Unveiling the Ace in the Hole: Leveraging Uncertainty Quantification for Computer Vision Systems

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät

der Eberhard Karls Universität Tübingen

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

vorgelegt von

Uddeshya Upadhyay

aus Geburtsort

Kochin, Kerala, Indien

Tübingen

2023

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	09.11.2023
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter/-in:	Prof. Dr. Zeynep Akata
2. Berichterstatter/-in:	Prof. Dr. Matthias Hein

DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF TÜBINGEN & INTERNATIONAL MAX PLANCK RESEARCH SCHOOL FOR INTELLIGENT SYSTEMS

UNVEILING THE ACE IN THE HOLE: LEVERAGING UNCERTAINTY QUANTIFICATION FOR COMPUTER VISION SYSTEMS

UDDESHYA UPADHYAY

M.Tech. & B.Tech. in Computer Science and Engineering

Adviser: Zeynep Akata
Full Professor, University of Tübingen

Co-adviser: Sergios Gatidis
Associate Professor, Stanford University

Examination Committee

Chair: Philipp Hennig
Full Professor, University of Tübingen

Members: Zeynep Akata
Full Professor, University of Tübingen

Ulrike von Luxburg
Full Professor, University of Tübingen

Matthias Hein
Full Professor, University of Tübingen

DOCTORATE IN COMPUTER SCIENCE

University of Tübingen &
International Max Planck Research School for Intelligent Systems
, 2023

Unveiling the Ace in the Hole: Leveraging Uncertainty Quantification for Computer Vision Systems

This document was created with the (pdf/Xe/Lua)LaTeX processor and the NOVAthesis template (v7.0.3) [**novathesis-manual**].

Acknowledgements

I would like to express my heartfelt gratitude to my advisor, Prof. Zeynep Akata, for her unwavering support and guidance throughout the course of my Ph.D. at the University of Tuebingen and the International Max Planck Research School for Intelligent Systems. Her wisdom, patience, and mentorship have been instrumental and have profoundly impacted my Ph.D. journey and personal growth. I am also deeply grateful to the entire faculty and staff at the University and Research School, whose commitment to fostering a stimulating and supportive academic environment is truly remarkable. I would particularly like to thank my colleagues in the lab, who have contributed to my research through their insightful discussions and feedback.

A special mention goes to my friends, Jaivardhan Kapoor and Shyamgopal Karthik. You have been a source of camaraderie and support throughout this journey. Our shared moments of all-nighters and early-morning (extremely sorry Jai and Shyam!) brainstorming, problem-solving, and the occasional light-hearted (and sometimes extremely intense) banter have indeed made the rigors of research less daunting and more enjoyable. I am also grateful to Prof. Suyash P. Awate at IIT-Bombay, my undergraduate advisor, who first introduced me to the fascinating world of research. His unique approach to teaching, emphasizing the importance of precise scientific communication and the use of mathematics as a language, has been instrumental in laying the groundwork for my research endeavors. A huge thanks to Stan Lee, Robert Downey Jr., and Marvel Cinematic Universe (MCU) for making Tony Stark (Ironman) such a cool character; that inspired me as a kid to get into Science & Technology and made the idea of pursuing doctoral studies and getting a Ph.D. very cool.

I am indebted to my family for their enduring love, encouragement, and faith in my capabilities. Their unwavering belief in my potential has been a constant source of strength, helping me navigate the many challenges that have come my way. I am also grateful to all those who, directly or indirectly, have enriched my doctoral experience. It is a journey that I have not walked alone, and this acknowledgment is a humble expression of my appreciation for the collective effort and kindness that have made it possible.

Thank you.

Abstract

As machine learning systems become increasingly complex and autonomous, the integration of uncertainty quantification becomes crucial, especially in high-stakes domains like healthcare and autonomous driving, where ambiguity can lead to severe consequences. By offering a clear gauge of prediction confidence, uncertainty quantification supports informed decision-making and risk management.

Within the realm of healthcare, where diagnostic procedures often depend on various imaging modalities, modern machine-learning methods are being harnessed to aid diagnosis. Current advancements in generative machine learning explore the synthesis of different medical imaging modalities, predominantly through image-to-image translations. Our work demonstrates that integrating aleatoric uncertainty in Generative Adversarial Networks (GANs) for these translation tasks can amplify interpretability and accuracy. Consequently, this empowers healthcare professionals with better diagnostic and treatment decisions, thus enhancing patient outcomes.

In the context of autonomous driving and similar applications, ensuring resilience to unforeseen perturbations is vital. Traditional deterministic models may falter when confronted with new situations, constituting a safety hazard. We address this by implementing a probabilistic approach to dense computer vision tasks and utilizing the Likelihood Annealing technique for uncertainty estimation. These methods amplify the robustness to unexpected situations and provide a calibrated uncertainty measure, contributing to the development of safer autonomous systems.

While creating new probabilistic machine learning solutions for vital applications is a key research area, it's equally significant to develop methods that leverage large-scale pretrained models. These deterministic models can be adapted to estimate uncertainties in a cost-efficient manner regarding data, computation, and other resources, a direction we explore in this thesis. The work presented herein addresses this issue within the context of current computer vision systems, including large-scale vision language models crucial for enabling intelligent multimodal systems.

Zusammenfassung

Übersetzung mit "Google Translate". Mit zunehmender Komplexität und Autonomie von maschinellen Lernsystemen wird die Integration der Unsicherheitsquantifizierung unerlässlich, insbesondere in hochriskanten Bereichen wie dem Gesundheitswesen und dem autonomen Fahren, wo Unklarheiten schwerwiegende Folgen haben können. Durch die Bereitstellung eines klaren Maßes für die Vorhersagegenauigkeit unterstützt die Unsicherheitsquantifizierung fundierte Entscheidungen und Risikomanagement. Bereich der Gesundheitsversorgung, in dem diagnostische Verfahren oft auf verschiedenen Bildgebungsmodalitäten beruhen, werden moderne maschinelle Lernmethoden zur Unterstützung der Diagnose eingesetzt. Aktuelle Fortschritte im generativen maschinellen Lernen erforschen die Synthese verschiedener medizinischer Bildgebungsmodalitäten, hauptsächlich durch Bild-zu-Bild-Übersetzungen. Unsere Arbeit zeigt, dass die Integration von aleatorischer Unsicherheit in Generative Adversarial Networks (GANs) für diese Übersetzungsaufgaben die Interpretierbarkeit und Genauigkeit erhöhen kann. Dies ermöglicht es medizinischen Fachleuten, bessere diagnostische und therapeutische Entscheidungen zu treffen und somit die Patientenergebnisse zu verbessern. Kontext des autonomen Fahrens und ähnlicher Anwendungen ist die Gewährleistung der Widerstandsfähigkeit gegen unvorhergesehene Störungen von entscheidender Bedeutung. Traditionelle deterministische Modelle können versagen, wenn sie mit neuen Situationen konfrontiert werden, was ein Sicherheitsrisiko darstellt. Dies wird durch die Anwendung eines probabilistischen Ansatzes für dichte Computer Vision-Aufgaben und die Verwendung der Likelihood Annealing-Technik zur Unsicherheitsschätzung adressiert. Diese Methoden erhöhen die Robustheit gegenüber unerwarteten Situationen und liefern ein kalibriertes Unsicherheitsmaß, das zur Entwicklung sichererer autonomer Systeme beiträgt. Während die Entwicklung neuer probabilistischer maschineller Lernlösungen für wichtige Anwendungen ein Schlüsselbereich der Forschung ist, ist es ebenso wichtig, Methoden zu entwickeln, die großskalige vortrainierte Modelle nutzen. Diese deterministischen Modelle können angepasst werden, um Unsicherheiten auf kosteneffiziente Weise in Bezug auf Daten, Berechnungen und andere Ressourcen zu schätzen, eine Richtung, die wir in dieser Dissertation untersuchen. Die hier vorgestellten Arbeiten befassen sich mit diesem Thema im Kontext aktueller Computersichtsysteme, einschließlich großskaliger Vision-Sprachmodelle, die für die Ermöglichung intelligenter multimodaler Systeme unerlässlich sind.

Contents

List of Figures

List of Tables

Listings

1 Thesis Introduction

1.1 Motivation

The growing demand for intelligent systems in healthcare, autonomous driving, climate & weather predictions, financial forecasting systems, etc., calls for innovative techniques based on probabilistic machine learning that allows uncertainty quantification in the predictions made by the models that can potentially be extremely useful in designing robust models and also triggering human expert interventions for highly unreliable predictions, preventing fatal outcomes.

For instance, consider healthcare, where the diagnostic process often relies on imaging modalities. Recent works in the field often seek to apply modern machine-learning methods to help in the diagnosis. In fact, recent advancements in generative machine learning are being explored to synthesize various medical imaging modalities that eventually help in diagnosis. However, widely used deep learning-based methods are often deterministic in nature and do not provide a mechanism to flag a wrong/highly-unreliable prediction post-deployment in the real world. The presence of such a mechanism which can correctly quantify the uncertainty in the predictions by machine learning models is critical in designing human-in-the-loop systems with real-world experts. For instance, in a healthcare setting, this may look like the following: A highly accurate probabilistic machine learning model that produces well-calibrated uncertainty estimates may help healthcare practitioners in two ways. (i) The high accuracy of the model may allow significant automation in the diagnosis pipeline reducing the burden on the practitioners. (ii) The well-calibrated uncertainty estimates may allow flagging of erroneous or unreliable predictions that get diverted to human experts, demanding their attention only in critical cases, again easing the workload on healthcare practitioners while being cognizant about efficient use of resources in critical scenarios.

