4.4 Ablation study

As was shown in the ablation study in section 5.3.3, exploiting model ensembling was observed to particularly improve the OOD detection performance. This overall higher performance could be attributed to the fact that the DEs can more effectively capture different modes of the posterior

than the single-instance, Bayesian CNNs analyzed in this work, aligning the observations of Wilson *et al.* [104]. In contrast, both MCD and SVDKL, either can be interpreted as or are VI methods, respectively, which allows these two methods to solely capture a single mode of the model's true, most likely complex and multimodal posterior adding only marginal information gain to the BMA. Nonetheless, both the MCD and the extended SVDKL framework were observed to provide an additional benefit for the uncertainty calibration, particularly in combination with building DEs.

5.4.5 Prior probability shift

Note that throughout the experiments, oversampling was applied to balance the class distribution of the EyePACS training data. From a Bayesian perspective, this relates to assuming a uniform, uninformative prior over the classes mimicking the setting where the prevalence of the DR severity in a patient population is completely unknown. Despite the model selection being conducted on the originally distributed validation data, this distribution shift could lead to under- and overconfident predictions for majority and minority classes impairing the uncertainty calibration on the test data. However, prior probability shifts most likely occur naturally due to a mismatch of the prevalence in the training and test data to the real-world prevalence encountered in clinical practice, e.g., due to regional differences or whether patients suffer from T1DM or T2DM as discussed in section 3.2. Hence, the experimental setup of this work mimics this setting and analyzes each model's ability to cope with such a prior probability shift.

However, in case the prevalence for sDR levels in the target population that the model is applied to is known a priori, the posterior probability distribution could be corrected accordingly to recalibrate the model for practical use and improve calibration [32, p. 146]. This can be achieved by rescaling the posterior

[32] Bishop and Bishop, *Deep Learning: Foundations and Concepts* (2024)

$$p_{\tilde{\mathcal{D}}}(y \mid \mathbf{x}^*, \mathbf{X}, \mathbf{y}) = \frac{q_y}{\sum_{c \in C} q_{y=c}} \tag{5.46}$$

where

$$q_y = \frac{p_{\mathcal{D}}(y \mid \mathbf{x}^*, \mathbf{X}, \mathbf{y}) p_{\tilde{\mathcal{D}}}(\mathbf{y})}{p_{\mathcal{D}}(\mathbf{y})} \tag{5.47}$$

and $p_{\mathcal{D}}(\mathbf{y})$ and $p_{\tilde{\mathcal{D}}}(\mathbf{y})$ denote the estimated prior probabilities of the classes in the training $\mathcal{D}$ and the domain data $\tilde{\mathcal{D}}$, respectively. Moreover, to im-

prove the model's internally learned estimate of the prior probability, the architecture could be extended to include clinical data such as medical history (e.g., hyperglycemia, blood pressure) or the type and duration of DM the patient suffers from, which the model could exploit to adjust the internal prior probability of the individual classes accordingly — given sufficient training data accurately modeling these prior probability shifts.

5.4.6 Literature comparison

Comparing the obtained quality of the ID and OOD uncertainty estimates to the related literature is difficult due to the lack of a standardized, quantitative evaluation protocol. Frequently only qualitative results are provided that highlight the overall importance and value of using Bayesian approaches, i.e., to enable DL models to communicate their uncertainty, for example, by comparing erroneous to correct predictions, or ID to OOD samples through visualizing the distributions of the observed uncertainty estimates, or by reporting the observed AUROC and QWK performance scores for distinct fractions of referred patients. For instance, Leibig *et al.* [170] display accuracy- and AUROC rejection-curves and provide individual AUROC scores for referral rates of 0 %, 10 %, 20 %, and 30 %, but do not measure the area under the curve that would allow a quantitative comparison. They, in addition, show the distribution of observed uncertainty estimates collected for the ID EyePACS and the OOD ImageNet dataset using KDE plots but do not provide quantitative analyses of the detection performance of OOD samples. Equivalently, Jaskari *et al.* [168] only report AUROC and QWK scores for referring 0 %, 30 %, and 50 % of the images but spare a holistic summary statistic. Similarly, Aurajo *et al.* [169] show that high uncertainty estimates occur more frequently for misclassified examples using an average uncertainty matrix. They, in addition, provide QWK-rejection-curves for the ID validation data but also do not compute the area under the curve, rendering a quantitative comparison difficult.

In contrast, Band *et al.* [167] report quantitative results for the AUARC performance and the ECE score for the rDR detection task on the EyePACS dataset, which were best for using a ResNet-50-based, approximate Bayesian model that is similar to the DE-MCD-CNN used in this work. Whereas the observed ECE of (0.02 ± 0.00) is slightly lower than for the DE-SN-MCD-SVDKL model analyzed in this work that achieves a score of

(0.038 ± 0.007), the model proposed by Band *et al.* performs very similarly w.r.t. the AUARC metric, i.e., $(97.3 \pm 0.0)\,\%$ vs. $(97.0 \pm 0.1)\,\%$. However, Band *et al.* compute the AUARC score by varying the referral threshold instead of the fraction of referred samples, which renders the AUARC score more sensitive to the model temperature [105] and limits the comparability to the AUARC results reported within this chapter.

5.5 Conclusion and outlook

Because deterministic DNNs generally do not provide calibrated aleatoric and no epistemic uncertainty information, and standard approximate BNNs also often yield insufficient uncertainty calibration, this chapter analyzed whether the promising deep kernel learning framework can improve the predictive performance and particularly, the quality of the uncertainty on the task of sDR grading and rDR detection. This method combines DNNs with GPs into an end-to-end trainable model. The conducted experiments confirm the occurrence of the feature collapse of SVDKL-based models as observed in literature and highlight the need to adapt, e.g., a fully Bayesian feature extractor into the deep kernel learning framework to obtain well-calibrated ID and OOD uncertainty estimates [44], [172]. Moreover, the findings show that using the proposed model, i.e., a fusion of the EfficientNet-B0 and the extended SVDKL approach, yields a gain in both uncertainty calibration and diagnostic predictive performance over using either of the proposed approximate Bayesian CNNs. Although the use of DEs and MCD alone already provide simple and fast improvements in uncertainty calibration, additionally deploying deep kernel learning with proper extensions and tuning can again improve those results, albeit at the expense of increased computational complexity. That is, the DKL layer adds 5.32 M parameters and about 580 MB GPU-RAM but only about 4 ms inference time on an Nvidia GeForce RTX 3090 over using the plain EfficientNet-B0, i.e., with using 1280 final features and, hence, GPs in the additive layer.

Particularly, for safety-critical medical applications, such as the DR screening and grading settings analyzed in this work, that require the DL model to have a good understanding of its limitations, the additional application of deep kernel learning turns out to be beneficial, as it provides a better near-ID detection performance through the epistemic uncertainty estimates. The

results also highlight the benefit of disentangling epistemic from aleatoric uncertainty in order to detect unfamiliar inputs. Along with the improved reliability measured through the QWK and the observed good alignment of the ID uncertainty with the occurrence of erroneous predictions, the findings show the potential of the extended, EfficientNet-B0-based SVDKL models to reduce the risk of severe misclassification and missing pathologies or diseases in a referral-based screening setting that could otherwise have been detected by a specialist. Moreover, the results suggest an additional improvement in the diagnostic performance, which remains to be analyzed in more detail on real clinical data in future research. Furthermore, improving the uncertainty calibration of SVDKL model with more sophisticated and less computationally demanding methods than using Monte Carlo dropout and deep ensembling to create fully Bayesian SVDKL models would be of great interest to enhance the practical benefits and facilitate translation into clinical application.

6 | Tranparency through concept-based explanations

Following the introduction of the two independent approaches in the previous Chapters 4 and 5 aiming to provide fine-grained segmentation masks of DR-related biomarkers and improving uncertainty calibration w.r.t. the DR severity prediction, this chapter aims to join these approaches into a transparent, inherently interpretable, uncertainty-aware, and lightweight DNN. In section 6.1, the motivation and the most important related work to this approach are discussed, followed by section 6.2.1, which provides a description of the model architecture and the experimental setup for the analysis of the proposed approach. Sections 6.3 and 6.4 conclude this chapter by presenting and discussing the results of the conducted analysis.

6.1 Motivation and related work

As discussed in more detail in Chapter 2, standard DNNs are typically black-boxes, i.e., it is difficult, if not impossible, to trace their reasoning in order to explain decisions to users and patients. Moreover, as discussed in section 2.5, the most reliable and compelling approach to build explainable DL models to date is to use ante-hoc methods, i.e., adapted DNN architectures that inherently provide explanations for their decisions rather than searching post-hoc for explanations via, e.g., saliency methods, as the latter often are uninformative, incorrect, and imprecise. A promising approach to develop such inherently interpretable DNNs is to exploit concept-based learning through the use of the CBM design [36]. As outlined in section 2.5.4, this entails training the model to extract domain-expert-defined concepts as an intermediate bottleneck representation upon which the final decision is derived. Due to constraining the prediction to be derived from this bottleneck representation, the predicted concepts can be used as an explanation for the model and its decisions. Moreover, the end-

[36] Koh *et al.*, "Concept bottleneck models" (2020)

to-end model design allows joint optimization of the concept representation and final model prediction. As a result, the CBM promises to yield a competitive performance to vanilla DNNs, while mitigating shortcut learning due to the bottleneck design [36].

The idea of the CBM design is very similar to the method proposed by Abramoff *et al.* [10]. They train a set of Alexnet [5] and VGG [61] inspired CNNs to detect the presence of DR-related lesions such as HEs, HXs, NVs in retinal color fundus images based on which two random forest classifiers are optimized to predict the presence of rDR and vtDR. Thereby, the decoupling of first detecting DR-related lesions from the final classification task, i.e., using the intermediate representation entailing information about the presence of the lesions as some kind of bottleneck, is primarily motivated to prevent the model from shortcut learning and, hence, improving model generalization.

[10] Abràmoff *et al.*, "Improved automated detection of diabetic retinopathy on a publicly available dataset through integration of deep learning" (2016)

[5] Krizhevsky *et al.*, "ImageNet classification with deep convolutional neural networks" (2012)

[61] Simonyan and Zisserman, "Very deep convolutional networks for large-scale image recognition" (2015)

Similarly, De Fauw *et al.* [30] decouple the prediction of a generic referral suggestion from the segmentation of anatomic and pathologic biomarkers related to different retinal diseases in optical coherence tomography (OCT) images in order to render the pipeline to be seamlessly adaptable to different OCT devices. To this end, they exploit a two-stage pipeline of a U-Net and a custom CNN trained for the segmentation and classification tasks, respectively. Thereby the latter is optimized on the segmentation masks derived from the pretrained segmentation model so that the classifier is effectively independent of the input. Thus, updating the segmentation model would be sufficient to adapt the pipeline to shifted input data, which facilitates generalization to, for instance, new scanner types.

[30] De Fauw *et al.*, "Clinically applicable deep learning for diagnosis and referral in retinal disease" (2018)

Quellec *et al.* [35] propose to learn a set of saliency maps for DR severity classification in an ante-hoc manner through a weakly-supervised approach to circumvent relying on post-hoc saliency methods due to their ambiguous explanations. To this end, they propose a U-Net-like encoder-decoder network including an additional, shallow classification head attached to the high-resolution output feature maps retrieved from the decoder of the network. Differing from the vanilla CBM design, these feature maps being used as explanations are not explicitly constrained to match predefined concepts such as the individual DR-related lesions, i.e., they are not trained in a supervised manner. Instead, the model is optimized by solely exploiting image-level annotations. As a result, similar to post-hoc saliency maps, their model's explanations lack the precision and tied interpretation w.r.t.

[35] Quellec *et al.*, "Explain: explanatory artificial intelligence for diabetic retinopathy diagnosis" (2021)

the presence of the DR-related pathologies as, e.g, can be seen in [35, Figure 5], which is a key aspect of the CBM design. This becomes evident by comparing the quality of the provided explanations to the target annotations in the IDRiD dataset, which at most achieve an average AUPRC score $< 20\,\%$. This is significantly inferior compared to the performance observed for the U^2-Net, e.g., in Table 4.5.

Contrasting the CBM design, Wei *et al.* [135] propose a model architecture with two parallel branches for retinal lesion segmentation and DR severity classification, respectively. They implement a U-Net-inspired segmentation model and an InceptionV3 [202] as classifier, whereby a set of attention masks derived from the lesion segmentation masks is multiplied to the feature maps of the classifier prior to applying the final linear layer of the InceptionV3. Similar to this approach, Zhou *et al.* [138] propose to use a U-Net-based segmentation model to derive lesion segmentation masks. These are then used as attention to guide another NN for DR severity grading. Although the final model prediction is derived by exploiting the lesion segmentations through the attention mechanisms, the CNN-classifiers of both methods can learn to extract features from the input image independent of the segmentation model. Hence, the final decision is not as tightly coupled to the presence of the lesions as in the CBM design impairing the expressiveness of the model's explanations.

[135] Wei *et al.*, "Learn to segment retinal lesions and beyond" (2021)

[202] Szegedy *et al.*, "Rethinking the inception architecture for computer vision" (2016)

[138] Zhou *et al.*, "Collaborative learning of semi-supervised segmentation and classification for medical images" (2019)

Inspired by the above-discussed methods, this chapter aims to investigate the feasibility of the inherently interpretable CBM and a newly proposed sCBM architecture for the task of DR severity grading from retinal color fundus images. These model designs promise to tightly couple the provided lesion segmentation masks to the derived DR severity prediction and, hence, allow exploiting the fine-grained segmentation masks to be used as an explanation for the final decision. Moreover, by adopting the lightweight U^2-Net and the uncertainty-aware SVDKL approach analyzed in the previous sections, this work aims to additionally render the CBM and sCBM architectures uncertainty-aware and lightweight. While in general both these models could be trained in an end-to-end manner, the preliminary analysis conducted in this thesis focuses on assessing the performance of an SVDKL classification head stacked upon a fixed, pre-trained U^2-Net to examine the overall feasibility of the concept-based approach leaving the evaluation of the joint model performance for future work.

6.2 Methods

In the following, first the CBM method and the relaxed sCBM design will be introduced in the context of adopting the SVDKL-framework and the U^2-Net as the model's backbone, followed by a description of the experimental setup, the model implementation, and the evaluation protocol.

6.2.1 Concept bottleneck models

To build an inherently interpretable model according to the CBM [36] architecture, the DR severity prediction has to be computed solely based on a set of predefined concepts, i.e., the final segmentation masks $\mathbf{s} \in \mathcal{S}$ produced by the U^2-Net, which will be used as segmentation backbone in this analysis. As introduced in Chapter 4, the mapping of the U^2-Net from the input image to the segmentation masks is defined as $h \circ f_{\mathbf{s}} : \mathcal{X} \to \mathcal{Z} \to \hat{\mathcal{S}} \to \mathcal{S}$, where $\mathcal{X}$, $\mathcal{Z}$, $\hat{\mathcal{S}}$, and $\mathcal{S}$ denote the input images, latent multi-scale features, saliency maps, and final segmentation masks of the U^2-Net, respectively.[13] Please refer to section 4.2.3 and Figure 4.3 for details on the U^2-Net architecture. Based on these lesion maps, a classifier $f_{\mathbf{y}} : \mathcal{S} \to \mathcal{Y}$ can be optimized to predict the DR severity grades $\mathbf{y} \in \mathcal{Y}$, i.e., the CBM is defined as

$$f_{\text{CBM}} : \mathcal{X} \overset{(h)}{\to} \mathcal{Z} \overset{(f_{\mathbf{s}})}{\to} \mathcal{S} \overset{(f_{\mathbf{y}})}{\to} \mathcal{Y} \, . \tag{6.1}$$

This design forces the classifier to derive a prediction that solely relies on the learned concepts given by the lesion segmentation masks allowing the latter to be used as an explanation for the predicted sDR grade.