Furthermore, many applications in the real world are often embedded in an evolving environment that may change the nature or quality of the input data over time, also known as "data drift" a machine learning model trained with a static dataset will potentially degrade over a time as the data drift becomes stronger. In a healthcare setting, this may

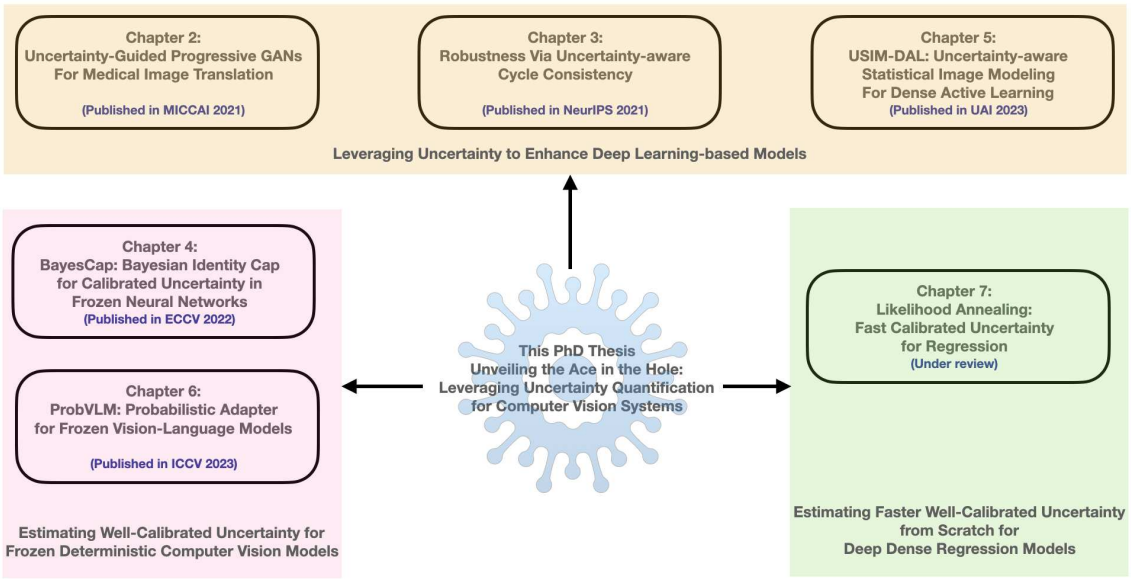

Figure 1.1: Conceptual relation between different chapters of this thesis (along with their publication venues). Chapters 2, 3, and 5 discuss various methods where uncertainty quantification can help in the existing deep learning-based methods for various problems in computer vision. Chapters 4 and 6 discuss methods to leverage pretrained large-scare deterministic computer vision models and estimate the uncertainty for the same efficiently. Chapter 7 discusses a faster method to estimate the uncertainty of the deep neural network.

manifest in several forms. For example, the model may be trained with certain demography of the patients, but over time the demography of incoming patients at the healthcare center, where the original model is deployed, changes. Even in these scenarios, if the model can quantify the predictive uncertainty, then one can potentially use it to detect substantial drift in the data and trigger retraining the model with an updated dataset in order to have a more robust and reliable model.

Above mentioned scenarios are some of the examples highlighting the importance of well-calibrated uncertainty quantification in machine learning-based solutions for critical applications. Therefore, studying and designing such systems are essential for modern machine-learning models, and this thesis highlights several contributions in this direction.

1.2 Contributions

This thesis primarily deals with tackling uncertainty quantification in the computer vision domain. It highlights novel use cases of the derived uncertainties and proposes new ways of estimating the uncertainty in some important concepts in computer vision. Figure 1.1 shows the concept map for this thesis, i.e., how different chapters of the thesis are related (along with the venues where these chapters are published).

A variety of tasks are considered in the course of work, including image translation, image enhancement, out-of-distribution detection, and robustness. Moreover, we also consider recent advanced large-scale multimodal models such as vision-language models

(VLMs). In the following sections, we discuss several lines of contributions.

First, we consider an important class of problems in computer vision called image-to-image translation. These applied problems have often been tackled using generative models like generative adversarial networks (GANs). In particular, we consider medical image-to-image translation tasks using GANs that have shown impressive outcomes. However, conventional GAN-based frameworks lack the ability to estimate the uncertainties in the predictions made by the network, a key aspect in making informed medical decisions. To overcome this limitation, our work proposes an uncertainty-guided progressive learning scheme for image-to-image translation. It effectively incorporates aleatoric uncertainty as attention maps for GANs that are trained by progressively focussing on the region in the synthesized images that are highly unreliable (i.e., high uncertainty), achieving superior performance in several medical image translation tasks.

Second, building on the concept of image-to-image translation, it's essential to address the issue of learning inter-image-domain mapping without corresponding image pairs, a process known as unpaired image-to-image translation. Existing methods often fail to account for robustness to outliers, causing performance degradation in the face of unseen perturbations. Our work proposes a novel probabilistic method based on Uncertainty-aware Generalized Adaptive Cycle Consistency (UGAC) to address this issue, exhibiting significant robustness towards unseen perturbations in test data.

Third, while probabilistic machine learning techniques are valuable in estimating uncertainty, training them on large-scale datasets remains a challenge, especially for many problems in computer vision, often failing to deliver models competitive with non-Bayesian counterparts. Our work proposes a novel method named "BayesCap", offering a memory-efficient solution that can be trained on a small fraction of the original dataset on top of a large-scale pretrained deterministic computer vision model. This not only equips pretrained non-Bayesian computer vision models with calibrated uncertainty estimates but also does so without hampering the model's performance or necessitating expensive retraining.

Fourth, while uncertainty estimates have been incorporated in active learning for classification settings and scalar regression settings, however, it has not been studied if uncertainty estimates can be used to design active learning methods for dense regression tasks in computer vision (e.g., image-to-image translation). Our work proposes an active learning method for dense regression tasks in computer vision models which leverages the statistical properties of color images to learn informative priors that help in quantifying uncertainty and serves as a proxy for error, guiding the active learning process. The active learning method provides a promising solution for addressing the high cost of annotation and labeling in computer vision, demonstrated for applications in medical imaging and remote sensing.

Fifth, our proposed work also explores probabilistic approaches in the realm of vision-language models (VLMs), with a focus on estimating probability distributions for the embeddings of pre-trained VLMs. By aligning inter/intra-modal, our proposed work

accurately estimates multi-modal embedding uncertainties. Furthermore, estimated uncertainty aids active learning and model selection, extending the model's usability.

Lastly, our work highlights a method that learns to estimate calibrated uncertainty for regression tasks with improved convergence of deep regression models, providing calibrated uncertainty without any post hoc calibration phase. This is a significant improvement on the conventional optimization problem involved in training a model capable of estimating regression uncertainty, which often produces poorly calibrated uncertainty estimates that need to be corrected in a post hoc fashion.

In summary, this thesis explores various methods of quantifying uncertainty in widely applicable machine learning-based computer vision methods and leveraging the derived uncertainty estimates to enhance the capabilities and/or performance of machine learning-based computer vision methods.

1.3 Outline

This section briefly overviews every thesis chapter, referencing respective publications and collaborations, including their contribution to the overall work. Four chapters, i.e., 2,3,4,5, correspond to the published content, and chapter 6&7 corresponds to two under-review preprints. All the publications are first-author or shared first-author publications.

Chapter 1: Thesis introduction.

This chapter motivates uncertainty quantification in machine learning-based computer vision methods, along with the summary of contributions and the outline for the thesis.

Chapter 2: Uncertainty-Guided Progressive GANs for Medical Image Translation.

This chapter corresponds to a published paper at MICCAI 2021, where Tobias Hepp and Sergios Gatidis played medical advisors, whereas Yabei Chen and Zeynep Akata played machine learning advisors. The chapter presents a novel approach to improving the performance of generative adversarial networks (GANs) in medical imaging. It proposes an uncertainty-guided progressive learning scheme for image-to-image translation where aleatoric uncertainty estimates are incorporated as attention maps, allowing the GAN to produce progressively high-fidelity images, leading to improvements in various medical image translation tasks.

Chapter 3: Robustness via Uncertainty-aware Cycle Consistency.

This chapter corresponds to a published paper at NeurIPS 2021, where Yabei Chen and Zeynep Akata played advisors. It introduces a novel probabilistic method for unpaired image-to-image translation. It explains how the model, capable of handling heavy-tailed distributions, enhances robustness to outliers and unseen perturbations in test data. Comparisons with other state-of-the-art methods on diverse tasks further highlight the strengths of the proposed approach.

Chapter 4: BayesCap: Bayesian Identity Cap for Calibrated Uncertainty in Frozen Neural Networks.

This chapter corresponds to a published paper at ECCV 2022, where Shyamgopal Karthik was the joint first author, and Massimilano Mancini, Yabei Chen, and Zeynep Akata played advisors. It proposes BayesCap, a method that allows the estimation of uncertainty in pre-trained frozen deterministic deep-learning computer vision models. This approach works by learning a Bayesian identity mapping for pretrained models, enabling the provision of calibrated uncertainty estimates. The method emphasizes BayesCap's memory-efficient nature and ability to enhance various computer vision models.

Chapter 5: `USIM-DAL`: Uncertainty-aware Statistical Image Modeling-based Dense Active Learning for Super-resolution

This chapter corresponds to a published paper at UAI 2023, where Vikrant Rangnekar was the joint first author, and Biplab Banerjee and Zeynep Akata played advisors. It dives into the intersection of active learning and dense regression models. It proposes a new framework, USIM-DAL, that uses probabilistic deep neural networks and aleatoric uncertainty for active learning in high-dimensional computer vision regression tasks. The effectiveness of this approach is demonstrated through a variety of applications, including natural images, medical imaging, and remote sensing.

Chapter 6: `ProbVLM`: Probabilistic Adapter for Frozen Vison-Language Models.

This chapter corresponds to a preprint (under review), where Shyamgopal Karthik was the joint first author, and Massimilano Mancini and Zeynep Akata played advisors. This chapter introduces ProbVLM, a probabilistic adapter for pretrained vision-language models (VLMs). It details how the method estimates multi-modal embedding uncertainties. Furthermore, this chapter explores the use of estimated uncertainty for active learning and model selection in real-world tasks using vision-language models.

Chapter 7: Likelihood Annealing: Fast Calibrated Uncertainty for Regression.

This chapter corresponds to a preprint (under review), where Jae Myung Kim, Cordelia Schmidt, Bernhard Schölkopf, and Zeynep Akata played advisors. This presents a fast-calibrated uncertainty estimation method for regression tasks named Likelihood Annealing. It explains how the approach accelerates the convergence of deep regression models and provides calibrated uncertainty without a post hoc calibration phase.

Chapter 8: Thesis Discussion and Conclusion.

This chapter completes the thesis and puts its results into the perspective of the research field. We discuss the contributions and point to the limitations of our current approaches, and suggest how they could be addressed in the future.

This arrangement of chapters encapsulates the breadth and depth of the research conducted, offering insights into each of the investigated areas.

2

Uncertainty-Guided Progressive GANs for Medical Image Translation

2.1 Abstract

Image-to-image translation plays a vital role in tackling various medical imaging tasks such as attenuation correction, motion correction, undersampled reconstruction, and denoising. Generative adversarial networks have been shown to achieve the state-of-the-art in generating high fidelity images for these tasks. However, the state-of-the-art GAN-based frameworks do not estimate the uncertainty in the predictions made by the network that is essential for making informed medical decisions and subsequent revision by medical experts and has recently been shown to improve the performance and interpretability of the model. In this work, we propose an uncertainty-guided progressive learning scheme for image-to-image translation. By incorporating aleatoric uncertainty as attention maps for GANs trained in a progressive manner, we generate images of increasing fidelity progressively. We demonstrate the efficacy of our model on three challenging medical image translation tasks, including PET to CT translation, undersampled MRI reconstruction, and MRI motion artefact correction. Our model generalizes well in three different tasks and improves performance over state of the art under full-supervision and weak-supervision with limited data. Code is released here: https://github.com/ExplainableML/UncerGuidedI2I

2.2 Introduction

In the medical domain, each imaging modality reflects particular physical properties of the tissue under examination. This results in images with different dimensionality, spatial resolution, and contrast. Various imaging modalities provide a complimentary stream of information for clinical diagnostics or technical pre and post-processing steps. Moreover, acquiring medical images is susceptible to various kinds of noise and modality-specific artefacts. To remedy these issues, translating images between different domains is of great importance.

Inter-modal image-to-image translation can potentially replace additional acquisition procedures, reducing examination costs and time. Besides, intra-modality image-to-image translation enables complex artefact and noise correction. For example, attenuation correction of positron emission tomography (PET) data is challenging in situations where no density distribution is available from computed tomography (CT) data, as in the case for stand-alone PET scanners or combined PET/magnetic resonance imaging (MRI). In these situations, the generation of pseudo-CTs from PET data can be helpful. Further examples are related to image reconstruction and/or correction in MRI: Reconstruction of undisturbed artifact-free images is hard to achieve with traditional methods; deep-learning-based image-to-image translation can solve this challenge. In particular, generative adversarial networks (GAN) based on convolutional neural networks (CNN) have proven to provide a high visual quality of the generated synthetic images. However, predictions of GANs can be unreliable, and particularly in medical applications, the quantification of uncertainty is of high importance for the interpretation of the results. In this work, we propose a generic end-to-end model that introduces high-capacity conditional progressive GANs to synthesize high-quality images, using aleatoric uncertainty estimates as the guide to focus on improving image quality in regions where the network is highly uncertain about the prediction. We perform experiments on three challenging and vital medical imaging tasks: PET to CT translation, undersampled MRI reconstruction, and motion correction in MRI. Moreover, we empirically demonstrate the efficacy of our model under weak supervision with limited data.

2.2.1 Related Works

Traditional machine learning techniques for medical image translation rely on explicit feature representations [101, 330, 136, 222]. More recently, convolutional neural networks have been proposed for various image translation tasks [158, 235, 56, 93, 115, 32] and state-of-the-art performance is achieved by generative adversarial networks [186, 295, 307, 321, 107, 313, 116, 5, 7, 259, 260]. The existing methods propose conditional GAN architectures with deterministic outputs that typically uses $\mathcal{L}_1/\mathcal{L}_2$-based fidelity loss for the generator assumes a pixel-wise *homoscedasticity* and also assumes the pixel-wise error (i.e., residual) to be *independent and identically distributed* (i.i.d) following a Laplace or Gaussian distribution. This is a limiting assumption as explained in [119, 262, 278]. While these methods can provide synthetic images of high visual quality, the image content may still deviate significantly from the corresponding ground-truth. This results in overconfidence or misinterpretation with negative consequences, particularly in the medical domain. There have been recent works on quantifying aleatoric and epistemic uncertainty in task-specific medical imaging algorithms like classification, segmentation, super-resolution etc [182, 278, 280, 279, 250] quantifying it for general image-to-image translation problem largely remains unexplored. Thus, the central motivation of our work is to provide measures of uncertainty for image-to-image translation tasks that can

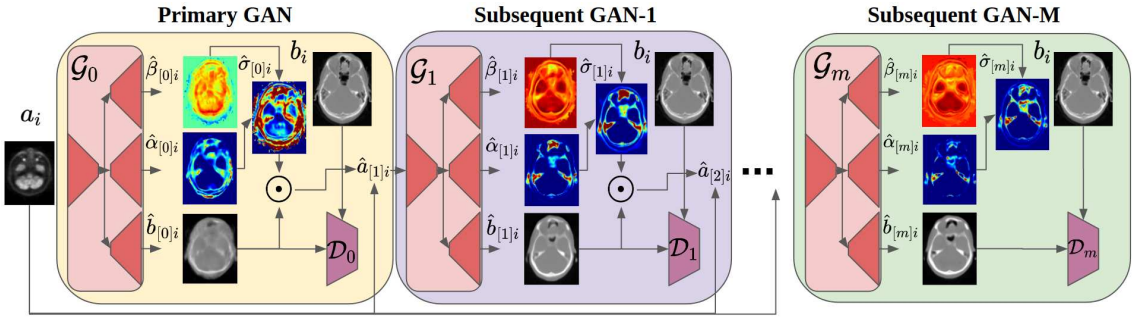

Figure 2.1: Uncertainty-guided Progressive GANs (UP-GAN): The primary GAN takes the input image from domain A, while subsequent GANs absorb outputs from the preceding GAN (see Eq. 2.3 and 2.4). Explicitly guided by the attention maps, the uncertainty maps are estimated from the preceding GAN.

contribute to safe applications of results.

Moreover, recent work has shown that high-capacity generators that are progressive in nature lead to high-quality results as described in [5, 7, 116]. However, the progressive generation of high-quality images remains unguided without specifically attending to poorly translated regions. Prior works indicate a correlation between estimated uncertainty and prediction error [327, 250, 262]. We exploit this relationship for the progressive enhancement of synthetic images, which has not been investigated by prior work before.

2.3 Uncertainty-Guided Progressive GAN (UP-GAN)

Let A and B be two image domains with a set of images $S_A := \{a_1, a_2 ... a_n\}$ and $S_B := \{b_1, b_2 ... b_m\}$ where a_i and b_i represent the i^{th} image from domain A and B respectively. Let each image drawn from an underlying *unknown* probability distribution $\mathcal{P}_{AB}$, i.e., $(a_i, b_i) \sim \mathcal{P}_{AB} \forall i$ have K pixels, and u_{ik} represent the k^{th} pixel of a particular image u_i. Our goal is to learn a mapping from domain A to B ($A \rightarrow B$) in a paired manner, i.e., learning the underlying conditional distribution $\mathcal{P}_{B|A}$ from the set of given samples $\{(a_i, b_i)\}$, following the distribution $\mathcal{P}_{AB}$. For a given image a_i in domain A, the estimated image in domain B is called $\hat{b}_i$. The pixel wise error is defined as $\epsilon_{ij} = \hat{b}_{ij} - b_{ij}$. While the existing framework models the residual as the i.i.d as described above, we relax that assumption by modelling the residual as non i.i.d variables and learning the optimal distribution from the dataset, as described in the following.

Figure 6.2 shows our model that consists of cascaded GANs, where each generator is capable of estimating the aleatoric uncertainty, along with generating images. Our solution alleviates the aforementioned limitations of recent methods by modelling the underlying per-pixel residual distribution as *independent* but *non-identically* distributed *zero-mean generalized Gaussian distribution* (GGD) as in [262], where the network learns to predict the optimal *scale* (α) and *shape* (β) of the GGD for every pixel, Therefore, $\hat{b}_{ij} = b_{ij} + \epsilon_{ij}$ with, $\epsilon_{ij} \sim GGD(\epsilon; 0, \alpha_{ij}, \beta_{ij}) \equiv \beta_{ij}(2\alpha_{ij}\Gamma(\beta_{ij}^{-1}))^{-1} \exp\left(-\alpha_{ij}^{-1}|\epsilon|^{\beta_{ij}}\right)$. We generate images in

multiple phases, with each phase generating output images along with the aleatoric uncertainty estimates. The outputs from one phase serve as the input to the subsequent GAN in the next phase, explicitly guided by the attention map derived from uncertainty estimates. Importantly, this uncertainty-based guidance enforces the model to focus on refining the uncertain regions that are likely to be poorly synthesized, resulting in progressively improving quality.

Our framework is composed of a sequence of M GANs, where the m^{th} GAN is represented by a pair of networks, generator and discriminator, given by, $(\mathcal{G}_m(\cdot;\theta_m), \mathcal{D}_m(\cdot;\phi_m))$. Both the generator and discriminator can have arbitrary network architecture as long as generator can estimate *aleatoric uncertainty* as described in [262]. We choose all the discriminators to be the patch discriminators from [107] and generators to be modified U-Net [217], where the head is split into three to estimate the parameters of the GGD as shown in Figure 6.2 and in [262].