[13] Note, that for convenience the saliency maps $\hat{\mathcal{S}}$ will be omitted in the following.

disadvantages of the CBM design

However, CBMs were observed to underperform in case the true concepts are not comprehensive or do not perfectly align with the main task compared to vanilla CNNs. Hence, a sufficiently comprehensive set of DR-related biomarkers is essential to achieve a sufficient diagnostic performance. This would have to particularly comprise lesions related to more severe stages of DR in addition to the lesions entailed within the IDRiD dataset. That is, a dataset including additional annotations for pathologic areas affected, e.g., by NVs, VB, and IRMA is expected to be essential to achieve the most precise DR severity grading system [38]. Moreover, solely relying on the fixed set of concepts, i.e., the presence of the DR-related lesions, might impair the detection of, e.g., unknown pathologies that do not align with the predefined concepts or images with insufficient quality

[38] Wilkinson *et al.*, "Proposed international clinical diabetic retinopathy and diabetic macular edema disease severity scales" (2003)

as the corresponding variations in the input image might be suppressed by the bottleneck due to being irrelevant w.r.t. the actual segmentation. consequently, it might not be uniquely conclusive whether an, e.g., empty set of segmentation masks indicates a healthy sample, i.e., no lesions are present in the image, or an unfamiliar OOD input for which the segmentation model fails to predict meaningful segmentation masks, which is highly undesirable w.r.t. patient security. Whereas the latter might to some extent be manageable by a full Bayesian modeling of the segmentation model, the concept incompleteness is still expected to limit the diagnostic performance.

soft-concept bottleneck model

Hence, particularly in the case of insufficiently annotated data, slightly relaxing the bottleneck constraint provides a suitable alternative. This can be achieved by allowing the classifier to access the latent multi-scale features $\mathbf{z} \in \mathcal{Z}$ of the U^2-Net on which the final segmentation masks are derived — similar to the underlying idea of the concept-learning framework proposed by Sarkar *et al.* [37]. That is, to optimize a shallow classifier $f_{\mathbf{y}} : \mathcal{Z} \to \mathcal{Y}$ to derive the prediction for the DR severity grade. With this, the sCBM design is defined as

[37] Sarkar *et al.*, "A framework for learning ante-hoc explainable models via concepts" (2022)

$$f_{\text{sCBM}} : \; \mathcal{X} \overset{(h)}{\to} \mathcal{Z} \overset{(f_{\mathbf{s}})}{\underset{(f_{\mathbf{y}})}{\lessgtr}} \begin{matrix} \mathcal{S} \\ \mathcal{Y} \end{matrix} . \tag{6.2}$$

Due to the deep supervision applied to the saliency maps $\hat{\mathbf{s}} \in \hat{\mathcal{S}}$ and the shallow mapping $f_{\mathbf{s}} : \mathcal{Z} \to \hat{\mathcal{S}} \to \mathcal{S}$ solely consists of two sequential layers of 1×1 convolutions, the latent features $\mathbf{z}$ can be viewed as a set of visually-interpretable and weakly-supervised soft concepts. Moreover, the hard sharing of these soft concepts between $f_{\mathbf{s}}$ and $f_{\mathbf{y}}$ ensures that the segmentation masks $\mathbf{s} \in \mathcal{S}$ could still be used as explanations by means of visualizing present biomarkers that contributed to the predicted DR severity score, albeit with a slightly weakened class-concept relation. That is, the expressiveness of the provided explanation would be diminished in case the sCBM learns to encode some soft concepts $\mathbf{z}_i \in \mathbf{z}$ that do not contribute to the segmentation of the set of biomarkers but are important markers for a precise classification of the DR severity. However, this additional flexibility would in turn not only improve the diagnostic performance but also enable the classifier $f_{\mathbf{y}}$ to better recognize unfamiliar inputs and communicate their occurrence by means of increased predictive uncertainty.

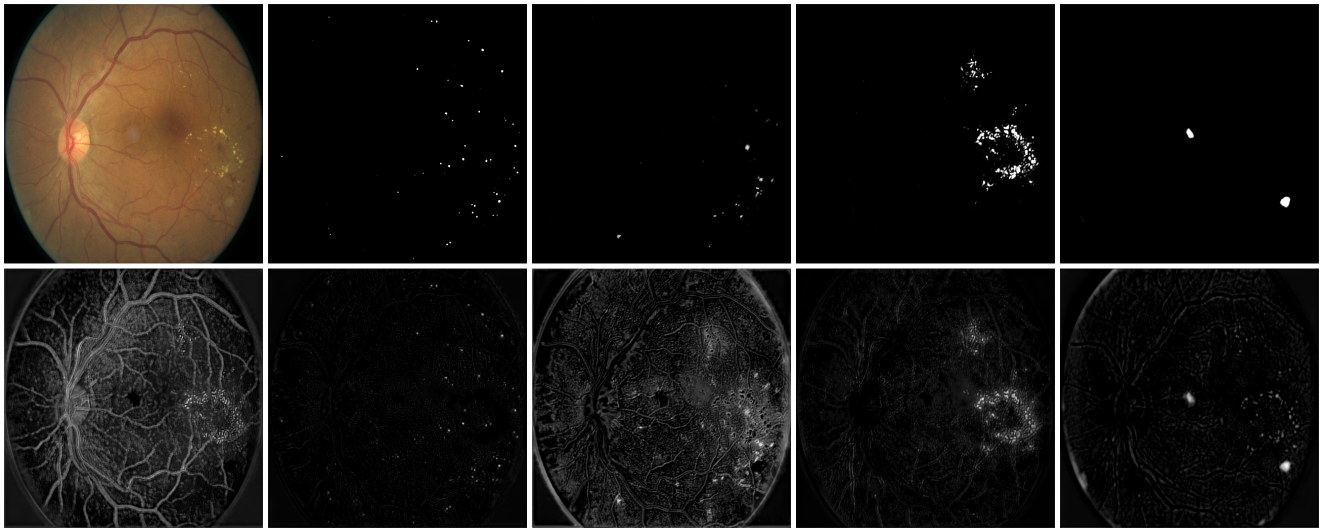


FIGURE 6.1: Example image from the EyePACS [140] dataset along with the extracted segmentation masks $\mathbf{s} \in \mathcal{S}$ and an excerpt from the multi-scale feature maps $\mathbf{z} \in \mathcal{Z}$. The top row shows the input image, the MA, HE, HX, and CWS segmentation masks from left to right. The bottom row shows six selected feature maps contained in $\mathbf{z}$.

6.2.2 Study data and preprocessing

For the analysis conducted in this chapter, the EyePACS dataset is used for both model training and evaluation. First, the samples of the dataset are cropped to a minimum-sized square containing the visible retinal disc as described in section 5.2.2. Afterwards, the images are preprocessed following the procedure described in section 4.2.2, i.e., images are resized to a resolution of 512×512 pixels prior to applying CLAHE and image standardization. Finally, both the segmentation masks $\mathbf{s} \in \mathcal{S}$ and multi-scale feature maps $\mathbf{z} \in \mathcal{Z}$ are extracted for all samples comprised in the EyePACS dataset. They are obtained by feeding the images through each of the ten U^2-Net-S models used in multi-task mode, which in Chapter 4 were trained on the IDRiD dataset. Figure 6.1 exemplarily shows the segmentation masks along with a selected excerpt of the multi-scale feature maps for an image of the EyePACS dataset retrieved from one of the pre-trained U^2-Net-S models. No additional data augmentation is applied to the extracted segmentation masks and feature maps for model training.

To enrich the feature representation, the segmentation masks of all four lesions are both globally average- and max-pooled and concatenated to a single feature vector $s' \in \mathcal{S}' \subset \mathbb{R}^{N \times 8}$. The multi-scale feature maps are max-pooled and concatenated to a low-resolution feature vector $z' \in \mathcal{Z}' \subset$

$\mathbb{R}^{N\times 192\times 16\times 16}$. Thereby, the spatial resolution corresponds to the deepest feature maps retrieved from the bottleneck RSU block in the U^2-Net-S for an input image of resolution 512×512. This leads to each $i = 1, \ldots, 10$ datasets

$$\mathcal{D}^{(i)}_{\text{EyePACS}\to\mathcal{S}'} \quad \text{and} \quad \mathcal{D}^{(i)}_{\text{EyePACS}\to\mathcal{Z}'} \tag{6.3}$$

for the set comprising the pooled segmentation masks and the multi-scale feature maps, respectively. Each of these datasets is split into a training, validation, and test set adhering to the original data split of the EyePACS dataset as described in section 5.2.1.

6.2.3 Experimental setup

Implementation As for the analyses conducted in Chapters 4 and 5, all models are implemented and trained using the PyTorch [149] and GPyTorch [185] libraries. As a baseline, a CBM with a simple logistic regressor is implemented, i.e. a fully connected layer with subsequent softmax activation is applied to the resolution-compressed features obtained from the segmentation masks, i.e., the $\mathcal{D}_{\text{EyePACS}\to\mathcal{S}'}$ dataset. In line with the experiments in the previous chapter on improving the uncertainty calibration and awareness, an SVDKL-CBM is set up. Thereby, the shallow classifier is chosen as the additive GP layer as introduced in section 5.2.4 with $J = 8$ GPs. Note that this joint model configuration equals a SVDKL model w.r.t. the DR grading and, hence, will referred to as SVDKL-CBM. Equally to the experimental setup of the previous chapter, the CBM and the SVDKL-CBM are optimized using the NLL and approximate ELBO objectives introduced in (5.19) and (5.20), respectively.

[149] Paszke *et al.*, "PyTorch: an imperative style, high-performance deep learning library" (2019)

[185] Gardner *et al.*, "GPyTorch: blackbox matrix-matrix Gaussian process inference with GPU acceleration" (2018)

(CBM)

(SVDKL-CBM)

The sCBM utilizes the $\mathcal{D}_{\text{EyePACS}\to\mathcal{Z}'}$ dataset. To allow the model to extract meaningful features and exploit spatial information from the low-resolution feature maps, firstly, a shallow CNN is implemented. This network consists of two sequentially applied MBC blocks followed by a 1×1 convolution, which throughout keeps the number of features identical to that of the input features, i.e., no feature expansion or reduction is applied. Afterwards — equally to the implementation of the CBM — a baseline sCBM and a SVDKL-sCBM model are obtained by optimizing a logistic regressor and an additive GP layer with $J = 192$ GPs upon the globally average-pooled output features of this shallow network, respectively.

(sCBM)

(SVDKL-sCBM)

All GPs are set up as in section 5.2.7, i.e., with using 64 inducing points and grid bounds set to $(-10, 10)$. Following the observations of the previous chapter, MCD is applied for the SVDKL-sCBM to learn a distance-preserving feature representation within the shallow network and retain a good uncertainty awareness. To this end, stochastic depth as described in section 5.2.6 is applied to the residual connections of the MBC blocks.

Training Closely in line with the experiments described in section 5.2.8, all model instances are optimized using the Adam optimizer, PyTorch's automatic mixed precision package, and by balancing the class distribution via randomly sampling the classes in the training set with probability corresponding to their inverse relative class frequency. All training hyperparameters were chosen based on a simple, manual grid search. For each dataset $\mathcal{D}^{(i)}_{\text{EyePACS}\to\mathcal{S}'}$ obtained from one of the different U^2-Net-S models, a single CBM and SVDKL-CBM is trained, respectively, resulting in a total of ten trials per model. Equally, for each set $\mathcal{D}^{(i)}_{\text{EyePACS}\to\mathcal{Z}'}$ a single sCBM and SVDKL-sCBM is optimized.

CBM and SVDKL-CBM training

Both the CBM and SVDKL-CBM are trained for a maximum number of 80 epochs with a batch size of 2048 samples. After ten initial epochs of regular training, early stopping is used to terminate model optimization after 20 epochs without improvement w.r.t. the validation performance measured through the AUROC-macro on the EyePACS's public test dataset. A learning rate of 1e−3 is used for the vanilla CBM whereas the variational- and the kernel hyperparameters of the SVDKL-CBM's GP layer were updated using a learning rate of 1e−2 and 1e−4, respectively. A plateau scheduler is implemented to halve the learning rate after four epochs without improvement w.r.t. the performance on the validation data. L2-norm regularization is applied with a decay weight of 1e−4.

sCBM and SVDKL-sCBM training

Due to the additional complexity of the classifier heads resulting from the shallow networks, the maximum number of epochs is increased to 100 iterations to train the sCBMs. Stochastic depth was applied with a survival probability of 90 % for the SVDKL-sCBM while drawing $T = 1$, 32, and 32 MCD samples during training, validation, and testing, respectively. The learning rate of the variational- and the kernel hyperparameters of the GP layer were changed to 1e−3 and 5e−4, respectively. Both the shallow network and the additive GP layer were trained jointly from scratch without pretraining the weights of the former.

Evaluation In this chapter, model performance is evaluated equally to the ID diagnostic performance and uncertainty analysis conducted in Chapter 5 for the sDR grading task using the EyePACS test data, i.e., computing the metrics $\mathcal{M}_{\text{ID, P}}$ and $\mathcal{M}_{\text{ID, U}}$ as defined in (5.23) and (5.26).

Evaluating the quality of the provided explanations is a challenging task. One approach proposed by Sarkar *et al.* is to measure the *faithfulness* as well as the *explanation error* of the concepts. They define faithfulness as the classification performance based on an auxiliary classifier directly applied to the concepts. Hence, the performance of the CBM architectures can be used as an approximation to the faithfulness of the provided concepts by means of the lesion segmentation masks. Sarkar *et al.* furthermore define the explanation error as the deviation of the predicted concepts to the ground truth, i.e., the precision of the lesion segmentations w.r.t. the target annotations. As target annotations only exist for the IDRiD dataset on which the U^2-Net-S models were trained, estimating the explanation error for the EyePACS data within this preliminary analysis is not feasible. However, the evaluation conducted in Chapter 4 can be viewed as a proxy for the explanation error and, therefore, will not be discussed again herein.

6.3 Results

The results of this chapter's analysis are displayed in Table 6.1, which shows the average results of the observed diagnostic performance and ID uncertainty calibration across the ten trials for each of the CBMs and sCBMs. For comparison, the table is augmented by the performance observed for the baseline vanilla CNN of Chapter 5, i.e. the EfficientNet-B0.

CBM

Regarding the CBM, an overall low performance is observed throughout the applied metrics indicating a limited faithfulness of the explanations given by the final segmentation masks. In particular, the observed accuracy and QWK of 30.0 % and 12.1 % indicate a diagnostic performance that is only slightly superior to randomly guessing the associated class labels. Regarding the ECE score, the CBM marginally outperforms the baseline CNN indicating that, despite the low diagnostic performance, the logistic regressor predicts with a comparably well-calibrated confidence. However, as observed from the low AUARC score, the model's uncertainty estimates do not align very well with the model's prediction errors.

TABLE 6.1: Performance comparison of the CBMs and sCBMs as introduced in section 6.2.1 to the vanilla CNN (EfficientNet-B0) of Chapter 5 for the sDR grading task evaluated on the EyePACS test set showing the mean and standard deviation per metric.

	AUROC [%] ↑	ACC [%] ↑	QWK [%] ↑	NLL ↓	ECE ↓	AUARC [%] ↑	
						total	aleatoric
CNN	82.5 (0.3)	63.7 (5.4)	61.9 (1.5)	0.912 (0.036)	**0.105 (0.045)**	75.5 (4.5)	-
CBM	61.3 (4.4)	30.0 (8.2)	12.1 (5.5)	1.560 (0.040)	**0.099 (0.027)**	38.3 (12.2)	-
SVDKL-CBM	**67.6 (1.2)**	**47.7 (4.8)**	**22.1 (3.2)**	**1.403 (0.020)**	0.146 (0.043)	**61.2 (3.2)**	61.3 (3.2)
sCBM	79.1 (0.7)	56.8 (3.3)	47.8 (2.2)	1.056 (0.037)	**0.071 (0.021)**	64.3 (3.5)	-
SVDKL-sCBM	**80.1 (0.7)**	**60.6 (5.3)**	**50.4 (3.1)**	**1.033 (0.050)**	0.135 (0.041)	**69.4 (4.5)**	67.5 (4.7)

SVDKL-CBM

In contrast to the CBM, exploiting the additive GP layer instead of the logistic regressor, i.e., utilizing the SVDKL-CBM, shows a significant improvement in the diagnostic performance. This particularly affects the predictive accuracy and the QWK score indicating a significantly improved alignment of the model's predictions with the true sDR grades. Contrasting the observations made for utilizing the vanilla CBM, this suggests a higher faithfulness of the concepts. Moreover, the AUARC score observed for the SVDKL-CBM outperforms that of the CBM by about 59.8 % and, with this, reduces the performance gap to that of the baseline CNN model to a relative performance difference of 18.9 %. This demonstrates a significantly improved alignment of the predictive uncertainty with the model's errors — again highlighting the benefit of the additive GP layer as observed in the previous chapter. Nonetheless, the overall diagnostic performance with accuracy and QWK scores around 50 % and 22 %, respectively, implies that the model still makes many errors — including several severe misclassifications.

sCBM

Directly exploiting the soft concepts by means of the sCBM again significantly improves the measured diagnostic performance. As a result, the sCBM shows an AUROC score of about 79.1 % indicating a fair diagnostic performance. Particularly, the QWK measure benefits from the sCBM design by more than doubling the observed score compared to the SVDKL-CBM. This indicates a significantly improved alignment of the model's predictions with the true sDR grades, i.e., fewer, and less severe misclassifications are made by the model. Still, the diagnostic performance is not on par with the baseline CNN. Regarding the uncertainty calibration, also the AUARC significantly improved compared to the vanilla CBM, and the ECE is observed to even outperform the baseline CNN.

SVDKL-sCBM

By additionally exploiting the deep kernel learning approach in combination with the sCBM design, the diagnostic performance and uncertainty calibration w.r.t. the AUARC are again observed to improve. This results in AUROC and accuracy scores, which are almost on par ($< 3\,\%$ and $< 5\,\%$, respectively) with the baseline CNN. Notably, the QWK score is observed to suffer the most compared to the vanilla CNN, which can learn features for the sDR grading task directly from the high-resolution input images and is not constrained to the fixed set of (soft) concepts as the CBM and sCBM architectures are.

effect of SVDKL

Comparing the SVDKL-based CBMs to their non-Bayesian counterparts reflects the observations of the analysis conducted in the previous chapter: On the one hand, the SVDKL-based models exhibit a higher calibration error for the sDR grading task on the ID EyePACS dataset. On the other hand, the additive GP layer significantly enhances both the diagnostic performance with regard to predictive accuracy and reliability, as well as uncertainty-calibration, as measured through the AUARC, nearly attaining the same AUROC and accuracy as the vanilla CNN.

6.4 Discussion

CBM design

As expected, the faithfulness of the concepts, i.e., the diagnostic performance of the CBM, was observed to be overall insufficient for practical use. This is assumed to be mainly caused by the incompleteness of the considered biomarkers, i.e., the U^2-Net models were solely trained to predict the presence of MAs, HEs, HXs, as well as CWSs and, hence, the concepts entail no information about other DR-related lesions such as NVs. This limits a precise differentiation of the individual DR severity levels, particularly for more severe stages. Followingly, extending the set of considered biomarkers for the segmentation task, i.e., exploiting a more comprehensive set of DR-related biomarkers, is expected to significantly improve the diagnostic performance of the CBMs and, hence, the faithfulness of the model's explanations.

Another possible reason for this low performance is the high compression by means of the global pooling applied to the segmentation masks. Thus, the compressed masks $\mathcal{S}'$ no longer entail any spatial information. This is likely to hamper the differentiation of the individual DR severity levels, as information, e.g., about the number of quadrants affected by

retinal hemorrhages is lost, which however could be important to differentiate severity levels two and three according to the ICDR grading scheme provided in Table 3.1. Hence, extracting more sophisticated features in addition to the global average and max pooling of the feature maps would likely improve the predictive performance of the CBM. For instance, the biomarker-specific lesion count and affected area per quadrant in the retina as well as the average lesion size could provide a benefit for the grading performance.

Moreover, for convenience, the U^2-Net-S used in multi-task mode was adopted for this preliminary analysis, which itself showed inferior segmentation performance compared to, e.g., the U^2-Net-S used in mixed-task setting. Thus, replacing the segmentation model with one of the superior variants proposed in Chapter 4 might additionally improve the DR severity grading performance.

sCBM design

In contrast to the overall low performance observed for utilizing the CBM design, relaxing the bottleneck constraint by means of the sCBM lead to a significant improvement of the diagnostic performance. This large performance difference, on the one hand, is likely caused by the fact that the soft concepts entail more holistic and expressive retinal features w.r.t. the DR severity grading task. On the other hand, the applied classifier of the sCBM architecture is more flexible. Hence, evaluating the effect of increasing the capacity of the CBM's classifier and preserving at least some spatial information in the concept features as done for the sCBM design on the diagnostic performance would be of great interest for future work.

Despite still being slightly inferior compared to the vanilla CNN and, hence, also to the SVDKL-based EfficientNet models analyzed in Chapter 5, the observed diagnostic performance of the SVDKL-sCBM is very promising. In particular regarding that (a) the U^2-Nets were solely trained on 43 images of the high-quality IDRiD dataset, (b) the EyePACS image data is very noisy and, thus, poses a significant distribution shift to the pretrained segmentation models, and (c) the U^2-Nets were not adapted for the additional DR severity grading task by joint training, the observed diagnostic performance can be considered comparably good.

joint, end-to-end training

Given these limitations, both the diagnostic performance and quality of the lesion segmentation are expected to significantly benefit from a joint, end-to-end training of both the CBM and sCBM, which exploits image data and annotations from both the DR grading and lesion segmentation

domain. That is, a comprehensive concept representation could ease solving the classification task and improve model generalization. Similarly, the supervision by the image-level targets can improve the detection or suppression of severity-related biomarkers as pixel-level annotations often are scarce and noisy and some lesions are rarely encountered within the often small-sized segmentation datasets.

In particular, the latter could be explicitly enforced by means of weak supervision, e.g., by introducing an auxiliary loss that penalizes the occurrence or absence of specific lesions depending on image-level severity grades. That is, if the retinal image was, for instance, classified as less than PDR (< 4) but more than no DR (> 0) the segmentation masks for NVs should be empty everywhere and that of the MAs not. Equivalently, when being classified as having no DR ($= 0$), none of the predicted lesions should be present anywhere. Aligning this idea, the occlusion loss proposed by Quellec *et al.* [35] promises a convenient way to improve the concept representation given by the segmentation masks. It is computed based on an augmented input image, which is obtained by occluding the image at the location where the model predicted the presence of any lesion, and accordingly updating the DR severity grade target to having no DR. Particularly if lacking a comprehensive set of labeled biomarkers, this occlusion loss in combination with the sparsity enforcing L1-norm loss — equally proposed by Quellec *et al.* [35] — might help to build a more comprehensive concept representation entailing a mixture of fully- and weakly-supervised concepts. Moreover, as already discussed in section 4.3.4, adversarial supervision as proposed by Zhou *et al.* [138] could be exploited. That is, to train an auxiliary discriminator network that has to predict whether the given segmentation mask is derived from a database entailing pixel-level annotations or not. Analyzing the effect of such an end-to-end training applied to both the CBM and sCBM would be of great interest for future work and promises to yield another improvement in the diagnostic performance.

class-concept relation in sCBMs

However, as briefly motivated in section 6.2.1, end-to-end training of the sCBM might result in learning features unrelated to the actual concepts used to explain the model's decision and, hence, weaken the class-concept relation. Consequently, it may be necessary to restrict the flexibility of the sCBM's classifier in order to mitigate the learning of shortcuts from the model input to the DR severity prediction, i.e., the divergence of the soft concepts to the final, true segmented biomarkers. This could for instance

be achieved by penalizing the weights of the shallow mapping f_s to be close to zero, thereby enforcing the model to utilize all soft concepts for the lesion prediction. Another potential solution would be to either lower the dimension of the soft concept representation of the U^2-Net or to use a subset of only the most relevant soft concepts w.r.t. the lesion segmentation as input to the classification head. This should result in the final segmentation masks providing well-aligned explanations for the given prediction, largely preserving the class-concept relationship of the sCBM. In addition, due to both the deep supervision and the single-layer mapping from the soft concepts to the final lesion segmentation masks, the former could be viewed as interpretable mostly spatially resolved features that could provide additional insights into the model's decision-making process. Nonetheless, given either a comprehensive or augmented set of biomarkers exploiting the above-discussed weak and self-supervised approaches, the CBM would be preferable to achieve optimal coupling of the explanations and the DR severity grade predictions. This tight coupling of the explanations and the predictions in both the CBM and sCBM design is a major advantage over other explainability approaches, which — as will be discussed in more detail in section 7.2 — allows for human interventions on the explanations and to manually probe and correct the given prediction.

explanation quality

Despite improving the concept completeness and, with this, the quality of the provided explanations is indisputably beneficial, assessing particularly the practical benefit of the explanations remains difficult and open for future work. Whereas, e.g., the proposed faithfulness metric of Sarkar *et al.* [37] seems to be appealing in order to quantify the quality of explanations, it is highly dependent on the applied classifier as becomes obvious from the differing faithfulness estimates obtained for either using the SVDKL-based or the simple logistic regressor within the CBM design. In addition, the ability to visualize the affected regions itself is expected to provide a substantial information gain for the user. Specifically, it enables users to verify the plausibility of the prediction, challenge the system, and detect potential failures of the screening system. For instance, when the classifier predicts the absence of DR but the segmentation masks indicate the presence of the disease. Ultimately, evaluating the benefit of the provided explanations would be best assessed through a user study, which, however, is beyond the scope of this thesis.

uncertainty calibration

Similar results as for the previous Chapter 5 are obtained in this analysis regarding the effect of the SVDKL framework on the uncertainty calibration. That is, the uncertainty estimates are observed to be better aligned with the occurrence of errors compared to the non-Bayesian CBMs, i.e. the AUARC performance is superior. However, the ECE of the SVDKL-based CBM and sCBM observed on the ID EyePACS data is higher. Analyzing whether the calibration error of the SVDKL-CBM and -sCBM equally improves w.r.t. the data and task shift settings as analyzed in the previous section, and why this phenomenon occurs would be of great interest for future work.

computational complexity

Due to its efficient sampling scheme, the runtime of the SVDKL-based models increases only marginally over using a vanilla CNN. Furthermore, compared to the SVDKL-based EfficientNet-B0 proposed in Chapter 5, the additional number of parameters introduced by the GP layer in both the proposed SVDKL-CBM and -sCBM is significantly lower, due to the much smaller feature vector dimension upon which the layer is applied. As a result, the number of additional parameters introduced by using the GP-based over the vanilla CBM and sCBM is limited to 0.03 M and 0.80 M parameters, respectively. Taking the parameters of the U^2-Net-S backbone into account, the SVDKL-CBM and -sCBM in total comprise about 0.32 M and 1.23 M parameters, respectively. For comparison, the vanilla EfficientNet-B0 comprises 4.01 M parameters, and additionally exploiting the SVDKL framework again adds 5.32 M parameters. Thereby, the memory requirement of the SVDKL-CBM and -sCBM required for prediction sums up to about 303 MB and 410 MB in total while being primarily dominated by the memory usage of the U^2-Net-S. With this, the SVDKL-based CBM and sCBM have fewer parameters than both the vanilla (4.01 M) and SVDKL-based (9.34 M) EfficientNet-B0 while fairly increasing the memory consumption over the former (182 MB) and being less demanding than the latter (762 MB).

6.5 Conclusion and outlook

Summarizing the results of the herein conducted analysis, the proposed SVDKL-sCBM architecture shows to provide a promising diagnostic performance despite the limited set of concepts and without full end-to-end model training, which is close to that of the vanilla EfficientNet-B0. Thereby, the

model (a) inherently provides explanations for the model's decision in the form of detailed segmentation masks of the most important DR-related lesions, (b) improves the diagnostic performance and the detection of erroneous predictions based on the model's uncertainty over the non-Bayesian, vanilla sCBM, and (c) has a reasonable computational complexity close to that of the EfficientNet-B0 promising straightforward mobile and edge-compatible application.

Further experiments have to be conducted to verify whether the above-discussed end-to-end model optimization of both the SVDKL-CBM and -sCBM can yield the expected performance improvements, close the remaining gap compared to the fully-extended SVDKL models particularly regarding the reliability measured through the QWK metric, and the benefits of the deep kernel learning framework over the standard BNNs observed for particularly the near-OOD uncertainty calibration in Chapter 5 will translate to the CBM and sCBM design.

7 | Conclusion and outlook

DR screening is expected to impose a severely increasing burden in the near future on secondary and tertiary medical care due to the increasing patient population at risk. Hence, there is great interest in exploiting DL to shift the task of DR screening to primary care. However, as discussed in the introduction in Chapter 1 of this thesis, the lack of transparency and explainability of vanilla DNNs has been identified as a confounding limit to trust, acceptance, and the adoption of DL-based systems in clinical practice, which creates a significant societal demand for the development and deployment of responsible AI [11]. Furthermore, future AI tools in the high-risk medical field will be required to not only comply with the EU GDPR [17] and MDR [16] but also the recently passed EU AI Act [15] — a risk-based regulatory guideline fostering the development of safe and trustworthy AI. These explicitly demand transparency and explainability to inform individuals about the reasoning behind the provided decision, to detect potential model failures, to enable informed human oversight, and allow challenging and inspecting the AI system's reasoning, which is impossible for using black-box DNNs.

[11] Petersen *et al.*, "Responsible and regulatory conform machine learning for medicine: a survey of challenges and solutions" (2022)

[17] European Parliament, Council of the European Union, "General Data Protection Regulation" (2016)

[16] European Parliament, Council of the European Union, "Medical Device Regulation" (2023)

[15] European Commission, Directorate-General for Communications Networks, Content and Technology, "Artificial Intelligence Act" (2021)

thesis contributions

To comply with this regulatory and societary demand for deploying transparent and responsible AI and to allow secure use, the objective of this thesis is to narrow the existing gap for the translation of DL-based diagnostic systems such as DR screening into clinical practice. A lightweight, end-to-end trainable, transparent, and inherently interpretable DL-based system for mobile use is proposed, aiming to mitigate these shortcomings. Aligning the state-of-the-art, this system is conceived to incorporate domain knowledge into the DL model architecture to partially open the black box instead of relying on vanilla, opaque DNNs. With this, an explainable gray-box approach is created that exploits task-related concepts as an intermediate representation — based on which the final classification is derived along with a trustworthy uncertainty estimate.

In detail, the model relies on generating fine-grained segmentations of the

DR-related biomarkers using a lightweight U-shaped backbone CNN from which the progression of the disease can be derived. This additionally allows for mobile application and precise visualization of the presence, frequency, and locations of the disease-related biomarkers enabling users to sanity-check or challenge the predictions of the algorithm. Due to the reliance of the predicted disease on the present biomarkers, the model is explainable by design aligning the demand for human oversight stipulated, e.g., by the EU AI Act. By additionally exploiting Bayesian DL, the predictive transparency of the method by means of high-quality uncertainty estimates is targeted to be improved. This allows for assessing the reliability of the given prediction and informed human supervision promising to help detect model failures.

This thesis examines these promising methods for their suitability to implement the proposed lightweight, transparent DR screening system that provides trustworthy and informative uncertainty estimates. Nonetheless, the approach is expected to be readily applicable to similar tasks that follow a comparable imaging-based grading scheme, such as the classification of age-related macular degeneration and hypertensive retinopathy. The following section 7.1 will summarize and discuss the proposed methodology and the results of the analyses conducted in this thesis in order to address the aforementioned challenges. Finally, section 7.2 provides an outlook on further steps to be taken to translate the proposed system into clinical practice and on interesting future research directions.

7.1 Summary and discussion

lightweight lesion segmentation

To derive the required fine-grained segmentation on low-performance hardware allowing for mobile use of the system, a promising CNN architecture, i.e., the U^2-Net [39], was adapted in Chapter 4 by means of feature scaling and deploying DCs. It was analyzed for its usability to yield precise segmentations while keeping computational complexity low and searching for the optimal task design.

[39] Qin *et al.*, "U^2-Net: going deeper with nested U-structure for salient object detection" (2020)

Thereby, the U^2-Net-S and U^2-Net-M model variants, which comprise significantly fewer parameters and computational operations, were observed to yield a good trade-off between segmentation precision measured through the AUPRC and computational complexity, when deployed in dual-task configuration, i.e., for simultaneously segmenting MAs and HEs as well as

HXs and CWSs. Moreover, exploiting model ensembling and a mixed-task configuration, i.e., deploying a single-task model for both HX and CWS as well as a dual-task model for MA and HE segmentation, showed to again yield a boost in performance that is on par with the state-of-the-art literature — particularly outperforming the CWS segmentation — while significantly reducing the computational complexity compared to both the original U-Net and U^2-Net.

Although the performance of the MA segmentation was not able to match that of the literature w.r.t. the AUPRC, the HD revealed the U^2-Net's MA segmentation to be superior to that of the HEs, which, in turn, were observed to be on par with the literature. This highlights the necessity to evaluate the model performance in more detail in future work, e.g., by computing lesion-instance-level metrics for the detection precision or by a qualitative validation of the predicted segmentations through experts, rather than solely relying on pixel-level-dependent metrics such as the AUPRC score.