Primary GAN. We train the first GAN ($\mathcal{G}_0$) using the dataset S_A and S_B. The predictions of the generator are given by $(\hat{\alpha}_{[0]i}, \hat{\beta}_{[0]i}, \hat{b}_{[0]i})$. The network is trained with an adaptive fidelity loss function $\mathcal{L}^G_{\alpha\beta}$ [262] and an adversarial loss $\mathcal{L}^G_{\text{adv}}$ [338], combined as $\mathcal{L}^G_{\text{tot}}$ for the generator $(\mathcal{G}_0(\cdot;\theta_0) : A \rightarrow B)$:

$$\mathcal{L}^G_{\alpha\beta}(\hat{b}_{[0]i}, \hat{\alpha}_{[0]i}, \hat{\beta}_{[0]i}, b_i) = \frac{1}{K}\sum_j \left(\frac{|\hat{b}_{[0]ij} - b_{ij}|}{\hat{\alpha}_{[0]ij}}\right)^{\hat{\beta}_{[0]ij}} - \log\frac{\hat{\beta}_{[0]ij}}{\hat{\alpha}_{[0]ij}} + \log\Gamma(\hat{\beta}^{-1}_{[0]ij}) \tag{2.1}$$

$$\mathcal{L}^G_{\text{adv}} = \mathcal{L}_2(\mathcal{D}_1(\hat{b}_{[0]i}), 1) \text{ and } \mathcal{L}^G_{\text{tot}} = \lambda_1\mathcal{L}^G_{\alpha\beta} + \lambda_2\mathcal{L}^G_{\text{adv}}. \tag{2.2}$$

The patch discriminator ($\mathcal{D}_1$) is trained using the adversarial loss from [338] given by $\mathcal{L}^D_{\text{adv}} = \mathcal{L}_2(\mathcal{D}^A(b_i), 1) + \mathcal{L}_2(\mathcal{D}^A(\hat{b}_{[0]i}), 0)$.

Subsequent GANs. The m^{th} GAN (where $m > 0$) takes the output produced by the $(m-1)^{th}$ GAN, i.e. $(\hat{\alpha}_{[m-1]i}, \hat{\beta}_{[m-1]i}, \hat{b}_{[m-1]i})$, along with the original sample a_i from domain A as its input and generates a refined output. The image estimated by the $(m-1)^{th}$ GAN along with its uncertainty map learns to create the input feature $f_{[m]i}$ for the m^{th} GAN, where the uncertainty map serves as an attention mechanism to highlight the uncertain regions in the image. The input $a_{[m]i}$ for the m^{th} generator is given by concatenating a_i and $f_{[m]i}$, i.e.,

$$\hat{\sigma}_{[m-1]i} = \hat{\alpha}_{[m-1]i}\sqrt{\frac{\Gamma(3/\hat{\beta}_{[m-1]i})}{\Gamma(1/\hat{\beta}_{[m-1]i})}}, \text{ and } f_{[m]i} = \hat{b}_{[m-1]i} \odot \frac{\hat{\sigma}_{[m-1]i}}{\sum_j \hat{\sigma}_{[m-1]ij}} \tag{2.3}$$

$$a_{[m]i} = \texttt{concat}(f_{[m]i}, a_i) \tag{2.4}$$

The input $a_{[m]i}$ for the m^{th} GAN encourages the generator to further refine the highly uncertain regions in the image given the original input context. The generator and the discriminator are trained using $\mathcal{L}^G_{\text{tot}}$ and $\mathcal{L}^D_{\text{adv}}$, respectively.

Progressive training scheme. We initialize the parameters $\theta \cup \phi$ sequentially. First, we initialize $\theta_1 \cup \phi_1$ using the training set (S_A, S_B) to minimize the loss function given by $\mathcal{L}^G_{\text{tot}}$ and $\mathcal{L}^D_{\text{adv}}$. Then, for the subsequent GANs, we initialize the $\theta_m \cup \phi_m$ $(m > 1)$ by

fixing the weights of all the previous generators and training the m^{th} GAN alone (see Eq. 2.3 and 2.4 with losses $\mathcal{L}^G_{\text{tot}}$ and $\mathcal{L}^D_{\text{adv}}$). Once all the parameters have been initialized (i.e., $\theta_m \cup \phi_m \forall m$), we do further fine tuning by training all the networks end-to-end by combining the loss functions of all the intermediate phases and a significantly smaller learning-rate.

2.4 Experiments

In this section, we first detail the experimental setup and comparative methods in Section 7.5.1, and present the corresponding results in Section 2.4.2.

2.4.1 Experimental Setup

Tasks and datasets. We evaluate our method on the following three tasks.

(i) PET to CT translation: We synthesize CT images from PET scans to be used for the attenuation correction, e.g. for PET-only scanners or PET/MRI. We use paired data sets of non-attenuation-corrected PET and the corresponding CT of the head region of 49 patients acquired on a state-of-the-art PET/CT scanner (Siemens Biograph mCT), approved by ethics committee of the Medical Faculty of the University of Tübingen. Data is split into 29/5/15 for training/val/test sets. Figure 2.2 shows exemplary slices for co-registered PET and CT.

(ii) Undersampled MRI reconstruction: We translate undersampled MRI images to fully-sampled MRI images. We use MRI scans from the open-sourced IXI [1] dataset that consists of T1-weighted (T1w) MRI scans. We use a cohort of 500 patients split into 200/100/200 for training/val/test, and retrospectively create the undersampled MRI with an acceleration factor of 12.5×, i.e., we preserve only 8% of the fully-sampled k-space measurement (from the central region) to obtain the undersampled image.

(iii) MRI Motion correction: We generate sharp images from motion corrupted images. We retrospectively create the motion artefacts in the T1w MRI from IXI following the transformations in the *k-space* as described in [233]. Figure 2.3-(ii) shows the input MRI scan with artefacts and ground-truth.

Training details and evaluation metrics. All GANs are first initialized using the aforementioned progressive learning scheme with (λ_1, λ_2) in Eq. 2.2 set to $(1, 0.001)$. We use Adam [122], with the hyper-parameters $\beta_1 := 0.9$, $\beta_2 := 0.999$, an initial learning rate of 0.002 for initialization and 0.0005 post-initialization that decays based on cosine annealing over 1000 epochs, using a batch size of 8. We use three widely adopted metrics to evaluate image generation quality: PSNR measures $20 \log \text{MAX}_I / \sqrt{\text{MSE}}$, where MAX_I is the highest possible intensity value in the image and MSE is the mean-squared-error between two images. SSIM computes the structural similarity between two images [289].

[1] from https://brain-development.org/ixi-dataset/

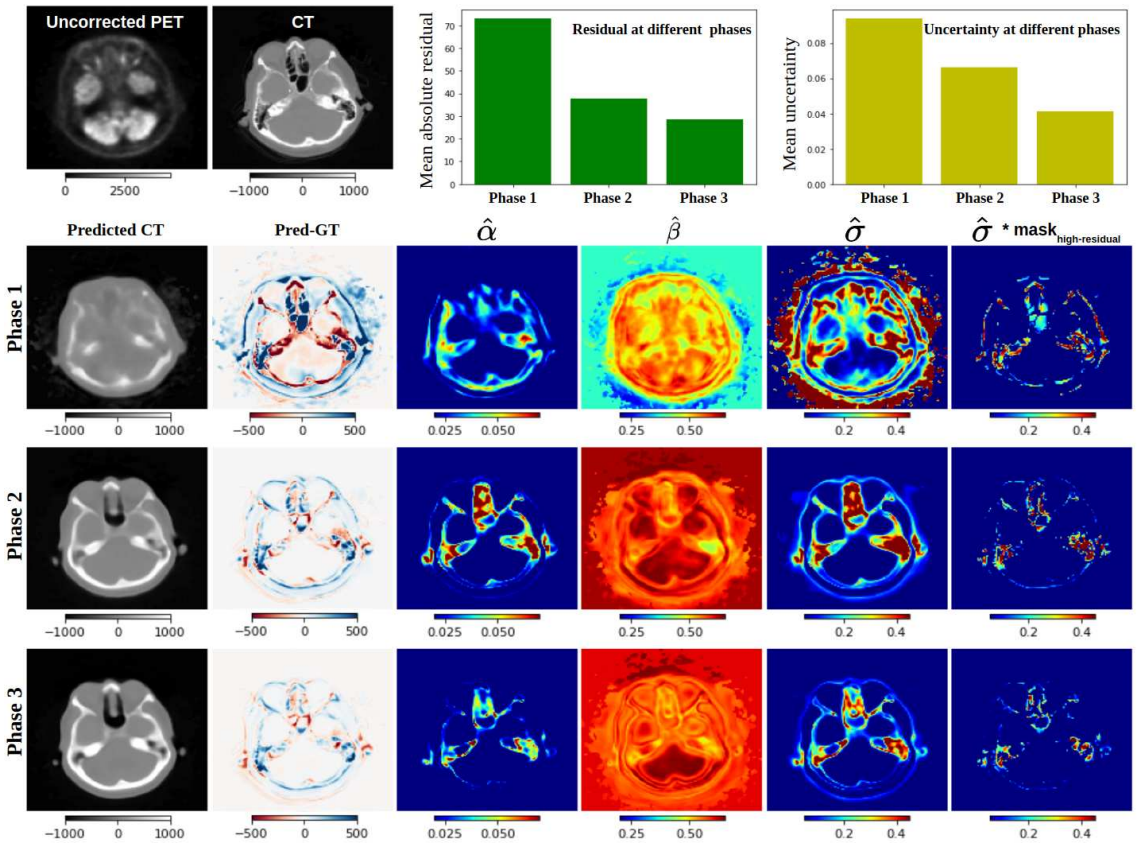

Figure 2.2: Outputs from different phases of UP-GAN (with M=3). (Top) The input (uncorrected PET), the corresponding ground-truth CT, mean residual values over different phases, mean uncertainty values over different phases. (Bottom) Each row shows the predicted output, the residual between the prediction and the ground-truth, the predicted scale (α) map, the predicted shape (β) map, the uncertainty map, and the uncertainty in high residual regions.