Whereas the use of DCs as implemented in this work did not yield a compelling reduction of the computational cost w.r.t. the observed associated decrease in segmentation performance, further optimization of the model architecture, e.g., by using MBCs [151], and training process as discussed in more detail in section 4.3.4 could enhance upon the already achieved tradeoff improvements through the applied feature scaling. Moreover, the real-world applicability to low-performance hardware has to be analyzed in future work by means of usability tests as well as analyzing runtime latency, energy consumption, and memory requirements on edge-device hardware. Nonetheless, the U^2-Net architecture promises to be a suitable foundational building block to be used within a lightweight DR grading system that follows the proposed transparent, interpretable model design.

[151] Tan and Le, "EfficientNet: rethinking model scaling for convolutional neural networks" (2019)

DR grading with calibrated uncertainty

To furthermore improve the predictive transparency of this model design, Chapter 5 analyzed the use of BNNs to provide reliable uncertainty estimates for the task of DR severity grading, which could provide insights about difficult decisions or unknown operating conditions and be seamlessly integrated into the model architecture. The analysis thereby focussed on the promising deep kernel learning [42] approach and its effect on the predictive performance and uncertainty calibration analyzed for ID and OOD as well as task-related and -unrelated data. The deep kernel learning framework combines CNNs and nonparametric GPs, which are frequently used

[42] Wilson *et al.*, "Deep kernel learning" (2016)

as gold standard for uncertainty quantification, into a single, end-to-end trainable model and promises to provide both good feature extraction and high-fidelity uncertainty estimates. Due to the non-Gaussian posterior resulting from the categorical likelihood function, i.e., the nonlinear softmax activation required to compute the class probabilities for the DR severity grades, SVDKL [171] was adapted for this task, which derives a scalable, variational approximation to the intractable posterior by exploiting sparse GPs and VI.

[171] Wilson *et al.*, "Stochastic variational deep kernel learning" (2016)

Furthermore, the occurrence and potential solutions to the feature collapse recently observed in literature, i.e., the susceptibility of the SVDKL's feature extractor to over-correlate the training samples leading to a collapse of the latent features and, hence, uncertainty estimates, were examined within this thesis. As fully Bayesian modeling was shown to prevent this collapse [44], [172], the SVDKL model's feature extractor was rendered approximately Bayesian by building DEs and applying MCD. In addition, the effect of deploying a regularization through limiting the spectral norm of the feature extractor was analyzed, which was recently shown to mitigate this issue while promising a sample and memory-efficient solution [173].

[44] Tran *et al.*, "Calibrating deep convolutional gaussian processes" (2019)

[172] Ober *et al.*, "The promises and pitfalls of deep kernel learning" (2021)

[173] Liu *et al.*, "A simple approach to improve single-model deep uncertainty via distance-awareness" (2023)

The results of the analysis confirmed the occurrence of the feature collapse in the proposed EfficientNet-based [151] SVDKL model trained for DR severity grading and showed the extensions, particularly, applying MCD and building DEs to consistently improve the SVDKL's uncertainty calibration. Expectedly, both the SVDKL models as well as the baseline BNNs analyzed in this work were observed to not only improve the predictive performance but also the awareness for difficult ID samples and the occurrence of OOD and task-unrelated inputs. Thereby, the ablation analysis showed that both deploying MCD and building DEs were found to provide the largest benefit for uncertainty calibration across the baseline BNN approximations and the SVDKL models. Nonetheless, particularly the fully Bayesian SVDKL models were found to provide the best awareness of their knowledge and limitations w.r.t. the detection of task-related but unfamiliar samples, i.e., OOD retinal images that comprise other diseases than DR.

Whereas the ability to detect such potential failures, e.g., that the patient might correctly receive a negative diagnosis for DR but suffers from a different disease, which would remain undetected, is a very important first step, determining and validating a clinically viable threshold that balances

false positive and negative predictions, as well as the number of rejections depending on the associated risks to filter for these failures is a key challenge beyond the scope of this work. Moreover, neither of the BNNs, including the SVDKL-based models, analyzed in this work provided optimal uncertainty calibration. That is, despite the improved model transparency observed for the extended SVDKL model by means of more reliably communicating predictive uncertainty, implementing rigorous OOD tests beyond those deployed in this thesis and, if necessary, targeting determined issues would be required prior to clinical use to ensure functional safety and determining specific circumstances for which the model could not be trusted [11]. Nonetheless, the analysis of this chapter showed the SVDKL approach to provide a promising framework that could improve uncertainty calibration over standard approximate BNNs when taking care of the feature collapse, albeit at the expense of increased computational complexity.

transparent, explainable, and lightweight DR screening

Both analyses conducted in Chapter 4 and Chapter 5 highlighted the suitability of the U^2-Net and SVDKL framework to be used as basic building blocks within the transparent, explainable, and lightweight screening model as proposed in this thesis. In Chapter 6, a preliminary analysis of the performance of this joint model according to the CBMs [36] was conducted. The CBM architecture is set up to learn a set of concepts as an intermediate bottleneck representation of the CNN backbone upon which the final decision is derived. The key idea of this design is that these concepts inherently provide explanations for the model's decisions and, hence, render the model interpretable.

[36] Koh *et al.*, "Concept bottleneck models" (2020)

In this thesis, the lesion segmentation masks derived from the pretrained U^2-Net-S models adopted from Chapter 4 were used as bottleneck concepts based on which a SVDKL-based classifier was trained to predict the DR severity grades. However, due to the incompleteness of the considered biomarkers in the segmentation model, which is expected to limit the diagnostic performance of the CBM, the sCBM was proposed, similar to the idea of [37]. Within this architecture the bottleneck constraint is slightly relaxed, i.e., the the multi-scale feature maps of the U^2-Net are utilized as input for the classifier instead of the final segmentation masks, as these are expected to entail more comprehensive retinal features.

[37] Sarkar *et al.*, "A framework for learning ante-hoc explainable models via concepts" (2022)

As expected, the diagnostic performance of this SVDKL-CBM was observed to be insufficient for practical use. However, the relaxation to the SVDKL-sCBM design was observed to add a significant benefit to the di-

agnostic performance, whereby it promises to retain a strong class-concept relation when sufficiently regularizing the model and, hence, the expressiveness of the explanations. Thereby, the diagnostic performance was observed to almost being on par with the EfficientNet-B0-based CNN of Chapter 5 despite the limited set of concepts, the distribution shift from the IDRiD to the EyePACS dataset imposed to the U^2-Net-S, as well as the disjoint training setting.

Again, the SVDKL-CBM and -sCBM are contrasted by the use of a vanilla CBM and sCBM that utilize a simple linear fully connected classifiers to verify the benefit of the Bayesian modeling through the GP layer. Aligning the observations of Chapter 5, the results showed the SVDKL-based models to provide a significant benefit for both the diagnostic performance and the alignment of the predictive uncertainty with erroneous predictions compared to the vanilla classifier.

Along with the comparably low computational complexity of the models, this preliminary analysis showed particularly the SVDKL-sCBM's potential to be used as a transparent, explainable, and lightweight DR screening system by integrating both the lightweight U^2-Net architecture and SVDKL framework. As discussed in more detail in Chapter 6, completing the set of concepts, integrating superior U^2-Net backbones, and conducting full end-to-end training are expected to further improve the diagnostic performance of both the CBM and sCBM designs as well as the quality and benefit of the provided explanations, which is left to analyze in future work.

7.2 Outlook

practical considerations and implications

Disregarding which of the CBM or sCBM approaches will be adopted, particularly transferring the ensemble-based U^2-Net and SVDKL models, which were found to provide both the best predictive performance and most reliable uncertainty estimates, to these holistic systems while retaining viable computational cost requires additional engineering. Two suitable methods that could help to address these challenges are batch [203] or group ensembles [204] and methods such as uncertainty-aware distillation [205]. Particularly, the latter could be utilized to transfer the good uncertainty calibration obtained from the DEs to single model instances, which promises to ease integration to the CBM design and to minimize computational costs.

[203] Wen *et al.*, "Batchensemble: an alternative approach to efficient ensemble and lifelong learning" (2020)

[204] Chen and Shrivastava, "Group ensemble: learning an ensemble of convnets in a single convnet" (2020)

[205] Shen *et al.*, "Real-time uncertainty estimation in computer vision via uncertainty-aware distribution distillation" (2021)

Moreover, as was found in Chapter 5, to retain the uncertainty calibration of the additive GP layer and prevent the occurrence of the feature collapse in case of conducting full end-to-end training of either the SVDKL-based CBM or sCBM, the U^2-Net would have to be modeled in a fully Bayesian manner. This, in turn, could also improve the quality of the concepts, i.e., the precision of the segmentation masks, and allow the detection of unknown lesions and biomarkers additionally enhancing the quality of the provided explanations. To this end, deterministic uncertainty quantification methods such as regularizing the CNN backbone using SN [173] on first sight promise a more efficient implementation compared to Monte Carlo-based sampling approaches. However, this work found the former to be difficult to optimize and result in inconsistent uncertainty calibration. Consequently, using MCD-based sampling or DEs might be preferable in case the additional computational cost and inference time are acceptable for the application at hand.

human interventions

Besides its benefit for the model transparency and explainability, the CBM design additionally allows for human-model interactions, which could enable medical experts to intervene in the prediction [36]. This would provide both deployers and developers the opportunity to analyze the dependency of the predicted DR severity grades on the concepts, could prevent model failures through correcting erroneous concept predictions, and would benefit the required transparent model design and validation process that is part of the quality management process required for approval by means of the EU AI Act [15] and the MDR [16].

To intervene in a prediction of the CBM and probe the final estimated DR severity grade, false positive or negative predicted lesions could be removed or added within the segmentation masks manually or semi-automatically by, e.g., exploiting predictive uncertainty obtained through full Bayesian modeling of the SVDKL-CBM as discussed above. Subsequently, the prediction of the severity grade can directly be updated based on the modified segmentation maps. In case of using the sCBM architecture design, the changes introduced to the segmentation masks by the intervention could be projected back to the shared soft concepts, based on which the severity classification could be updated accordingly. This could, for instance, be accomplished by optimizing the soft concepts using gradient descent on the error between the original and the updated concept segmentation maps.

Aligning the intrinsic motivation of this work on fostering transparent decision-making by incorporating and quantifying model uncertainty, additionally taking care of human uncertainty on these interventions as proposed by [206] would be beneficial and interesting to study in future work.

[206] Collins *et al.*, "Human uncertainty in concept-based ai systems" (2023)

active learning and data denoising

Moreover, full Bayesian modeling of both the segmentation and classification task as well as the possibility to intervene in predictions could provide a means to apply active learning strategies and conduct data denoising. This would allow experts to add annotations for previously unlabeled data or to update erroneous pixel- and image-level annotations for which either the severity grade or the predicted lesions are very uncertain or do not match each other. The latter could, e.g., be detected automatically by a set of hardcoded rules asking for manual intervention whenever, e.g., the presence of a NV is detected but the classifier predicts a DR severity level less than PDR. In addition, this would allow to warn users to carefully inspect and validate the diagnosis and, if necessary, to intervene in the system's prediction.

information leakage through concept representations

However, despite promising superiority to, particularly, post-hoc explanations that rely, e.g., on saliency methods by means of the well-thought theoretical motivation of CBMs, i.e., explicitly enforcing the alignment of the latent representation used for classification to align with human-interpretable concepts via the bottleneck design, these were observed to unintendedly leak information encoded in the statics of the concept representations that the classifier might exploit [207]. This could weaken the class-concept relation of the provided explanations, which to analyze and prevent has to be subject to future research.

[207] Mahinpei *et al.*, "Promises and pitfalls of black-box concept learning models" (2021)

general improvements

Moreover, both the above-discussed SVDKL-CBM and -sCBM focus primarily on providing explanations regarding the classified severity levels, but not for the segmentation of the biomarkers themselves. That is, the model lacks explanations as to why a segmented area is considered, e.g., a MA rather than a HE. Taking care of these issues by, e.g., additionally exploiting the idea of prototype models such as *This looks like that* [84] to explain the presence of the lesions themselves could further improve the overall model's transparency, explainability, and trustworthiness. Furthermore, pairing the algorithm with a prediction for the disease progression risk to provide a recommendation for the next examination and extending the model to predict the presence and severity of other eye diseases such as DME would significantly increase the clinical utility of the system.

[84] Chen *et al.*, "This looks like that: deep learning for interpretable image recognition" (2019)

final conclusion

The proposed transparent model design of this work based on the CBM design, the lightweight U^2-Net variants, and the SVDKL framework showed to improve the transparency, explainability, and uncertainty calibration of the model predictions while promising straightforward mobile application. This is expected to help narrow the gap for the translation of DL-based diagnostic systems into clinical practice by taking regulatory and societary demands for deploying responsible AI into account and lowering deployment cost. Evaluating the effect of the discussed end-to-end training of the CBM and the proposed sCBM with additional active learning and weak supervision on the DR grading performance, quality of the explanations given by the lesion segmentation masks, and uncertainty calibration would be of interest for future work. Moreover, analyzing whether these theoretical findings of the proposed, promising screening system translate to real-world use remains open for future work, which ultimately should entail validating the explainability and predictive transparency through the uncertainty calibration in a study involving end-users, i.e., general practitioners and medical assistants, as well as patients.

Appendix

TABLE 1: Out-of-distribution AUROC for the RFMID and SIIM-ISIC data, for the task of distinguishing between samples from these and the EyePACS dataset based on each of the model's predictive uncertainty estimate. The highest mean values per metric are displayed in bold. Statistical significance according to section 5.2.10 is indicated with an asterisk.

					DR severity		referable DR	
		SN	MCD	DE	total	epistemic	total	epistemic
IDRiD	CNN	-	-	-	42.9 (2.8)	-	33.9 (2.3)	-
	BCNN	-	√	-	41.9 (5.4)	71.5 (3.7)	32.6 (1.3)	41.6 (1.9)
		-	-	√	**50.3 (1.3)**	70.6 (0.8)	**34.8 (0.8)**	43.0 (2.1)
		-	√	√	45.8 (2.1)	**76.5 (1.3)**	33.3 (0.5)	42.4 (0.9)
	SVDKL	-	√	√	46.2 (2.1)	74.4 (1.6)	34.0 (0.9)	**43.9 (1.2)**
		√	√	√	45.8 (1.1)	73.9 (0.9)	33.0 (0.8)	43.6 (2.1)
RFMID-DR	CNN	-	-	-	54.6 (4.1)	-	50.8 (2.5)	-
	BCNN	-	√	-	55.4 (7.0)	69.1 (3.5)	50.8 (2.5)	54.6 (3.0)
		-	-	√	**64.9 (1.5)**	74.7 (1.7)	**54.4 (0.8)**	57.7 (0.9)
		-	√	√	60.1 (1.9)	72.7 (1.1)	52.7 (0.6)	56.4 (1.2)
	SVDKL	-	√	√	60.9 (2.3)	75.2 (1.3)	53.9 (1.4)	**60.3 (1.8)***
		√	√	√	63.6 (1.8)	**77.5 (1.5)***	54.2 (1.0)	60.0 (1.8)*
RFMID	CNN	-	-	-	59.0 (4.0)	-	56.8 (2.6)	-
	BCNN	-	√	-	59.3 (5.1)	74.2 (2.7)	56.9 (2.6)	64.7 (2.1)
		-	-	√	**70.1 (1.5)**	79.0 (1.3)	**63.0 (0.9)**	67.6 (1.0)
		-	√	√	65.5 (1.3)	78.4 (0.7)	60.3 (0.6)	68.1 (0.8)
	SVDKL	-	√	√	67.2 (1.6)	81.0 (1.0)*	61.7 (1.5)	**71.2 (1.7)***
		√	√	√	69.3 (1.3)	**82.0 (1.2)***	62.2 (1.0)	69.9 (1.3)*
RFMID-O	CNN	-	-	-	63.8 (4.4)	-	63.7 (3.2)	-
	BCNN	-	√	-	63.8 (4.8)	76.0 (2.8)	64.1 (3.2)	72.5 (2.3)
		-	-	√	**75.1 (1.7)**	80.2 (1.1)	**71.6 (0.9)**	75.4 (1.0)
		-	√	√	70.5 (1.2)	79.6 (0.7)	68.5 (0.7)	76.3 (0.9)
	SVDKL	-	√	√	72.9 (1.3)	**82.8 (1.1)***	70.2 (1.3)	**79.4 (1.3)***
		√	√	√	74.5 (1.1)	**82.8 (1.3)***	70.9 (1.1)	77.6 (1.0)*
SIIM-ISIC	CNN	-	-	-	56.8 (11.9)	-	56.3 (7.2)	-
	BCNN	-	√	-	68.8 (11.5)	94.5 (4.1)	67.6 (6.5)	90.6 (3.3)
		-	-	√	85.6 (1.8)	95.6 (0.6)	76.1 (1.9)	87.4 (1.7)
		-	√	√	**87.9 (2.4)**	**99.1 (0.4)**	**80.5 (2.2)**	**96.7 (0.8)**
	SVDKL	-	√	√	84.4 (2.6)	97.9 (0.6)	76.2 (2.0)	94.2 (0.6)
		√	√	√	83.1 (2.1)	97.5 (0.4)	71.9 (3.8)	89.1 (1.6)