MAE computes the mean absolute error between two images. Higher PSNR, SSIM, and lower MAE indicate a higher quality of the generated images (wrt ground-truth).

Compared methods. We compare our model to representative state-of-the-art methods for medical image translation, including Pix2pix [107], a baseline conditional adversarial networks for image-to-image translation tasks using GANs, PAN [277], and MedGAN [5], a GAN-based method that relies on *external-pre-trained feature extractors*, with a generator that refines the generated images progressively. MedGAN is shown to perform superior to methods like, Fila-sGAN [328], ID-cGAN [321], and achieve state-of-the-art performance for several medical image-to-image translation problems.

2.4.2 Results and Analysis

Qualitative results. Figure 2.2 visualizes the (intermediate) outputs of the generators at different phases of the framework. The visual quality of the generated image content increasingly improves along the network phases (as shown in the first column, second

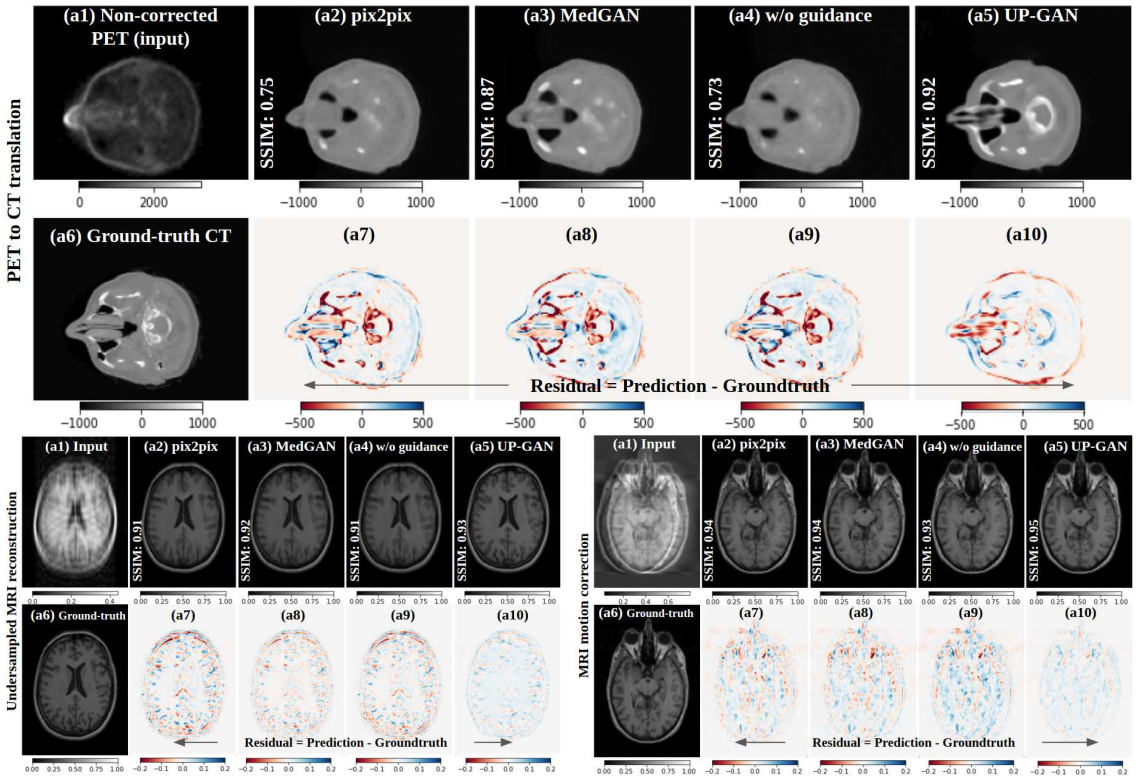

Figure 2.3: Qualitative results. (Top) PET to CT translation. (Bottom) Undersampled MRI reconstruction (left), and MRI motion correction (right). We note that UP-GAN consistently generates higher-quality output that captures much finer details as compared to baselines such as pix2pix, MedGAN, UP-GAN w/o guidance.

Methods	**PET to CT**			**Undersampled MRI Recon.**			**MRI Motion Correction**		
	SSIM	PSNR	MAE	SSIM	PSNR	MAE	SSIM	PSNR	MAE
pix2pix [107]	0.89±0.04	26.0±2.0	38.5±10.7	0.92±0.03	28.5±0.9	27.6±9.3	0.94±0.06	29.6±1.4	26.3±8.2
PAN [277]	0.90±0.08	26.5±4.5	37.2±15.6	0.93±0.05	28.8±0.7	26.2±10.4	0.95±0.10	30.1±2.8	24.9±9.7
MedGAN [5]	0.90±0.04	27.1±2.5	35.4±11.8	0.94±0.02	**29.7±1.9**	24.2±8.7	0.95±0.04	30.8±1.8	23.6±9.1
UP-GAN	**0.95±0.05**	**28.9±0.4**	**24.7±12.9**	**0.97±0.07**	29.4±2.1	**24.1±7.5**	**0.96±0.03**	**32.1±0.3**	**22.8±11.1**

Table 2.1: Evaluation of various methods on three real-world medical image translation tasks: PET to CT translation, undersampled MRI reconstruction, and MRI motion correction. We note that UP-GAN consistently performs better than baselines such as pix2pix, PAN, and MedGAN in terms of metrics like SSIM, PSNE, and MAE.

row onward). At the same time, prediction error and uncertainty decrease continuously (second column and fifth column, second row onward, respectively). High uncertainty values are found in anatomical regions with fine osseous structures, such as the nasal cavity and the inner ear in the petrous portion of the temporal bone. Particularly in such regions of high uncertainty, we achieve a progressive improvement in the level of detail.

Figure 2.3-(Top) visualizes the generated CT images from the PET for all the compared

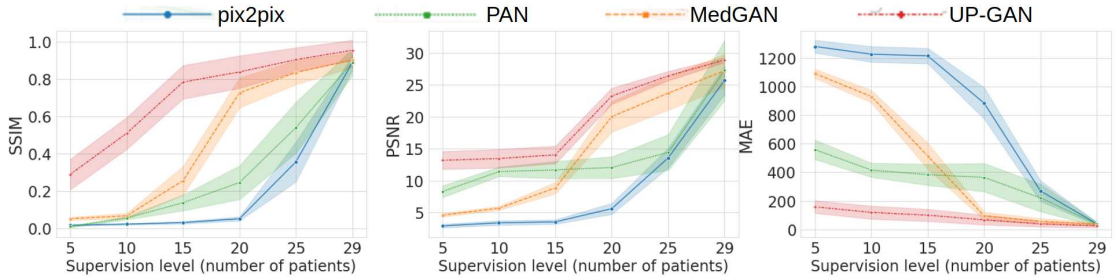

Figure 2.4: Quantitative results in the presence of limited labeled training data. We compare the performance of UP-GAN and baselines like pix2pix, PAN, and MedGAN in terms of metrics like SSIM, PSNR, and MAE with different numbers of training samples. We note that UP-GAN performs better than baselines with limited training samples.

methods along with our methods. We observe that more high-frequency features are present in our prediction compared to the previous state-of-the-art model (MedGAN). We also observe that the overall residual is significantly lower for our method compared to the other baselines. MedGAN performs better than pix2pix in synthesizing high-frequency features and sharper images. Figure 2.3-(Bottom) shows similar results for the undersampled MRI reconstruction task and MRI motion correction task. In both cases, our model yields superior images, as can be seen via relatively neutral residual maps.

Quantitative results. Table 2.1 shows the quantitative performance of all the methods on the three tasks; for all the tasks, our method outperforms the recent models. In particular, for the most challenging task, PET to CT translation, our method with uncertainty-based guide outperforms the previous state-of-the-art method, MedGAN (that relies on task-specific external feature extractor), *without using any external feature extractor*. Therefore, the uncertainty guidance reduces the burden of having an externally trained task-specific feature extractor to achieve high fidelity images. The same trend holds for undersampled MRI reconstruction and motion correction in MRI. The statistical tests on SSIM values of MedGAN and our UP-GAN gives us a *p-value* of 0.016 for PET-to-CT translation, 0.021 for undersampled MRI reconstruction, and 0.036 for MRI motion correction. As all the *p-values* are < 0.05, results are statistically significant.

Ablation study. We study the model that does not utilize the estimated uncertainty maps as attention maps and observe that the model without the uncertainty as the guide performs inferior to the UP-GAN with a performance (SSIM/PSNR/MAE) of (0.87/25.4/40.7), (0.93/27.3/38.7), and (0.92/26.2/35.1) for PET to CT translation, undersampled MRI reconstruction, and MRI motion correction, respectively. UP-GAN model leverages the uncertainty map to refine the predictions where the model is uncertain, which is also correlated to the regions where the translation is poor. The model without uncertainty-based guidance does not focus on the regions mentioned above in the prediction and is unable to perform as well as UP-GAN.

Evaluating models with weak supervision. We evaluate all the models for PET to CT synthesis by limiting the number of paired image samples used for training. We define

five supervision levels corresponding to different amounts of cross-domain pairwise training sample slices. For this experiment, we train the recent state-of-the-art models with a varying number of patients in the training stage, i.e., we use 5, 10, 15, 20, and 29 patients, respectively. Figure 2.4 shows the performance of all the models at varying supervision levels. We observe that our model with uncertainty guidance outperforms all the baselines at full supervision (with 29 patients). Moreover, our model sharply outperforms the baselines with limited training data (with < 29 patients). UP-GAN produces intermediate uncertainty maps that have higher values under weak supervision (compared to the full supervision case), but this still allows UP-GAN to focus on highly uncertain regions, that the current state-of-the-art models do not have access to, hence are not able to leverage that to refine the predicted images.