Acronyms

ACC	accuracy
AI	artificial intelligence
ANOVA	analysis of variance
AUARC	area under the accuracy-rejection curve
AUARCa	AUARC-aleatoric
AUARCt	AUARC-total
AUPRC	area under the precision-recall curve
AUROC	area under the receiver-operating-characteristic
AUROCe	AUROC-epistemic
AUROCt	AUROC-total
BCE	binary cross-entropy
BMA	Bayesian model average
BN	batch normalization
BNN	Bayesian neural network
CBM	concept bottleneck model
CE	cross-entropy
CNN	convolutional neural network
CWS	cotton wool spot
DC	depthwise separable convolution
DE	deep ensemble
DKL	deep kernel learning
DL	deep learning
DM	diabetes mellitus
DME	diabetic macular edema
DNN	deep neural network
DR	diabetic retinopathy

DSC	Dice similiarty coefficient
ECE	expected calibration error
ELBO	evidence lower bound
ETDRS	Early Treatment Diabetic Retinopathy Study
EU	European Union
fBCE	focal binary cross-entropy
FDA	Food and Drug Administration
GDPR	General Data Protection Regulation
GP	Gaussian process
HD	Hausdorff-distance
HE	haemorrhage
HMC	Hamilton Monte Carlo
HX	hard exudate
ICDR	international clinical diabetic retinopathy
ID	in-distribution
IQR	interquartile range
IRMA	intraretinal microvascular abnormality
KDE	kernel density estimation
KL	Kullback-Leibler
LLM	large language model
LMM	large multimodal model
MA	microaneurysm
MAC	multiply-accumulate
MAP	maximum a posteriori
MBC	mobile inverted bottleneck convolution
MCD	Monte Carlo dropout
MCMC	Markov chain Monte Carlo
MDR	medical device regulation

MFVI	mean field variational inference
ML	machine learning
MLE	maximum likelihood estimation
MSE	mean squared error
mtmDR	more-than-mild DR
NLL	negative log-likelihood
NN	neural network
NPDR	non-proliferative DR
NV	neovascularization
OCT	optical coherence tomography
OD	optic disc
OOD	out-of-distribution
PDR	proliferative DR
QWK	quadratically weighted Cohen's kappa
rDR	referable DR
ReLU	rectified linear unit
RSU	residual U-block
sCBM	soft-concept bottleneck model
sDR	DR severity
SE	squeeze-and-excitation
SGD	stochastic gradient descent
SN	spectral norm
SVDKL	stochastic variational deep kernel learning
SVGP	sparse variational GP
SVI	stochastic variational inference
T1DM	type 1 DM
T2DM	type 2 DM
U.S.	United States

UK	United Kingdom
VB	venous beading
VEGF	vascular endothelial growth factor
VI	variational inference
vtDR	vision-threatening DR
XAI	explainable artificial intelligence

List of Figures

List of Tables

Bibliography

[1] S. Hawking. "Creating AI Could Be the Biggest & Last Event in Human History, Google Zeitgeist". (2015), [Online]. Available: https://youtu.be/a1X5x3OGduc?si=A253dd6hzIjeD4Ap (visited on 06/14/2024).

[2] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks", *Science*, vol. 313, no. 5786, pp. 504–507, 2006. DOI: 10.1126/science.1127647.

[3] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning", *Nature*, vol. 521, no. 7553, pp. 436–444, 2015. DOI: 10.1038/nature14539.

[4] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets", *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006. DOI: 10.1162/neco.2006.18.7.1527.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks", in *Advances in Neural Information Processing Systems (NeurIPS)*, F. Pereira, C. J. Burges, L. Bottou, and K. Q. Weinberger, Eds., vol. 25, Curran Associates, Inc., 2012.

[6] Y.-C. Peng, W.-J. Lee, Y.-C. Chang, W. P. Chan, and S.-J. Chen, "Radiologist burnout: Trends in medical imaging utilization under the national health insurance system with the universal code bundling strategy in an academic tertiary medical centre", *European Journal of Radiology*, vol. 157, p. 110 596, 2022. DOI: 10.1016/j.ejrad.2022.110596.

[7] U.S. Food & Drug Administration. "Artificial Intelligence and Machine Learning (AI/ML)-Enabled Medical Devices". (2024), [Online]. Available: https://www.fda.gov/medical-devices/software-medical-device-samd/artificial-intelligence-and-machine-learning-aiml-enabled-medical-devices (visited on 06/14/2024).

[8] S. Zhu, M. Gilbert, I. Chetty, and F. Siddiqui, "The 2021 landscape of FDA-approved artificial intelligence/machine learning-enabled medical devices: An analysis of the characteristics and intended use", *International Journal of Medical Informatics*, vol. 165, p. 104 828, 2022. DOI: 10.1016/j.ijmedinf.2022.104828.

[9] D. Lyell, E. Coiera, J. Chen, P. Shah, and F. Magrabi, "How machine learning is embedded to support clinician decision making: An analysis of fda-approved medical devices", *BMJ Health & Care Informatics*, vol. 28, no. 1, 2021. DOI: 10.1136/bmjhci-2020-100301.

[10] M. D. Abràmoff, Y. Lou, A. Erginay, W. Clarida, R. Amelon, J. C. Folk, and M. Niemeijer, "Improved automated detection of diabetic retinopathy on a publicly available dataset through integration of deep learning", *Investigative Opthalmology & Visual Science*, vol. 57, no. 13, p. 5200, 2016. DOI: 10.1167/iovs.16-19964.

[11] E. Petersen, Y. Potdevin, E. Mohammadi, S. Zidowitz, S. Breyer, D. Nowotka, S. Henn, L. Pechmann, M. Leucker, P. Rostalski, and C. Herzog, "Responsible and regulatory conform machine learning for medicine: A survey of challenges and solutions", *IEEE Access*, vol. 10, pp. 58 375–58 418, 2022. DOI: 10.1109/access.2022.3178382.

[12] T. Y. Wong and N. M. Bressler, "Artificial intelligence with deep learning technology looks into diabetic retinopathy screening", *JAMA*, vol. 316, no. 22, p. 2366, 2016. DOI: 10.1001/jama.2016.17563.

[13] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI", *Information Fusion*, vol. 58, pp. 82–115, 2020. DOI: 10.1016/j.inffus.2019.12.012.

[14] P. Rajpurkar, E. Chen, O. Banerjee, and E. J. Topol, "AI in health and medicine", *Nature Medicine*, vol. 28, no. 1, pp. 31–38, 2022. DOI: 10.1038/s41591-021-01614-0.

[15] European Commission, Directorate-General for Communications Networks, Content and Technology. "Artificial Intelligence Act: Proposal for a Regulation of the European Parliament and of the council laying down harmonised rules on artificial intelligence (Artificial Intelligence Act) and amending certrain union legislative acts". (2021), [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex:52021PC0206 (visited on 06/14/2024).

[16] European Parliament, Council of the European Union. "Medical Device Regulation: Regulation (EU) 2017/745 of the European Parliament and of the Council of 5 April 2017 on medical devices, amending Directive 2001/83/EC, Regulation (EC) No 178/2002 and Regulation (EC) No 1223/2009 and repealing Council Directives 90/385/EEC and 93/42/EEC". (2023), [Online]. Available: http://data.europa.eu/eli/reg/2017/745/oj (visited on 06/14/2024).

[17] European Parliament, Council of the European Union. "General Data Protection Regulation: Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)". (2016), [Online]. Available: http://data.europa.eu/eli/reg/2016/679/oj (visited on 06/14/2024).

[18] C. Ryngaert and M. Taylor, "The GDPR as global data protection regulation?", *AJIL Unbound*, vol. 114, pp. 5–9, 2020. DOI: 10.1017/aju.2019.80.

[19] B. Goodman and S. Flaxman, "European Union regulations on algorithmic decision making and a "right to explanation"", *AI Magazine*, vol. 38, no. 3, pp. 50–57, 2017. DOI: 10.1609/aimag.v38i3.2741.

[20] American Diabetes Association Professional Practice Committee, "4. Comprehensive medical evaluation and assessment of comorbidities: Standards of care in diabetes—2024", *Diabetes Care*, vol. 47, no. Supplement_1, pp. 52–76, 2023. DOI: 10.2337/dc24-s004.

[21] Deutsche Diabetes Gesellschaft (DDG). "S3-Leitlinie Therapie des Typ-1-Diabetes, Version 5". (2023), [Online]. Available: https://www.ddg.info/fileadmin/user_upload/05_Behandlung/01_Leitlinien/Evidenzbasierte_Leitlinien/2023/S3-LL-Therapie-Typ-1-Diabetes-Version-5-20230922.pdf (visited on 06/14/2024).

[22] Bundesärztekammer, Kassenärztliche Bundesvereinigung, and Arbeitsgemeinschaft der Wissenschaftlichen Medizinischen Fachgesellschaften. "Nationale Versorgungs-Leitlinie Typ-2-Diabetes". (2023), [Online]. Available: https://www.leitlinien.de/themen/diabetes/version-3 (visited on 06/14/2024).

[23] T. Y. Wong, J. Sun, R. Kawasaki, P. Ruamviboonsuk, N. Gupta, V. C. Lansingh, M. Maia, W. Mathenge, S. Moreker, M. M. K. Muqit, S. Resnikoff, J. Verdaguer, P. Zhao, F. Ferris, L. P. Aiello, and H. R. Taylor, "Guidelines on diabetic eye care: The international council of ophthalmology recommendations for screening, follow-

up, referral, and treatment based on resource settings", *Ophthalmology*, vol. 125, pp. 1608–1622, 10 2018. DOI: 10.1016/J.OPHTHA.2018.04.007.

[24] S. R. Benoit, B. Swenor, L. S. Geiss, E. W. Gregg, and J. B. Saaddine, "Eye care utilization among insured people with diabetes in the U.S., 2010–2014", *Diabetes Care*, vol. 42, no. 3, pp. 427–433, 2019. DOI: 10.2337/dc18-0828.

[25] J. C. Javitt, L. P. Aiello, Y. Chiang, F. L. Ferris, J. K. Canner, and S. Greenfield, "Preventive eye care in people with diabetes is cost-saving to the federal government: Implications for health-care reform", *Diabetes Care*, vol. 17, no. 8, pp. 909–917, 1994. DOI: 10.2337/diacare.17.8.909.

[26] B. Foot and C. MacEwen, "Surveillance of sight loss due to delay in ophthalmic treatment or review: Frequency, cause and outcome", *Eye*, vol. 31, no. 5, pp. 771–775, 2017. DOI: 10.1038/eye.2017.1.

[27] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim, "Sanity checks for saliency maps", in *Advances in Neural Information Processing Systems (NeurIPS)*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31, Curran Associates, Inc., 2018. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2018/file/294a8ed24b1ad22ec2e7efea049b8737-Paper.pdf.

[28] N. Arun, N. Gaw, P. Singh, K. Chang, M. Aggarwal, B. Chen, K. Hoebel, S. Gupta, J. Patel, M. Gidwani, J. Adebayo, M. D. Li, and J. Kalpathy-Cramer, "Assessing the trustworthiness of saliency maps for localizing abnormalities in medical imaging", *Radiology: Artificial Intelligence*, vol. 3, no. 6, 2021. DOI: 10.1148/ryai.2021200267.

[29] A. Saporta, X. Gui, A. Agrawal, A. Pareek, S. Q. H. Truong, C. D. T. Nguyen, V.-D. Ngo, J. Seekins, F. G. Blankenberg, A. Y. Ng, M. P. Lungren, and P. Rajpurkar, "Benchmarking saliency methods for chest x-ray interpretation", *Nature Machine Intelligence*, vol. 4, no. 10, pp. 867–878, 2022. DOI: 10.1038/s42256-022-00536-x.

[30] J. De Fauw, J. R. Ledsam, B. Romera-Paredes, S. Nikolov, N. Tomasev, S. Blackwell, H. Askham, X. Glorot, B. O'Donoghue, D. Visentin, G. van den Driessche, B. Lakshminarayanan, C. Meyer, F. Mackinder, S. Bouton, K. Ayoub, R. Chopra, D. King, A. Karthikesalingam, C. O. Hughes, R. Raine, J. Hughes, D. A. Sim, C. Egan, A. Tufail, H. Montgomery, D. Hassabis, G. Rees, T. Back, P. T. Khaw, M. Suleyman, J. Cornebise, P. A. Keane, and O. Ronneberger, "Clinically applicable deep learning for diagnosis and referral in retinal disease", *Nature Medicine*, vol. 24, no. 9, pp. 1342–1350, 2018. DOI: 10.1038/s41591-018-0107-6.

[31] L. Hansen, M. Sieren, M. Hobe, A. Saalbach, H. Schulz, J. Barkhausen, and M. P. Heinrich, "Radiographic assessment of CVC malpositioning: How can AI best support clinicians?", in *Proceedings of the International Conference on Medical Imaging with Deep Learning (MIDL)*, 2021. [Online]. Available: https://openreview.net/forum?id=ImcP8kkqtfZ.

[32] C. M. Bishop and H. Bishop, *Deep Learning: Foundations and Concepts*. Springer International Publishing, 2024. DOI: 10.1007/978-3-031-45468-4.

[33] L. V. Jospin, H. Laga, F. Boussaid, W. Buntine, and M. Bennamoun, "Hands-on bayesian neural networks—a tutorial for deep learning users", *IEEE Computational Intelligence Magazine*, vol. 17, no. 2, pp. 29–48, 2022. DOI: 10.1109/mci.2022.3155327.

[34] P. Porwal, S. Pachade, R. Kamble, M. Kokare, G. Deshmukh, V. Sahasrabuddhe, and F. Meriaudeau, "Indian diabetic retinopathy image dataset (IDRiD): A database for diabetic retinopathy screening research", *Data*, vol. 3, no. 3, p. 25, 2018. DOI: 10.3390/data3030025.

[35] G. Quellec, H. Al Hajj, M. Lamard, P.-H. Conze, P. Massin, and B. Cochener, "Explain: Explanatory artificial intelligence for diabetic retinopathy diagnosis", *Medical Image Analysis*, vol. 72, p. 102 118, 2021. DOI: 10.1016/j.media.2021.102118.

[36] P. W. Koh, T. Nguyen, Y. S. Tang, S. Mussmann, E. Pierson, B. Kim, and P. Liang, "Concept bottleneck models", in *Proceedings of the 37th International Conference on Machine Learning (ICML)*, H. D. III and A. Singh, Eds., ser. Proceedings of Machine Learning Research, vol. 119, PMLR, 2020, pp. 5338–5348. [Online]. Available: https://proceedings.mlr.press/v119/koh20a.html.

[37] A. Sarkar, D. Vijaykeerthy, A. Sarkar, and V. N. Balasubramanian, "A framework for learning ante-hoc explainable models via concepts", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2022. DOI: 10.1109/cvpr52688.2022.01004.

[38] C. P. Wilkinson, F. L. Ferris, R. E. Klein, P. P. Lee, C. D. Agardh, M. Davis, D. Dills, A. Kampik, R. Pararajasegaram, and J. T. Verdaguer, "Proposed international clinical diabetic retinopathy and diabetic macular edema disease severity scales", *Ophthalmology*, vol. 110, no. 9, pp. 1677–1682, 2003. DOI: 10.1016/s0161-6420(03)00475-5.

[39] X. Qin, Z. Zhang, C. Huang, M. Dehghan, O. R. Zaiane, and M. Jagersand, "U^2-Net: Going deeper with nested U-structure for salient object detection", *Pattern Recognition*, vol. 106, p. 107 404, 2020. DOI: 10.1016/j.patcog.2020.107404.

[40] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation", in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, Springer International Publishing, 2015, pp. 234–241. DOI: 10.1007/978-3-319-24574-4_28.

[41] M. Siebert and P. Rostalski, "Performance evaluation of lightweight convolutional neural networks on retinal lesion segmentation", in *Medical Imaging 2022: Computer-Aided Diagnosis*, K. Drukker and K. M. Iftekharuddin, Eds., International Society for Optics and Photonics, SPIE, 2022, p. 1 203 334. DOI: 10.1117/12.2611796.

[42] A. G. Wilson, Z. Hu, R. Salakhutdinov, and E. P. Xing, "Deep kernel learning", in *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics ((AISTATS)*, A. Gretton and C. C. Robert, Eds., ser. Proceedings of Machine Learning Research, vol. 51, Cadiz, Spain: PMLR, 2016, pp. 370–378. [Online]. Available: https://proceedings.mlr.press/v51/wilson16.html.

[43] J. Liu, Z. Lin, S. Padhy, D. Tran, T. Bedrax-Weiss, and B. Lakshminarayanan, "Simple and principled uncertainty estimation with deterministic deep learning via distance awareness", in *Advances in Neural Information Processing Systems (NeurIPS)*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33, Curran Associates, Inc., 2020, pp. 7498–7512. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/543e83748234f7cbab21aa0ade66565f-Paper.pdf.

[44] G.-L. Tran, E. V. Bonilla, J. Cunningham, P. Michiardi, and M. Filippone, "Calibrating deep convolutional gaussian processes", in *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS)*, K. Chaudhuri and M. Sugiyama, Eds., ser. Proceedings of Machine Learning Research, vol. 89, PMLR, 2019, pp. 1554–1563. [Online]. Available: https://proceedings.mlr.press/v89/tran19a.html.

[45] M. Siebert, J. Graßhoff, and P. Rostalski, "Uncertainty analysis of deep kernel learning methods on diabetic retinopathy grading", *IEEE Access*, vol. 11, pp. 146 173–146 184, 2023. DOI: 10.1109/access.2023.3343642.

[46] K. P. Murphy, *Probabilistic Machine Learning: An introduction*. MIT press, 2022. [Online]. Available: https://probml.github.io/book1.

[47] I. Neutelings. "Tikz.net - graphics with tikz in latex - neural networks". Licensed under CC BY-SA 4.0., [Online]. Available: https://tikz.net/neural_networks/ (visited on 06/14/2024).

[48] Q. Jarosz. "Neuron hand-tuned.svg". Licensed under CC BY-SA 4.0. (2019), [Online]. Available: https://commons.wikimedia.org/wiki/File:Neuron_Hand-tuned.svg (visited on 06/14/2024).

[49] F. Rosenblatt, *Principles of Neurodynamics: Perceptrons and the Theory of Brain-Mechanisms.* Spartan Books, 1962.

[50] C. Molnar, *Interpretable Machine Learning: A Guide for Making Black Box Models Explainable*, 2nd ed. 2022. [Online]. Available: https://christophm.github.io/interpretable-ml-book.

[51] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning", in *Proceedings of the 30th International Conference on Machine Learning (ICML)*, S. Dasgupta and D. McAllester, Eds., ser. Proceedings of Machine Learning Research, vol. 28, Atlanta, Georgia, USA: PMLR, 2013, pp. 1139–1147. [Online]. Available: https://proceedings.mlr.press/v28/sutskever13.html.

[52] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization", in *International Conference on Learning Representations (ICLR)*, 2015. DOI: 10.48550/ARXIV.1412.6980.

[53] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks", in *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS)*, G. Gordon, D. Dunson, and M. Dudík, Eds., ser. Proceedings of Machine Learning Research, vol. 15, Fort Lauderdale, FL, USA: PMLR, 2011, pp. 315–323. [Online]. Available: https://proceedings.mlr.press/v15/glorot11a.html.

[54] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions", 2018. [Online]. Available: https://openreview.net/forum?id=SkBYYyZRZ.

[55] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift", in *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, F. Bach and D. Blei, Eds., ser. Proceedings of Machine Learning Research, vol. 37, Lille, France: PMLR, 2015, pp. 448–456. [Online]. Available: https://proceedings.mlr.press/v37/ioffe15.html.

[56] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry, "How does batch normalization help optimization?", in *Advances in Neural Information Processing Systems (NeurIPS)*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31, Curran Associates, Inc., 2018. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2018/file/905056c1ac1dad141560467e0a99e1cf-Paper.pdf.

[57] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition", IEEE, 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[58] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, "Visualizing the loss landscape of neural nets", in *Advances in Neural Information Processing Systems (NeurIPS)*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31, Curran Associates, Inc., 2018. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2018/file/a41b3bb3e6b050b6c9067c67f663b915-Paper.pdf.

[59] V. Dumoulin and F. Visin, "A guide to convolution arithmetic for deep learning", *arXiv*, 2016. DOI: 10.48550/ARXIV.1603.07285. [Online]. Available: https://github.com/vdumoulin/conv_arithmetic.

[60] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions", in *International Conference on Learning Representations (ICLR)*, 2016. DOI: 10.48550/ARXIV.1511.07122.

[61] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", in *International Conference on Learning Representations (ICLR)*, 2015. DOI: 10.48550/arXiv.1409.1556.

[62] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[63] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting", *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014. [Online]. Available: http://jmlr.org/papers/v15/srivastava14a.html.

[64] F. Wenzel, J. Snoek, D. Tran, and R. Jenatton, "Hyperparameter ensembles for robustness and uncertainty quantification", in *Advances in Neural Information Processing Systems (NeurIPS)*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33, Curran Associates, Inc., 2020, pp. 6514–6527. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/481fbfa59da2581098e841b7afc122f1-Paper.pdf.

[65] R. Geirhos, J.-H. Jacobsen, C. Michaelis, R. Zemel, W. Brendel, M. Bethge, and F. A. Wichmann, "Shortcut learning in deep neural networks", *Nature Machine Intelligence*, vol. 2, no. 11, pp. 665–673, 2020. DOI: 10.1038/s42256-020-00257-z.

[66] J. R. Zech, M. A. Badgeley, M. Liu, A. B. Costa, J. J. Titano, and E. K. Oermann, "Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study", *PLOS Medicine*, vol. 15, no. 11, A. Sheikh, Ed., pp. 1–17, 2018. DOI: 10.1371/journal.pmed.1002683.

[67] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks", in *International Conference on Learning Representations (ICLR)*, 2014. DOI: 10.48550/ARXIV.1312.6199.

[68] Y. Zhang, M. Gong, T. Liu, G. Niu, X. Tian, B. Han, B. Schölkopf, and K. Zhang, "Adversarial robustness through the lens of causality", in *International Conference on Learning Representations (ICLR)*, 2022. [Online]. Available: https://openreview.net/forum?id=cZAi1yWpiXQ.

[69] J. Su, D. V. Vargas, and K. Sakurai, "One pixel attack for fooling deep neural networks", *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 5, pp. 828–841, 2019. DOI: 10.1109/tevc.2019.2890858.

[70] S. G. Finlayson, J. D. Bowers, J. Ito, J. L. Zittrain, A. L. Beam, and I. S. Kohane, "Adversarial attacks on medical machine learning", *Science*, vol. 363, no. 6433, pp. 1287–1289, 2019. DOI: 10.1126/science.aaw4399.

[71] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples", 2015. DOI: 10.48550/ARXIV.1412.6572.

[72] V. Dignum, *Responsible Artificial Intelligence: How to Develop and Use AI in a Responsible Way* (Artificial Intelligence: Foundations, Theory, and Algorithms). Springer International Publishing, 2019. DOI: 10.1007/978-3-030-30371-6.

[73] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks", in *Proceedings of the 34th International Conference on Machine Learning (ICML)*, D. Precup and Y. W. Teh, Eds., ser. Proceedings of Machine Learning Research, vol. 70, PMLR, 2017, pp. 1321–1330. [Online]. Available: https://proceedings.mlr.press/v70/guo17a.html.

[74] J. H. Friedman, "Greedy function approximation: A gradient boosting machine.", *The Annals of Statistics*, vol. 29, no. 5, 2001. DOI: 10.1214/aos/1013203451.

[75] A. Goldstein, A. Kapelner, J. Bleich, and E. Pitkin, "Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation", *Journal of Computational and Graphical Statistics*, vol. 24, no. 1, pp. 44–65, 2015. DOI: 10.1080/10618600.2014.907095.

[76] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization", *International Journal of Computer Vision*, vol. 128, no. 2, pp. 336–359, 2019. DOI: 10.1007/s11263-019-01228-7.

[77] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net", in *International Conference on Learning Representations (ICLR) Workshop*, 2015. DOI: 10.48550/ARXIV.1412.6806.

[78] J. K. Winkler, C. Fink, F. Toberer, A. Enk, T. Deinlein, R. Hofmann-Wellenhof, L. Thomas, A. Lallas, A. Blum, W. Stolz, and H. A. Haenssle, "Association between surgical skin markings in dermoscopic images and diagnostic performance of a deep learning convolutional neural network for melanoma recognition", *JAMA Dermatology*, vol. 155, no. 10, p. 1135, 2019. DOI: 10.1001/jamadermatol.2019.1735.

[79] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead", *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019. DOI: 10.1038/s42256-019-0048-x.

[80] M. Ghassemi, L. Oakden-Rayner, and A. L. Beam, "The false hope of current approaches to explainable artificial intelligence in health care", *The Lancet Digital Health*, vol. 3, no. 11, e745–e750, 2021. DOI: 10.1016/s2589-7500(21)00208-9.

[81] M. Ribeiro, S. Singh, and C. Guestrin, "'Why should I trust you?': Explaining the predictions of any classifier", in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, Association for Computational Linguistics, 2016. DOI: 10.18653/v1/n16-3020.

[82] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions", in *Advances in Neural Information Processing Systems (NeurIPS)*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30, Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf.

[83] O. Li, H. Liu, C. Chen, and C. Rudin, "Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions", in *Proceedings of the AAAI Conference on Artificial Intelligence*, ser. AAAI'18/IAAI'18/EAAI'18, New Orleans, Louisiana, USA: AAAI Press, 2018. DOI: 10.1609/aaai.v32i1.11771.

[84] C. Chen, O. Li, D. Tao, A. Barnett, C. Rudin, and J. K. Su, "This looks like that: Deep learning for interpretable image recognition", in *Advances in Neural Information Processing Systems (NeurIPS)*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32, Curran Associates, Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2019/file/adf7ee2dcf142b0e11888e72b43fcb75-Paper.pdf.

[85] Z. Carmichael, S. Lohit, A. Cherian, M. J. Jones, and W. J. Scheirer, "Pixel-grounded prototypical part networks", in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2024, pp. 4768–4779. [Online]. Available: https://openaccess.thecvf.com/content/WACV2024/papers/Carmichael_Pixel-Grounded_Prototypical_Part_Networks_WACV_2024_paper.pdf.

[86] Z. Chen, Y. Bei, and C. Rudin, "Concept whitening for interpretable image recognition", *Nature Machine Intelligence*, vol. 2, no. 12, pp. 772–782, 2020. DOI: 10.1038/s42256-020-00265-z.

[87] S. Gautam, M. M.-C. Höhne, S. Hansen, R. Jenssen, and M. Kampffmeyer, "This looks more like that: Enhancing self-explaining models by prototypical relevance propagation", *Pattern Recognition*, vol. 136, p. 109 172, 2023. DOI: 10.1016/j.patcog.2022.109172.

[88] O. Davoodi, S. Mohammadizadehsamakosh, and M. Komeili, "On the interpretability of part-prototype based classifiers: A human centric analysis", *Scientific Reports*, vol. 13, no. 1, 2023. DOI: 10.1038/s41598-023-49854-z.

[89] A. Hoffmann, C. Fanconi, R. Rade, and J. Kohler, "This looks like that... Does it? Shortcomings of latent space prototype interpretability in deep networks", in *Workshop on Theoretic Foundation, Criticism, and Application Trend of Explainable AI at the International Conference on Machine Learning (ICML)*, 2021. DOI: 10.48550/arxiv.2105.02968.

[90] G. Ciravegna, P. Barbiero, F. Giannini, M. Gori, P. Liò, M. Maggini, and S. Melacci, "Logic explained networks", *Artificial Intelligence*, vol. 314, p. 103 822, 2023. DOI: 10.1016/j.artint.2022.103822.

[91] L. Smith and Y. Gal, "Understanding measures of uncertainty for adversarial example detection", in *Proceedings of the 34th Conference on Uncertainty in Artificial Intelligence (UAI)*, AUAI Press, 2018. [Online]. Available: http://auai.org/uai2018/proceedings/papers/207.pdf.

[92] E. Hüllermeier and W. Waegeman, "Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods", *Machine Learning*, vol. 110, no. 3, pp. 457–506, 2021. DOI: 10.1007/s10994-021-05946-3.

[93] U. Bhatt, J. Antorán, Y. Zhang, Q. V. Liao, P. Sattigeri, R. Fogliato, G. Melançon, R. Krishnan, J. Stanley, O. Tickoo, L. Nachman, R. Chunara, M. Srikumar, A. Weller, and A. Xiang, "Uncertainty as a form of transparency: Measuring, communicating, and using uncertainty", in *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society (AIES)*, Association for Computing Machinery, 2021, pp. 401–413. DOI: 10.1145/3461702.3462571. [Online]. Available: https://doi.org/10.1145/3461702.3462571.

[94] N. Houlsby, F. Huszár, Z. Ghahramani, and M. Lengyel, "Bayesian active learning for classification and preference learning", *arXiv*, 2011. DOI: 10.48550/ARXIV.1112.5745.

[95] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. V. Dillon, B. Lakshminarayanan, and J. Snoek, "Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift", in *Advances in Neural Information Processing Systems (NeurIPS)*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32, Curran Associates, Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2019/file/8558cb408c1d76621371888657d2eb1d-Paper.pdf.

[96] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight uncertainty in neural network", in *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, F. Bach and D. Blei, Eds., ser. Proceedings of Machine Learning Research, vol. 37, Lille, France: PMLR, 2015, pp. 1613–1622. [Online]. Available: https://proceedings.mlr.press/v37/blundell15.html.

[97] M. Hein, M. Andriushchenko, and J. Bitterwolf, "Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2019. DOI: 10.1109/cvpr.2019.00013.

[98] K. P. Murphy, *Probabilistic Machine Learning: Advanced Topics*. MIT press, 2023. [Online]. Available: http://probml.github.io/book2.

[99] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya, V. Makarenkov, and S. Nahavandi, "A review of uncertainty quantification in deep learning: Techniques, applications

and challenges", *Information Fusion*, vol. 76, pp. 243–297, 2021. DOI: 10.1016/j.inffus.2021.05.008.

[100] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer New York, 2006. DOI: 10.1007/978-0-387-45528-0.

[101] Y. Gal, "Uncertainty in deep learning", Ph.D. dissertation, 2016. [Online]. Available: https://www.cs.ox.ac.uk/people/yarin.gal/website/thesis/thesis.pdf.

[102] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles", in *Advances in Neural Information Processing Systems (NeurIPS)*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30, Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/9ef2ed4b7fd2c810847ffa5fa85bce38-Paper.pdf.

[103] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning", in *Proceedings of The 33rd International Conference on Machine Learning (ICML)*, M. F. Balcan and K. Q. Weinberger, Eds., ser. Proceedings of Machine Learning Research, vol. 48, New York, New York, USA: PMLR, 2016, pp. 1050–1059. [Online]. Available: https://proceedings.mlr.press/v48/gal16.html.

[104] A. G. Wilson and P. Izmailov, "Bayesian deep learning and a probabilistic perspective of generalization", in *Advances in Neural Information Processing Systems (NeurIPS)*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33, Curran Associates, Inc., 2020, pp. 4697–4708. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/322f62469c5e3c7dc3e58f5a4d1ea399-Paper.pdf.

[105] A. Ashukha, A. Lyzhov, D. Molchanov, and D. Vetrov, "Pitfalls of in-domain uncertainty estimation and ensembling in deep learning", in *International Conference on Learning Representations (ICLR)*, 2020. [Online]. Available: https://openreview.net/forum?id=BJxI5gHKDr.

[106] Z. Punthakee, R. Goldenberg, and P. Katz, "Definition, classification and diagnosis of diabetes, prediabetes and metabolic syndrome", *Canadian Journal of Diabetes*, vol. 42, pp. 10–15, 2018. DOI: 10.1016/j.jcjd.2017.10.003.

[107] D. J. Magliano, E. J. Boyko, and IDF Diabetes Atlas 10th edition scientific committee, *IDF Diabetes Atlas*, 10th ed. International Diabetes Federation, 2021. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK581934/.