2.5 Conclusion

In this work, we propose a new generic model for medical image translation using uncertainty-guided progressive GANs. We demonstrate how uncertainty can serve as an attention map in progressive learning schemes. We demonstrate the efficacy of our method on three challenging medical image translation tasks, including PET to CT translation, undersampled MRI reconstruction, and motion correction in MRI. Our method achieves state-of-the-art in various tasks. Moreover, it allows the quantification of uncertainty and shows better generalizability with smaller sample sizes than recent approaches.

3

Robustness via Uncertainty-aware Cycle Consistency

3.1 Abstract

Unpaired image-to-image translation refers to learning inter-image-domain mapping without corresponding image pairs. Existing methods learn deterministic mappings without explicitly modelling the robustness to outliers or predictive uncertainty, leading to performance degradation when encountering unseen perturbations at test time. To address this, we propose a novel probabilistic method based on Uncertainty-aware Generalized Adaptive Cycle Consistency (UGAC), which models the per-pixel residual by generalized Gaussian distribution, capable of modelling heavy-tailed distributions. We compare our model with a wide variety of state-of-the-art methods on various challenging tasks including unpaired image translation of natural images, using standard datasets, spanning autonomous driving, maps, facades, and also in medical imaging domain consisting of MRI. Experimental results demonstrate that our method exhibits stronger robustness towards unseen perturbations in test data. Code is released here: https://github.com/ExplainableML/UncertaintyAwareCycleConsistency.

3.2 Introduction

Translating an image from a distribution, i.e. source domain, to an image in another distribution, i.e. target domain, with a distribution shift is an ill-posed problem as a unique deterministic one-to-one mapping may not exist between the two domains. Furthermore, since the correspondence between inter-domain samples may be missing, their joint-distribution needs to be inferred from a set of marginal distributions. However, as infinitely many joint distributions can be decomposed into a fixed set of marginal distributions [159, 66, 156], the problem is ill-posed in the absence of additional constraints.

Deep learning-based methods tackle the image-to-image translation task by learning inter-domain mappings in a paired or unpaired manner. Paired image translation methods [107, 287, 324, 169, 274, 219] exploit the inter-domain correspondence by penalizing

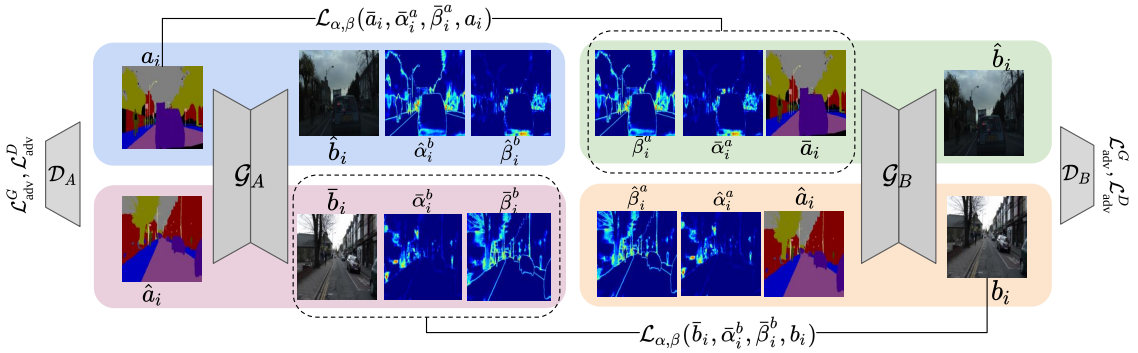

Figure 3.1: Our UGAC framework with the cycle between two generators. For translating from A to B ($A \rightarrow B$), the input a_i is mapped to generalized Gaussian distribution parameterized by $\{\hat{b}_i, \hat{\alpha}_i^b, \hat{\beta}_i^b\}$. The backward cycle ($A \rightarrow B \rightarrow A$) reconstructs the image distribution parameterized by $\{\bar{a}_i, \bar{\alpha}_i^a, \bar{\beta}_i^a\}$. UGAC uses $\mathcal{L}_{\alpha\beta}$ objective function in Eq. 3.8 and adversarial losses in Eq. 3.10 and 3.11.

the per-pixel residual (using l_1 or l_2 norm) between the output and corresponding ground-truth sample. Unpaired image translation approaches [159, 338, 195, 257, 329, 110] often use adversarial networks with an additional constraint on the image or feature space imposing structure on the underlying joint distribution of the images from the different domains.

Both paired and unpaired image translation approaches often learn a deterministic mapping between the domains where every pixel in the input domain is mapped to a fixed pixel value in the output domain. However, such a deterministic formulation can lead to mode collapse while at the same time not being able to quantify the model predictive uncertainty important for critical applications, e.g., medical image analysis. It is desirable to test the performance of the model on unseen perturbed input at test-time, to improve their applicability in the real world. While robustness to outliers is a focus in some domains [98, 77, 92, 23], it has not attracted as much attention in unpaired translation.

To address these limitations, we propose an unpaired (unsupervised) probabilistic image-to-image translation method trained without inter-domain correspondence in an end-to-end manner. The probabilistic nature of this method provides uncertainty estimates for the predictions. Moreover, modelling the residuals between the predictions and the ground-truth with heavy-tailed distributions makes our model robust to outliers and various unseen data. Accordingly, we compare various state-of-the-art models and our model in their capacity to handle samples from similar distribution as training-dataset as well as perturbed samples, in the context of unpaired translation.

Our contributions are as follows. (i) We propose an *unpaired* probabilistic image-to-image translation framework based on Uncertainty-aware Generalized Adaptive Cycle Consistency (UGAC). Our framework models the residuals between the predictions and the ground-truths with heavy-tailed distributions improving robustness to outliers. Probabilistic nature of UGAC also provides uncertainty estimates for the predictions. (ii) We evaluate UGAC on multiple challenging datasets: natural images consisting

Cityscapes [42], Google aerial maps and photos [107], CMP Facade [258] and medical images consisting of MRI from IXI [213]. We compare our model to seven state-of-the-art image-to-image translation methods [257, 68, 159, 195, 338, 13]. Our results demonstrate that while UGAC performs competitively when tested on unperturbed images, it improves state-of-the-art methods substantially when tested on unseen perturbations, establishing its robustness. (iii) We show that our estimated uncertainty scores correlate with the model predictive errors (i.e., residual between model prediction and the ground-truth) suggesting that it acts as a good proxy for the model's reliability at test time.

3.3 Related Work

Image-to-image translation. Image-to-image translation is often formulated as per-pixel deterministic regression between two image domains of [300, 161, 104]. In [107], this is done in a *paired* manner using conditional adversarial networks, while in [338, 159, 257, 68, 195] this is done in an *unpaired* manner by enforcing additional constraints on the joint distribution of the images from separate domains. Both CycleGAN [338] and UNIT [159] learn bi-directional mappings, whereas other recent methods [257, 68, 195] learn uni-directional mappings.

Quantification of uncertainty in the predictions made by the unpaired image-to-image translation models largely remains unexplored. Our proposed method operates at the intersection of uncertainty estimation and unsupervised translation. Critical applications such as medical image-to-image translation [270, 306, 49, 6, 260, 259] is an excellent testbed for our model as confidence in the network's predictions is desirable [172, 228] especially under the influence of missing imaging modalities.

Uncertainty estimation. Among two broad categories of uncertainties that can be associated with a model's prediction, *epistemic* uncertainty in the model parameters is learned with finite data whereas *aleatoric* uncertainty captures the noise/uncertainty inherent in the data [114, 119]. For image-to-image translation, various uncertainties can be estimated using Bayesian deep learning techniques [119, 127, 70, 139, 87]. In critical areas like medical imaging, the errors in the predictions deter the adoption of such frameworks in clinical contexts. Uncertainty estimates for the predictions would allow subsequent revision by clinicians [335, 97, 14, 311, 182, 278, 113, 263, 246, 250, 251].

Existing methods model the per-pixel *heteroscedasticity* as Gaussian distribution for regression tasks [119]. This is not optimal in the presence of outliers that often tend to follow heavy-tailed distributions [189, 26]. Therefore, we enhance the above setup by modelling per-pixel heteroscedasticity as generalized Gaussian distribution, which can model a wide variety of distributions, including Gaussian, Laplace, and heavier-tailed distribution.

3.4 Uncertainty-aware Generalized Adaptive Cycle-consistency (UGAC)

We present the formulation of the unpaired image-to-image translation problem. We discuss the shortcomings of the existing solution involving the cycle consistency loss called CycleGAN [338]. Finally, we present our novel probabilistic framework (UGAC) that overcomes the described shortcomings.

3.4.1 Preliminaries

Formulation. Let there be two image domains A and B. Let the set of images from domain A and B be defined by (i) $S_A := \{a_1, a_2...a_n\}$, where $a_i \sim \mathcal{P}_A \; \forall i$ and (ii) $S_B := \{b_1, b_2...b_m\}$, where $b_i \sim \mathcal{P}_B \; \forall i$, respectively. The elements a_i and b_i represent the i^{th} image from domain A and B respectively, and are drawn from an underlying *unknown* probability distribution $\mathcal{P}_A$ and $\mathcal{P}_B$ respectively.

Let each image have K pixels, and u_{ik} represent the k^{th} pixel of a particular image u_i. We are interested in learning a mapping from domain A to B ($A \rightarrow B$) and B to A ($B \rightarrow A$) in an unpaired manner so that the correspondence between the samples from $\mathcal{P}_A$ and $\mathcal{P}_B$ is not required at the learning stage. In other words, we want to learn the underlying joint distribution $\mathcal{P}_{AB}$ from the given marginal distributions $\mathcal{P}_A$ and $\mathcal{P}_B$. This work utilizes CycleGANs that leverage the cycle consistency to learn mappings from both directions ($A \rightarrow B$ and $B \rightarrow A$), but often we are only interested in one direction and the second direction is the auxiliary mapping that aids in learning process. We define the mapping $A \rightarrow B$ as primary and $B \rightarrow A$ as auxiliary.