[108] P. Saeedi, I. Petersohn, P. Salpea, B. Malanda, S. Karuranga, N. Unwin, S. Colagiuri, L. Guariguata, A. A. Motala, K. Ogurtsova, J. E. Shaw, D. Bright, and R. Williams, "Global and regional diabetes prevalence estimates for 2019 and projections for 2030 and 2045: Results from the International Diabetes Federation Diabetes Atlas, 9th edition", *Diabetes Research and Clinical Practice*, vol. 157, p. 107 843, 2019. DOI: 10.1016/j.diabres.2019.107843.

[109] Z. L. Teo, Y.-C. Tham, M. Yu, M. L. Chee, T. H. Rim, N. Cheung, M. M. Bikbov, Y. X. Wang, Y. Tang, Y. Lu, I. Y. Wong, D. S. W. Ting, G. S. W. Tan, J. B. Jonas, C. Sabanayagam, T. Y. Wong, and C.-Y. Cheng, "Global prevalence of diabetic retinopathy and projection of burden through 2045", *Ophthalmology*, vol. 128, no. 11, pp. 1580–1591, 2021. DOI: 10.1016/j.ophtha.2021.04.027.

[110] Y. Zheng, S. H. Ley, and F. B. Hu, "Global aetiology and epidemiology of type 2 diabetes mellitus and its complications", *Nature Reviews Endocrinology*, vol. 14, no. 2, pp. 88–98, 2017. DOI: 10.1038/nrendo.2017.151.

[111] G. Wilcox, "Insulin and insulin resistance.", *The Clinical biochemist. Reviews*, vol. 26, pp. 19–39, 2 2005. [Online]. Available: http://www.ncbi.nlm.nih.gov/pmc/articles/pmc1204764/.

[112] American Diabetes Association, "Diagnosis and classification of diabetes mellitus", *Diabetes Care*, vol. 34, no. Supplement_1, S62–S69, 2011. DOI: 10.2337/dc11-s062.

[113] World Health Organization, "Definition and diagnosis of diabetes mellitus and intermediate hyperglycaemia: Report of a WHO/IDF consultation", 2006. [Online]. Available: https://www.who.int/publications/i/item/definition-and-diagnosis-of-diabetes-mellitus-and-intermediate-hyperglycaemia.

[114] P. Hien, B. Böhm, B. Böhm, S. Claudi-Böhm, C. Krämer, and K. Kohlhas, *Diabetes-Handbuch*. Springer Berlin Heidelberg, 2013. DOI: 10.1007/978-3-642-34944-7.

[115] O. Simó-Servat, C. Hernández, and R. Simó, "Diabetic retinopathy in the context of patients with diabetes", *Ophthalmic Research*, vol. 62, pp. 211–217, 4 2019. DOI: 10.1159/000499541.

[116] S. D. Solomon, E. Chew, E. J. Duh, L. Sobrin, J. K. Sun, B. L. VanderBeek, C. C. Wykoff, and T. W. Gardner, "Diabetic retinopathy: A position statement by the american diabetes association", *Diabetes Care*, vol. 40, no. 3, pp. 412–418, 2017. DOI: 10.2337/dc16-2641.

[117] D. S. W. Ting, G. C. M. Cheung, and T. Y. Wong, "Diabetic retinopathy: Global prevalence, major risk factors, screening practices and public health challenges: A review", *Clinical & Experimental Ophthalmology*, vol. 44, no. 4, pp. 260–277, 2016. DOI: 10.1111/ceo.12696.

[118] S. Haider, R. Thayakaran, A. Subramanian, K. A. Toulis, D. Moore, M. J. Price, and K. Nirantharakumar, "Disease burden of diabetes, diabetic retinopathy and their future projections in the UK: Cross-sectional analyses of a primary care database", *BMJ Open*, vol. 11, no. 7, e050058, 2021. DOI: 10.1136/bmjopen-2021-050058.

[119] J. W. Y. Yau, S. L. Rogers, R. Kawasaki, E. L. Lamoureux, J. W. Kowalski, T. Bek, S.-J. Chen, J. M. Dekker, A. Fletcher, J. Grauslund, S. Haffner, R. F. Hamman, M. K. Ikram, T. Kayama, B. E. K. Klein, R. Klein, S. Krishnaiah, K. Mayurasakorn, J. P. O'Hare, T. J. Orchard, M. Porta, M. Rema, M. S. Roy, T. Sharma, J. Shaw, H. Taylor, J. M. Tielsch, R. Varma, J. J. Wang, N. Wang, S. West, L. Xu, M. Yasuda, X. Zhang, P. Mitchell, and T. Y. Wong, "Global prevalence and major risk factors of diabetic retinopathy", *Diabetes Care*, vol. 35, no. 3, pp. 556–564, 2012. DOI: 10.2337/dc11-1909.

[120] D. B. Rein, "The economic burden of major adult visual disorders in the United States", *Archives of Ophthalmology*, vol. 124, no. 12, p. 1754, 2006. DOI: 10.1001/archopht.124.12.1754.

[121] F. Grehn, *Augenheilkunde*. Springer Berlin Heidelberg, 2019. DOI: 10.1007/978-3-662-59154-3.

[122] S. Nian, A. C. Y. Lo, Y. Mi, K. Ren, and D. Yang, "Neurovascular unit in diabetic retinopathy: Pathophysiological roles and potential therapeutical targets", *Eye and Vision*, vol. 8, no. 1, 2021. DOI: 10.1186/s40662-021-00239-1.

[123] P. Walter and N. Plange, *Basiswissen Augenheilkunde*. Springer Berlin Heidelberg, 2017. DOI: 10.1007/978-3-662-52801-3.

[124] A. N. Kollias and M. W. Ulbig, "Diabetic retinopathy", *Dtsch Arztebl International*, vol. 107, pp. 75–84, 5 2010. DOI: 10.3238/arztebl.2010.0075.

[125] Y. Zhou, B. Wang, L. Huang, S. Cui, and L. Shao, "A benchmark for studying diabetic retinopathy: Segmentation, grading, and transferability", *IEEE Transactions on Medical Imaging*, vol. 40, no. 3, pp. 818–828, 2021. DOI: 10.1109/tmi.2020.3037771.

[126] M. D. Davis, L. D. Hubbard, J. Trautman, and R. Klein, "Studies of retinopathy: Methodology for assessment and classification with fundus photographs", *Diabetes*, vol. 34, no. Supplement_3, pp. 42–49, 1985. DOI: 10.2337/diab.34.3.s42.

[127] Early Treatment Diabetic Retinopathy Study Research Group, "Fundus photographic risk factors for progression of diabetic retinopathy: Etdrs report number 12", *Ophthalmology*, vol. 98, no. 5, Supplement, pp. 823–833, 1991. DOI: 10.1016/s0161-6420(13)38014-2.

[128] C. C. Wykoff, R. N. Khurana, Q. D. Nguyen, S. P. Kelly, F. Lum, R. Hall, I. M. Abbass, A. M. Abolian, I. Stoilov, T. M. To, and V. Garmo, "Risk of blindness among patients with diabetes and newly diagnosed diabetic retinopathy", *Diabetes Care*, vol. 44, no. 3, pp. 748–756, 2021. DOI: 10.2337/dc20-0413.

[129] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications", *arXiv*, 2017. DOI: 10.48550/ARXIV.1704.04861.

[130] S. Natarajan, A. Jain, R. Krishnan, A. Rogye, and S. Sivaprasad, "Diagnostic accuracy of community-based diabetic retinopathy screening with an offline artificial intelligence system on a smartphone", *JAMA Ophthalmology*, vol. 137, no. 10, p. 1182, 2019. DOI: 10.1001/jamaophthalmol.2019.2923.

[131] S. Sheikh and U. Qidwai, "Using MobileNetV2 to classify the severity of diabetic retinopathy", *International Journal of Simulation Systems Science & Technology*, 2020. DOI: 10.5013/ijssst.a.21.02.16.

[132] R. Gargeya and T. Leng, "Automated identification of diabetic retinopathy using deep learning", *Ophthalmology*, vol. 124, no. 7, pp. 962–969, 2017. DOI: 10.1016/j.ophtha.2017.02.008.

[133] W. M. Gondal, J. M. Kohler, R. Grzeszick, G. A. Fink, and M. Hirsch, "Weakly-supervised localization of diabetic retinopathy lesions in retinal fundus images", in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, IEEE, 2017. DOI: 10.1109/icip.2017.8296646.

[134] G. Quellec, K. Charrière, Y. Boudi, B. Cochener, and M. Lamard, "Deep image mining for diabetic retinopathy screening", *Medical Image Analysis*, vol. 39, pp. 178–193, 2017. DOI: 10.1016/j.media.2017.04.012.

[135] Q. Wei, X. Li, W. Yu, X. Zhang, Y. Zhang, B. Hu, B. Mo, D. Gong, N. Chen, D. Ding, and Y. Chen, "Learn to segment retinal lesions and beyond", in *Proceedings of the IEEE International Conference on Pattern Recognition (ICPR)*, IEEE, 2021. DOI: 10.1109/icpr48806.2021.9412088.

[136] Y. Yang, T. Li, W. Li, H. Wu, W. Fan, and W. Zhang, "Lesion detection and grading of diabetic retinopathy via two-stages deep convolutional neural networks", in *Lecture Notes in Computer Science*. Springer International Publishing, 2017, pp. 533–540. DOI: 10.1007/978-3-319-66179-7_61.

[137] Z. Wang, Y. Yin, J. Shi, W. Fang, H. Li, and X. Wang, "Zoom-in-Net: Deep mining lesions for diabetic retinopathy detection", in *Medical Image Computing and Computer Assisted Intervention (MICCAI)*, Springer International Publishing, 2017, pp. 267–275. DOI: 10.1007/978-3-319-66179-7_31.

[138] Y. Zhou, X. He, L. Huang, L. Liu, F. Zhu, S. Cui, and L. Shao, "Collaborative learning of semi-supervised segmentation and classification for medical images", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. [Online]. Available: https://openaccess.thecvf.com/content_CVPR_2019/papers/Zhou_Collaborative_Learning_of_Semi-Supervised_Segmentation_and_Classification_for_Medical_Images_CVPR_2019_paper.pdf.

[139] C. Playout, R. Duval, and F. Cheriet, "A novel weakly supervised multitask architecture for retinal lesions segmentation on fundus images", *IEEE Transactions on Medical Imaging*, vol. 38, no. 10, pp. 2434–2444, 2019. DOI: 10.1109/tmi.2019.2906319.

[140] E. Dugas, Jared, Jorge, and W. Cukierski. "Diabetic retinopathy detection". (2015), [Online]. Available: https://kaggle.com/competitions/diabetic-retinopathy-detection (visited on 06/14/2024).

[141] E. Decencière, X. Zhang, G. Cazuguel, B. Lay, B. Cochener, C. Trone, P. Gain, R. Ordonez, P. Massin, A. Erginay, B. Charton, and J.-C. Klein, "Feedback on a publicly distributed image database: The messidor database", *Image Analysis & Stereology*, vol. 33, no. 3, p. 231, 2014. DOI: 10.5566/ias.1155.

[142] F. Chollet, "Xception: Deep learning with depthwise separable convolutions", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2017. DOI: 10.1109/cvpr.2017.195.

[143] S. Guo, T. Li, K. Wang, C. Zhang, and H. Kang, "A lightweight neural network for hard exudate segmentation of fundus image", in *Artificial Neural Networks and Machine Learning – ICANN 2019: Image Processing*, ser. LNCS, vol. 11729 LNCS, Springer International Publishing, 2019, pp. 189–199. DOI: 10.1007/978-3-030-30508-6_16.

[144] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9. DOI: 10.1109/cvpr.2015.7298594.

[145] Z. Yan, X. Han, C. Wang, Y. Qiu, Z. Xiong, and S. Cui, "Learning mutually local-global U-nets for high-resolution retinal lesion segmentation in fundus images", in *Proceedings of the IEEE 16th International Symposium on Biomedical Imaging (ISBI)*, IEEE, 2019. DOI: 10.1109/isbi.2019.8759579.

[146] M. H. Sarhan, S. Albarqouni, M. Yigitsoy, N. Navab, and A. Eslami, "Multi-scale microaneurysms segmentation using embedding triplet loss", in *Medical Image Computing and Computer Assisted Intervention (MICCAI)*, Springer, N. Switzerland, AG, D. Shen, *et al.*, Eds., ser. LNCS, vol. 11764, Springer International Publishing, 2019, pp. 174–182. DOI: 10.1007/978-3-030-32239-7_20.

[147] S. Guo, T. Li, H. Kang, N. Li, Y. Zhang, and K. Wang, "L-seg: An end-to-end unified framework for multi-lesion segmentation of fundus images", *Neurocomputing*, vol. 349, pp. 52–63, 2019. DOI: 10.1016/j.neucom.2019.04.019.

[148] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation", in *Lecture Notes in Computer Science*. Springer International Publishing, 2016, pp. 483–499. DOI: 10.1007/978-3-319-46484-8_29.

[149] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An imperative style, high-performance deep learning library", in *Advances in Neural Information Processing Systems (NeurIPS)*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32, Curran Associates, Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.

[150] L. Zhu. "Lyken17 / pytorch-OpCounter". (2022), [Online]. Available: https://github.com/Lyken17/pytorch-OpCounter (visited on 06/14/2024).

[151] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks", in *Proceedings of the 36th International Conference on Machine Learning (ICML)*, K. Chaudhuri and R. Salakhutdinov, Eds., ser. Proceedings of Machine Learning Research, vol. 97, PMLR, 2019, pp. 6105–6114. [Online]. Available: https://proceedings.mlr.press/v97/tan19a.html.

[152] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection", in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, IEEE, 2017. DOI: 10.1109/iccv.2017.324.

[153] J. Ma, J. Chen, M. Ng, R. Huang, Y. Li, C. Li, X. Yang, and A. L. Martel, "Loss odyssey in medical image segmentation", *Medical Image Analysis*, vol. 71, p. 102 035, 2021. DOI: 10.1016/j.media.2021.102035.

[154] C. H. Sudre, W. Li, T. Vercauteren, S. Ourselin, and M. Jorge Cardoso, "Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations", in *Lecture Notes in Computer Science*. Springer International Publishing, 2017, pp. 240–248. DOI: 10.1007/978-3-319-67558-9_28.

[155] A. Mehrtash, W. M. Wells, C. M. Tempany, P. Abolmaesumi, and T. Kapur, "Confidence calibration and predictive uncertainty estimation for deep medical image segmentation", *IEEE Transactions on Medical Imaging*, vol. 39, no. 12, pp. 3868–3878, 2020. DOI: 10.1109/tmi.2020.3006437.

[156] M. Yeung, E. Sala, C.-B. Schönlieb, and L. Rundo, "Unified focal loss: Generalising dice and cross entropy-based losses to handle class imbalanced medical image segmentation", *Computerized Medical Imaging and Graphics*, vol. 95, p. 102 026, 2022. DOI: 10.1016/j.compmedimag.2021.102026.

[157] F. Hausdorff, *Grundzüge der Mengenlehre*. Veit & Comp., 1914. [Online]. Available: https://archive.org/details/grundzgedermen00hausuoft.

[158] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge, "Comparing images using the hausdorff distance", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 9, pp. 850–863, 1993. DOI: 10.1109/34.232073.

[159] M. Tan and Q. Le, "EfficientNetV2: Smaller models and faster training", in *Proceedings of the 38th International Conference on Machine Learning (ICML)*, M. Meila and T. Zhang, Eds., ser. Proceedings of Machine Learning Research, vol. 139, PMLR, 2021, pp. 10 096–10 106. [Online]. Available: https://proceedings.mlr.press/v139/tan21a.html.

[160] G. Lu, W. Zhang, and Z. Wang, "Optimizing depthwise separable convolution operations on GPUs", *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 1, pp. 70–87, 2022. DOI: 10.1109/tpds.2021.3084813.

[161] P. Porwal, S. Pachade, M. Kokare, G. Deshmukh, J. Son, W. Bae, L. Liu, J. Wang, X. Liu, L. Gao, T. Wu, J. Xiao, F. Wang, B. Yin, Y. Wang, G. Danala, L. He, Y. H. Choi, Y. C. Lee, S.-H. Jung, Z. Li, X. Sui, J. Wu, X. Li, T. Zhou, J. Toth, A. Baran,

A. Kori, S. S. Chennamsetty, M. Safwan, V. Alex, X. Lyu, L. Cheng, Q. Chu, P. Li, X. Ji, S. Zhang, Y. Shen, L. Dai, O. Saha, R. Sathish, T. Melo, T. Araújo, B. Harangi, B. Sheng, R. Fang, D. Sheet, A. Hajdu, Y. Zheng, A. M. Mendonça, S. Zhang, A. Campilho, B. Zheng, D. Shen, L. Giancardo, G. Quellec, and F. Mériaudeau, "IDRiD: Diabetic retinopathy – segmentation and grading challenge", *Medical Image Analysis*, vol. 59, p. 101 561, 2020. DOI: 10.1016/j.media.2019.101561.