Cycle consistency. Learning a joint distribution from the marginal distributions is an ill-posed problem with infinitely many solutions [156]. CycleGAN [338] enforces an additional structure on the joint distribution using a set of primary networks (forming a GAN) and a set of auxiliary networks. The primary networks are represented by $\{\mathcal{G}_A(\cdot;\theta_A^{\mathcal{G}}), \mathcal{D}_A(\cdot;\theta_A^{\mathcal{D}})\}$, where $\mathcal{G}_A$ represents a generator and $\mathcal{D}_A$ represents a discriminator. The auxiliary networks are represented by $\{\mathcal{G}_B(\cdot;\theta_B^{\mathcal{G}}), \mathcal{D}_B(\cdot;\theta_B^{\mathcal{D}})\}$. While the primary networks learn the mapping $A \rightarrow B$, the auxiliary networks learn $B \rightarrow A$ (see Figure 3.1). Let the output of the generator $\mathcal{G}_A$ translating samples from domain A (say a_i) to domain B be called $\hat{b}_i$. Similarly, for the generator $\mathcal{G}_B$ translating samples from domain B (say b_i) to domain A be called $\hat{a}_i$, i.e., $\hat{b}_i = \mathcal{G}_A(a_i;\theta_A^{\mathcal{G}})$ and $\hat{a}_i = \mathcal{G}_B(b_i;\theta_B^{\mathcal{G}})$. To simplify the notation, we will omit writing parameters of the networks in the equation. The cycle consistency constraint [338] re-translates the above predictions $(\hat{b}_i, \hat{a}_i)$ to get back the reconstruction in the original domain $(\bar{a}_i,\bar{b}_i)$, where, $\bar{a}_i = \mathcal{G}_B(\hat{b}_i)$ and $\bar{b}_i = \mathcal{G}_A(\hat{a}_i)$, and attempts to make reconstructed images $(\bar{a}_i, \bar{b}_i)$ similar to original input (a_i, b_i) by penalizing the residuals with $\mathcal{L}_1$ norm between the reconstructions and the original input images, giving the cycle consistency $\mathcal{L}_{\text{cyc}}(\bar{a}_i, \bar{b}_i, a_i, b_i) = \mathcal{L}_1(\bar{a}_i, a_i) + \mathcal{L}_1(\bar{b}_i, b_i)$.

Limitations of cycle consistency. The underlying assumption when penalizing with

the $\mathcal{L}_1$ norm is that the residual at *every pixel* between the reconstruction and the input follow *zero-mean and fixed-variance Laplace* distribution, i.e., $\bar{a}_{ij} = a_{ij} + \epsilon^a_{ij}$ and $\bar{b}_{ij} = b_{ij} + \epsilon^b_{ij}$ with,

$$\epsilon^a_{ij}, \epsilon^b_{ij} \sim Laplace(\epsilon; 0, \frac{\sigma}{\sqrt{2}}) \equiv \frac{1}{\sqrt{2\sigma^2}} e^{-\sqrt{2}\frac{|\epsilon-0|}{\sigma}}, \tag{3.1}$$

where σ^2 represents the fixed-variance of the distribution, a_{ij} represents the j^{th} pixel in image a_i, and ϵ^a_{ij} represents the noise in the j^{th} pixel for the estimated image $\bar{a}_{ij}$. This assumption on the residuals between the reconstruction and the input enforces the likelihood (i.e., $\mathscr{L}(\Theta|\mathcal{X}) = \mathcal{P}(\mathcal{X}|\Theta)$, where $\Theta := \theta^{\mathcal{G}}_A \cup \theta^{\mathcal{G}}_B \cup \theta^{\mathcal{D}}_A \cup \theta^{\mathcal{D}}_B$ and $\mathcal{X} := S_A \cup S_B$) to follow a *factored Laplace* distribution:

$$\mathscr{L}(\Theta|\mathcal{X}) \propto \prod_{ijpq} e^{-\frac{\sqrt{2}|\bar{a}_{ij}-a_{ij}|}{\sigma}} e^{-\frac{\sqrt{2}|\bar{b}_{pq}-b_{pq}|}{\sigma}}, \tag{3.2}$$

where minimizing the negative-log-likelihood yields $\mathcal{L}_{\text{cyc}}$ with the following limitations. The residuals in the presence of outliers may not follow the Laplace distribution but instead a heavy-tailed distribution, whereas the i.i.d assumption leads to fixed variance distributions for the residuals that do not allow modelling of *heteroscedasticity* to aid in uncertainty estimation.

3.4.2 Building Uncertainty-aware Cycle Consistency

We propose to alleviate the mentioned issues by modelling the underlying per-pixel residual distribution as independent but *non-identically* distributed *zero-mean generalized Gaussian distribution* (GGD) (Figure 3.2), i.e., with no fixed shape ($\beta > 0$) and scale ($\alpha > 0$) parameters. Instead, all the shape and scale parameters of the distributions are predicted from the networks and formulated as follows:

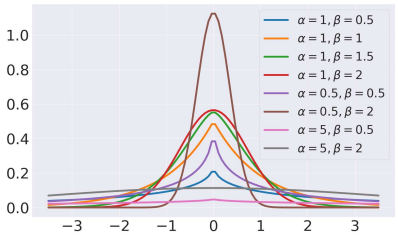

Figure 3.2: Probability density function (pdf) for generalized Gaussian distribution. Different scale (α) and shape (β) parameters lead to different tail behaviour. $(\alpha, \beta) = (1, 2)$ represents Gaussian distribution.

$$\epsilon^a_{ij}, \epsilon^b_{ij} \sim GGD(\epsilon; 0, \bar{\alpha}_{ij}, \bar{\beta}_{ij}) \equiv \frac{\bar{\beta}_{ij}}{2\bar{\alpha}_{ij}\Gamma(\frac{1}{\bar{\beta}_{ij}})} e^{-\left(\frac{|\epsilon-0|}{\bar{\alpha}_{ij}}\right)^{\bar{\beta}_{ij}}}. \tag{3.3}$$

For each ϵ_{ij}, the parameters of the distribution $\{\bar{\alpha}_{ij}, \bar{\beta}_{ij}\}$ may not be the same as parameters for other ϵ_{ik}s; therefore, they are non-identically distributed allowing modelling with heavier tail distributions. The likelihood for our proposed model is,

$$\mathscr{L}(\Theta|\mathcal{X}) = \prod_{ijpq} \mathscr{G}(\bar{\beta}^a_{ij}, \bar{\alpha}^a_{ij}, \bar{a}_{ij}, a_{ij}) \mathscr{G}(\bar{\beta}^b_{pq}, \bar{\alpha}^b_{pq}, \bar{b}_{pq}, b_{pq}), \tag{3.4}$$

where $(\bar{\beta}^a_{ij})$ represents the j^{th} pixel of domain A's shape parameter β^a_i (similarly for others). $\mathscr{G}(\bar{\beta}^u_{ij}, \bar{\alpha}^u_{ij}, \bar{u}_{ij}, u_{ij})$ is the pixel-likelihood at j^{th} pixel of image u_i (that can represent images of both domain A and B) formulated as,

$$\mathscr{G}(\bar{\beta}^u_{ij}, \bar{\alpha}^u_{ij}, \bar{u}_{ij}, u_{ij}) = GGD(u_{ij}; \bar{u}_{ij}, \bar{\alpha}^u_{ij}, \bar{\beta}^u_{ij}), \tag{3.5}$$

The negative-log-likelihood is given by,

$$-\ln \mathscr{L}(\Theta|\mathcal{X}) = -\sum_{ijpq}\left[\ln \frac{\bar{\beta}^a_{ij}}{2\bar{\alpha}^a_{ij}\Gamma(\frac{1}{\bar{\beta}^a_{ij}})} e^{-\left(\frac{|\bar{a}_{ij}-a_{ij}|}{\bar{\alpha}^a_{ij}}\right)^{\bar{\beta}^a_{ij}}} + \ln \frac{\bar{\beta}^b_{pq}}{2\bar{\alpha}^b_{pq}\Gamma(\frac{1}{\bar{\beta}^b_{pq}})} e^{-\left(\frac{|\bar{b}_{pq}-b_{pq}|}{\bar{\alpha}^b_{pq}}\right)^{\bar{\beta}^b_{pq}}}\right] \tag{3.6}$$

minimizing the negative-log-likelihood yields a new cycle consistency loss, which we call as the uncertainty-aware generalized adaptive cycle consistency loss $\mathcal{L}_{\text{ucyc}}$, given $\mathscr{A} = \{\bar{a}_i, \bar{\alpha}^a_i, \bar{\beta}^a_i, a_i\}$ and $\mathscr{B} = \{\bar{b}_i, \bar{\alpha}^b_i, \bar{\beta}^b_i, b_i\}$,

$$\mathcal{L}_{\text{ucyc}}(\mathscr{A}, \mathscr{B}) = \mathcal{L}_{\alpha\beta}(\mathscr{A}) + \mathcal{L}_{\alpha\beta}(\mathscr{B}), \tag{3.7}$$

where $\mathcal{L}_{\alpha\beta}(\mathscr{A}) = \mathcal{L}_{\alpha\beta}(\bar{a}_i, \bar{\alpha}^a_i, \bar{\beta}^a_i, a_i)$ is the new objective function corresponding to domain A,

$$\mathcal{L}_{\alpha\beta}(\bar{a}_i, \bar{\alpha}^a_i, \bar{\beta}^a_i, a_i) = \frac{1}{K}\sum_j \left(\frac{|\bar{a}_{ij} - a_{ij}|}{\bar{\alpha}^a_{ij}}\right)^{\bar{\beta}^a_{ij}} - \log\frac{\bar{\beta}^a_{ij}}{\bar{\alpha}^a_{ij}} + \log\Gamma(\frac{1}{\bar{\beta}^a_{ij}}), \tag{3.8}$$

where $(\bar{a}_i, \bar{b}_i)$ are the reconstructions for (a_i, b_i) and $(\bar{\alpha}^a_i, \bar{\beta}^a_i)$, $(\bar{\alpha}^b_i, \bar{\beta}^b_i)$ are scale and shape parameters for the reconstruction $(\bar{a}_i, \bar{b}_i)$, respectively.