[162] T. Li, Y. Gao, K. Wang, S. Guo, H. Liu, and H. Kang, "Diagnostic assessment of deep learning algorithms for diabetic retinopathy screening", *Information Sciences*, vol. 501, pp. 511–522, 2019. DOI: 10.1016/j.ins.2019.06.011.

[163] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 8, pp. 2011–2023, 2020. DOI: 10.1109/tpami.2019.2913372.

[164] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding", in *International Conference on Learning Representations (ICLR)*, 2016. DOI: 10.48550/ARXIV.1510.00149.

[165] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network", *arXiv*, 2015. DOI: 10.48550/ARXIV.1503.02531.

[166] M. Siebert, N. Tesmer, and P. Rostalski, "Stochastic variational deep kernel learning based diabetic retinopathy severity grading", 2, vol. 8, Walter de Gruyter GmbH, 2022, pp. 408–411. DOI: 10.1515/cdbme-2022-1104.

[167] N. Band, T. G. J. Rudner, Q. Feng, A. Filos, Z. Nado, M. W. Dusenberry, G. Jerfel, D. Tran, and Y. Gal, "Benchmarking Bayesian deep learning on diabetic retinopathy detection tasks", in *Workshop on Distribution Shifts: Connecting Methods and Applications at the Conference on Neural Information Processing Systems (NeurIPS)*, 2021. [Online]. Available: https://openreview.net/forum?id=uJ2_JTpVCvc.

[168] J. Jaskari, J. Sahlsten, T. Damoulas, J. Knoblauch, S. Sarkka, L. Karkkainen, K. Hietala, and K. K. Kaski, "Uncertainty-aware deep learning methods for robust diabetic retinopathy classification", *IEEE Access*, vol. 10, pp. 76 669–76 681, 2022. DOI: 10.1109/access.2022.3192024.

[169] T. Araújo, G. Aresta, L. Mendonça, S. Penas, C. Maia, Â. Carneiro, A. M. Mendonça, and A. Campilho, "DR|GRADUATE: Uncertainty-aware deep learning-based diabetic retinopathy grading in eye fundus images", *Medical Image Analysis*, vol. 63, p. 101 715, 2020. DOI: 10.1016/j.media.2020.101715.

[170] C. Leibig, V. Allken, M. S. Ayhan, P. Berens, and S. Wahl, "Leveraging uncertainty information from deep neural networks for disease detection", *Scientific Reports*, vol. 7, no. 1, 2017. DOI: 10.1038/s41598-017-17876-z.

[171] A. G. Wilson, Z. Hu, R. R. Salakhutdinov, and E. P. Xing, "Stochastic variational deep kernel learning", in *Advances in Neural Information Processing Systems (NeurIPS)*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29, Curran Associates, Inc., 2016. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2016/file/bcc0d400288793e8bdcd7c19a8ac0c2b-Paper.pdf.

[172] S. W. Ober, C. E. Rasmussen, and M. van der Wilk, "The promises and pitfalls of deep kernel learning", in *Proceedings of the 37th Conference on Uncertainty in Artificial Intelligence (UAI)*, C. de Campos and M. H. Maathuis, Eds., ser. Proceedings of Machine Learning Research, vol. 161, PMLR, 2021, pp. 1206–1216. [Online]. Available: https://proceedings.mlr.press/v161/ober21a.html.

[173] J. Z. Liu, S. Padhy, J. Ren, Z. Lin, Y. Wen, G. Jerfel, Z. Nado, J. Snoek, D. Tran, and B. Lakshminarayanan, "A simple approach to improve single-model deep uncertainty via distance-awareness", *Journal of Machine Learning Research*, vol. 24, no. 42, pp. 1–63, 2023. [Online]. Available: http://jmlr.org/papers/v24/22-0479.html.

[174] J. van Amersfoort, L. Smith, A. Jesson, O. Key, and Y. Gal, "On feature collapse and deep kernel learning for single forward pass uncertainty", in *Workshop on Bayesian Deep Learning at the Conference on Neural Information Processing Systems (NeurIPS)*, 2021. [Online]. Available: http://bayesiandeeplearning.org/2021/papers/28.pdf.

[175] S. Pachade, P. Porwal, D. Thulkar, M. Kokare, G. Deshmukh, V. Sahasrabuddhe, L. Giancardo, G. Quellec, and F. Mériaudeau, "Retinal fundus multi-disease image dataset (RFMiD): A dataset for multi-disease detection research", *Data*, vol. 6, no. 2, p. 14, 2021. DOI: 10.3390/data6020014.

[176] V. Rotemberg, N. Kurtansky, B. Betz-Stablein, L. Caffery, E. Chousakos, N. Codella, M. Combalia, S. Dusza, P. Guitera, D. Gutman, A. Halpern, B. Helba, H. Kittler, K. Kose, S. Langer, K. Lioprys, J. Malvehy, S. Musthaq, J. Nanda, O. Reiter, G. Shih, A. Stratigos, P. Tschandl, J. Weber, and H. P. Soyer, "A patient-centric dataset of images and metadata for identifying melanomas using clinical context", *Scientific Data*, vol. 8, p. 34, 1 2021. DOI: 10.1038/s41597-021-00815-z. [Online]. Available: https://www.nature.com/articles/s41597-021-00815-z.

[177] B. Graham, "Kaggle diabetic retinopathy detection competition report", 2015. [Online]. Available: https://www.kaggle.com/competitions/diabetic-retinopathy-detection/discussion/15801.

[178] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning* (Adaptive computation and machine learning), 3rd ed. Cambridge, Mass. [u.a.]: MIT Press, 2006, 248 pp. [Online]. Available: https://www.GaussianProcess.org/gpml.

[179] R. Calandra, J. Peters, C. E. Rasmussen, and M. P. Deisenroth, "Manifold Gaussian processes for regression", in *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2016. DOI: 10.1109/ijcnn.2016.7727626.

[180] N. S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, and P. T. P. Tang, "On large-batch training for deep learning: Generalization gap and sharp minima", in *International Conference on Learning Representations (ICLR)*, 2017. [Online]. Available: https://openreview.net/forum?id=H1oyRlYgg.

[181] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2018. DOI: 10.1109/cvpr.2018.00474.

[182] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, "Efficient object localization using convolutional networks", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[183] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger, "Deep networks with stochastic depth", in *Lecture Notes in Computer Science*. Springer International Publishing, 2016, pp. 646–661. DOI: 10.1007/978-3-319-46493-0_39.

[184] J. Antoran, J. Allingham, and J. M. Hernández-Lobato, "Depth uncertainty in neural networks", in *Advances in Neural Information Processing Systems (NeurIPS)*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33, Curran Associates, Inc., 2020, pp. 10 620–10 634. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/781877bda0783aac5f1cf765c128b437-Paper.pdf.

[185] J. R. Gardner, G. Pleiss, D. Bindel, K. Q. Weinberger, and A. G. Wilson, "GPyTorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration", in *Advances in Neural Information Processing Systems (NeurIPS)*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31, Curran Associates, Inc., 2018. [Online]. Available: https://proceedings.neurips.cc/

paper_files/paper/2018/file/27e8e17134dd7083b050476733207ea1-Paper.pdf.

[186] R. T. Q. Chen, J. Behrmann, D. K. Duvenaud, and J.-H. Jacobsen, "Residual flows for invertible generative modeling", in *Advances in Neural Information Processing Systems (NeurIPS)*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32, Curran Associates, Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2019/file/5d0d5594d24f0f955548f0fc0ff83d10-Paper.pdf.

[187] E. Bakshy, L. Dworkin, B. Karrer, K. Kashin, B. Letham, A. Murthy, and S. Singh, "AE: A domain-agnostic platform for adaptive experimentation", in *Workshop on Systems for ML at the Conference on Neural Information Processing Systems (NeurIPS)*, Montréal, Canada, 2018. [Online]. Available: http://learningsys.org/nips18/assets/papers/87CameraReadySubmissionAE%20-%20NeurIPS%202018.pdf.

[188] J. Cohen, "Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit.", *Psychological Bulletin*, vol. 70, no. 4, pp. 213–220, 1968. DOI: 10.1037/h0026256.

[189] J. Cohen, "A coefficient of agreement for nominal scales", *Educational and Psychological Measurement*, vol. 20, no. 1, pp. 37–46, 1960. DOI: 10.1177/001316446002000104.

[190] M. P. Naeini, G. Cooper, and M. Hauskrecht, "Obtaining well calibrated probabilities using Bayesian binning", *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, 2015. DOI: 10.1609/aaai.v29i1.9602.

[191] M. S. A. Nadeem, J.-D. Zucker, and B. Hanczar, "Accuracy-rejection curves (arcs) for comparing classification methods with a reject option", in *Proceedings of the 3rd International Workshop on Machine Learning in Systems Biology*, S. Džeroski, P. Guerts, and J. Rousu, Eds., ser. Proceedings of Machine Learning Research, vol. 8, Ljubljana, Slovenia: PMLR, 2009, pp. 65–81. [Online]. Available: https://proceedings.mlr.press/v8/nadeem10a.html.

[192] D. Ries, J. Michalenko, T. Ganter, R. I.-F. Baiyasi, and J. Adams, "Comparing the quality of neural network uncertainty estimates for classification problems", in *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA)*, IEEE, 2022. DOI: 10.1109/icmla55696.2022.00039.

[193] A. Krizhevsky, "Learning multiple layers of features from tiny images", 2009. [Online]. Available: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf (visited on 06/14/2024).

[194] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning", in *Workshop on Deep Learning and Unsupervised Feature Learning at the Conference on Neural Information Processing Systems (NeurIPS)*, 2011. [Online]. Available: http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf.

[195] P. Qian, Z. Zhao, C. Chen, Z. Zeng, and X. Li, "Two eyes are better than one: Exploiting binocular correlation for diabetic retinopathy severity grading", in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, 2021. DOI: 10.1109/embc46164.2021.9630812.

[196] J. Sahlsten, J. Jaskari, J. Kivinen, L. Turunen, E. Jaanio, K. Hietala, and K. Kaski, "Deep learning fundus image analysis for diabetic retinopathy and macular edema grading", *Scientific Reports*, vol. 9, no. 1, 2019. DOI: 10.1038/s41598-019-47181-w.

[197] M. Chetoui and M. A. Akhloufi, "Explainable diabetic retinopathy using EfficientNET", in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, 2020. DOI: 10.1109/embc44109.2020.9175664.

[198] J. Nixon, M. W. Dusenberry, L. Zhang, G. Jerfel, and D. Tran, "Measuring calibration in deep learning", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2019. [Online]. Available: https://openaccess.thecvf.com/content_CVPRW_2019/papers/Uncertainty%20and%20Robustness%20in%20Deep%20Visual%20Learning/Nixon_Measuring_Calibration_in_Deep_Learning_CVPRW_2019_paper.pdf.

[199] M.-H. Laves, S. Ihler, K.-P. Kortmann, and T. Ortmaier, "Well-calibrated model uncertainty with temperature scaling for dropout variational inference", in *Workshop on Bayesian Deep Learning at the Conference on Neural Information Processing Systems (NeurIPS)*, 2019. [Online]. Available: http://bayesiandeeplearning.org/2019/papers/77.pdf.

[200] R. Krishnan and O. Tickoo, "Improving model calibration with accuracy versus uncertainty optimization", in *Advances in Neural Information Processing Systems (NeurIPS)*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33, Curran Associates, Inc., 2020, pp. 18 237–18 248. [Online]. Available: https:

//proceedings.neurips.cc/paper_files/paper/2020/file/d3d9446802a44259755d38e6d163e820-Paper.pdf.

[201] P. Ruamviboonsuk, J. Krause, P. Chotcomwongse, R. Sayres, R. Raman, K. Widner, B. J. L. Campana, S. Phene, K. Hemarat, M. Tadarati, S. Silpa-Archa, J. Limwattanayingyong, C. Rao, O. Kuruvilla, J. Jung, J. Tan, S. Orprayoon, C. Kangwanwongpaisan, R. Sukumalpaiboon, C. Luengchaichawang, J. Fuangkaew, P. Kongsap, L. Chualinpha, S. Saree, S. Kawinpanitan, K. Mitvongsa, S. Lawanasakol, C. Thepchatri, L. Wongpichedchai, G. S. Corrado, L. Peng, and D. R. Webster, "Deep learning versus human graders for classifying diabetic retinopathy severity in a nationwide screening program", *npj Digital Medicine*, vol. 2, no. 1, 2019. DOI: 10.1038/s41746-019-0099-8.

[202] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[203] Y. Wen, D. Tran, and J. Ba, "Batchensemble: An alternative approach to efficient ensemble and lifelong learning", in *International Conference on Learning Representations (ICLR)*, 2020. [Online]. Available: https://openreview.net/forum?id=Sklf1yrYDr.

[204] H. Chen and A. Shrivastava, "Group ensemble: Learning an ensemble of convnets in a single convnet", *arXiv*, 2020. DOI: 10.48550/ARXIV.2007.00649.

[205] Y. Shen, Z. Zhang, M. R. Sabuncu, and L. Sun, "Real-time uncertainty estimation in computer vision via uncertainty-aware distribution distillation", in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2021, pp. 707–716.

[206] K. M. Collins, M. Barker, M. Espinosa Zarlenga, N. Raman, U. Bhatt, M. Jamnik, I. Sucholutsky, A. Weller, and K. Dvijotham, "Human uncertainty in concept-based ai systems", in *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society (AIES)*, ACM, 2023. DOI: 10.1145/3600211.3604692.

[207] A. Mahinpei, J. Clark, I. Lage, F. Doshi-Velez, and W. Pan, "Promises and pitfalls of black-box concept learning models", in *Workshop on Theoretic Foundation, Criticism, and Application Trend of Explainable AI at the International Conference on Machine Learning (ICML)*, 2021. DOI: 10.48550/arxiv.2106.13314.

List of Publications

Journal Articles

2024 J. Sauer, **M. Siebert**, L. Boudnik, N. M. Carbon, S. Walterspacher, P. Rostalski, "Signal quality evaluation of single-channel respiratory sEMG recordings," *Biomedical Signal Processing and Control*, no. 87, pp. 105414, 2024.

2023 **M. Siebert**, J. Graßhoff, P. Rostalski, "Uncertainty Analysis of Deep Kernel Learning Methods on Diabetic Retinopathy Grading," *IEEE Access*, vol. 11, pp. 146173–146184, 2023.

2020 G. Männel, **M. Siebert**, D. Kleinewalter, C. Brendle, and P. Rostalski, "Robust Model Predictive Control of an Anaesthesia Workstation Ventilation Unit," *at - Automatisierungstechnik*, vol. 68, no. 11, pp.941–952, 2020.

2019 L. Hansen, **M. Siebert**, J. Diesel, and M. Heinrich, "Fusing information from multiple 2D depth cameras for 3D human pose estimation in the operating room," *International Journal of Computer Assisted Radiology and Surgery*, vol. 14, no.11, pp. 1871–1879, 2019.

2018 C. Herzog, O. Thomsen, B. Schmarbeck, **M. Siebert**, and R. Brinkmann, "Temperature-controlled laser therapy of the retina via robust adaptive $\mathcal{H}_\infty$-control," *at - Automatisierungstechnik*, vol. 66, no.12, pp. 1051–1063, 2018.

Conference Papers

2022 **M. Siebert**, N. Tesmer, and P. Rostalski, "Stochastic variational deep kernel learning based diabetic retinopathy severity grading," in *Current Directions in Biomedical Engineering*, Walter De Gruyter GmbH, 2022, pp. 408–411.

2022 **M. Siebert**, and P. Rostalski, "Performance evaluation of lightweight convolutional neural networks on retinal lesion segmentation," in *Medical Imaging 2022: Computer-Aided Diagnosis*, K. Drukker and K. M. Iftekharuddin, Eds., SPIE, 2022, p. 1203334.

2020 G. Männel, **M. Siebert**, D. Kleinewalter, C. Brendle, and P. Rostalski, "Model Predictive Control of an Anaesthesia Workstation Ventilation Unit," in *IFAC-PapersOnLine*, Elsevier BV, 2020, pp. 6644–6649.

2020 G. Männel, **M. Siebert**, and P. Rostalski, "Tube-based MPC for Pressure Controlled Ventilation," in *Proceedings on Automation in Medical Engineering*, 2020.