The $\mathcal{L}_1$ norm-based cycle consistency ($\mathcal{L}_{\text{cyc}}$) is a special case of $\mathcal{L}_{\text{ucyc}}$ with $(\bar{\alpha}^a_{ij}, \bar{\beta}^a_{ij}, \bar{\alpha}^b_{ij}, \bar{\beta}^b_{ij}) = (1, 1, 1, 1) \forall i, j$. To utilize $\mathcal{L}_{\text{ucyc}}$, one must have the α maps and the β maps for the reconstructions of the inputs. To obtain the reconstructed image, α (scale map), and β (shape map), we modify the head of the generators (the last few convolutional layers) and split them into three heads, connected to a common backbone. Therefore, for inputs a_i and b_i to the generator $\mathcal{G}_A$ and $\mathcal{G}_B$, the outputs are:

$$\begin{aligned}(\hat{b}_i, \hat{\alpha}^b_i, \hat{\beta}^b_i) = \mathcal{G}_A(a_i) \text{ and } (\bar{a}_i, \bar{\alpha}^a_i, \bar{\beta}^a_i) = \mathcal{G}_B(\hat{b}_i)\\ (\hat{a}_i, \hat{\alpha}^a_i, \hat{\beta}^a_i) = \mathcal{G}_B(b_i) \text{ and } (\bar{b}_i, \bar{\alpha}^b_i, \bar{\beta}^b_i) = \mathcal{G}_A(\hat{a}_i),\end{aligned} \tag{3.9}$$

The estimates are plugged into Eq. (3.7) and the networks are trained to estimate all the parameters of the GGD modelling domain A and B, i.e. $(\bar{a}_{ij}, \bar{\alpha}^a_{ij}, \bar{\beta}^a_{ij})$ and $(\bar{b}_{ij}, \bar{\alpha}^b_{ij}, \bar{\beta}^b_{ij})$ $\forall ij$.

Furthermore, we apply adversarial losses [338] to the mapping functions, (i) $\mathcal{G}_A : A \rightarrow B$ and (ii) $\mathcal{G}_B : B \rightarrow A$, using the discriminators $\mathcal{D}_A$ and $\mathcal{D}_B$. The discriminators are inspired from patchGANs [107, 338] that classify whether 70x70 overlapping patches are real or not. The adversarial loss for the generators ($\mathcal{L}^G_{\text{adv}}$ [338]) is,

$$\mathcal{L}^G_{\text{adv}} = \mathcal{L}_2(\mathcal{D}^A(\hat{b}_i), 1) + \mathcal{L}_2(\mathcal{D}^B(\hat{a}_i), 1). \tag{3.10}$$

The loss for discriminators ($\mathcal{L}^D_{\text{adv}}$ [338]) is,

$$\mathcal{L}^D_{\text{adv}} = \mathcal{L}_2(\mathcal{D}^A(b_i), 1) + \mathcal{L}_2(\mathcal{D}^A(\hat{b}_i), 0) + \mathcal{L}_2(\mathcal{D}^B(a_i), 1) + \mathcal{L}_2(\mathcal{D}^B(\hat{a}_i), 0). \tag{3.11}$$

To train the networks we update the generator and discriminator sequentially at every step [338, 107, 83]. The generators and discriminators are trained to minimize $\mathcal{L}^G$ and $\mathcal{L}^D$ as follows:

$$\mathcal{L}^G = \lambda_1 \mathcal{L}_{\text{ucyc}} + \lambda_2 \mathcal{L}^G_{\text{adv}} \text{ and } \mathcal{L}^D = \mathcal{L}^D_{\text{adv}}. \tag{3.12}$$

Closed-form solution for aleatoric uncertainty. Although predicting parameters of the output image distribution allows to sample multiple images for the same input and compute the uncertainty, modelling the distribution as GGD gives us the uncertainty ($\sigma_{\text{aleatoric}}$) without sampling from the distribution as a closed form solution exists, $\sigma^2_{\text{aleatoric}} = \frac{\alpha^2\Gamma(\frac{3}{\beta})}{\Gamma(\frac{1}{\beta})}$. Epistemic uncertainty ($\sigma_{\text{epistemic}}$) is calculated by multiple forward passes ($T = 50$ times) with dropouts activated for the same input and computing the variance across the outputs ($\hat{u}_t$), i.e., $\sigma^2_{\text{epistemic}} = (\sum_t (\hat{u}_t - \sum_t \frac{\hat{u}_t}{T})^2)/T$. We define the total uncertainty (σ) as $\sigma^2 = \sigma^2_{\text{aleatoric}} + \sigma^2_{\text{epistemic}}$.

3.5 Experiments

In this section, we first describe our experimental setup and implementation details.We compare our model to a wide variety of state-of-the-art methods quantitatively and qualitatively. Finally we provide an ablation analysis to study the rationale of our model formulation.

3.5.1 Experimental Setup

Tasks. We study the robustness of unpaired image-to-image translation methods, where different methods are first trained on *clean* images and then evaluated on *perturbed* images The *clean* images are referred as noise-level 0 (NL0); while the *perturbed* images with *increasing* noise are referred as NL1, NL2, and NL3. We test three types of perturbation including Gaussian, Uniform, and Impulse. From NL0 to NL3, the standard deviation of the additive Gaussian noise is gradually increased. Similarly, for additive uniform noise, different levels are obtained by gradually increase the upper-bound of the uniform sampling interval [210] and for impulse noise we gradually increase the probability of pixel-value replacement [137].

Datasets. We evaluate on four standard datasets used for image-to-image translation: (i) *Cityscapes* [42] contains street scene images with segmentation maps, including 2,975 training and 500 validation and test images; (ii) *Google maps* [107] contains 1,096 training and test images scraped from Google maps with aerial photographs and maps; (iii) *CMP Facade* [258] contains 400 images from the CMP Facade Database including

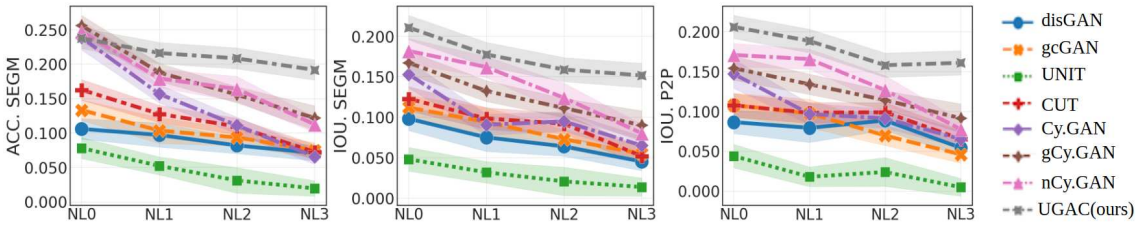

Figure 3.3: Evaluation of different methods on Cityscapes with Gaussian perturbation under varying noise levels. NL0 denotes clean images without noise, NL1, NL2, NL3 are unseen noise levels. `ACC.Segm`, `IoU.Segm`, `IoU.P2P` are three metrics for evaluating translation quality. Higher is better.

architectural facades labels and photos. (iv) *IXI* [213] is a medical imaging dataset with 15,000/5,000/10,000 training/test/validation images, including T1 MRI and T2 MRI.

Translation quality metrics. Following [13], we evaluate the translation quality of the generated segmentation maps and images, for the datasets with segmentation maps (e.g., Cityscapes). First, to evaluate the generated segmentation maps, we compute the Intersection over union (`IoU.SEGM`) and mean class-wise accuracy (`Acc.SEGM`) between the generated segmentation maps and the ground-truth segmentation maps. Second, to evaluate the generated images, we first feed the generated images X_{tr} to a pre-trained pix2pix model [107] (denoted as $p2p$, which is trained to translate images to segmentation maps) to obtain the segmentation maps $p2p(X_{tr})$. Then, we feed the original images X_{org} to the same pix2pix model to obtain another segmentation maps $p2p(X_{org})$, and compute the IoU between two outputs $p2p(X_{tr})$ and $p2p(X_{org})$ (`IoU.P2P`).

Metrics for model robustness. We define two metrics similar to [13] to test model robustness towards noisy inputs. (i) `AMSE` is the area under the curve measuring the MSE between the outputs of the noisy input and the clean input under different levels of noise, i.e., $\texttt{AMSE} = \int_{\eta_{\min}}^{\eta_{\max}} (\texttt{MSE}(\mathcal{G}_A(a_i + \eta), \mathcal{G}_A(a_i)))d\eta$, where η is the noise level, $\mathcal{G}_A$ denotes the generator that maps domain sample a_i (from domain A) to domain B. (ii) `ASSIM` is the area under the curve measuring the SSIM [289] between the outputs of the noisy input and the clean input under different levels of noise, i.e., $\texttt{ASSIM} = \int_{\eta_{\min}}^{\eta_{\max}} (\texttt{SSIM}(\mathcal{G}_A(a_i + \eta), \mathcal{G}_A(a_i))d\eta$. These two metrics show how much the output deviates when fed with the corrupted input from the output corresponding to clean input, averaged over multiple corruption/noise levels.

Implementation details. In our framework, the generator is a cascaded U-Net that progressively improves the intermediate features to yield high-quality output [6], we use a patch discriminator [107]. All the networks were trained using Adam optimizer [123] with a mini-batch size of 2. The initial learning rate was set to 2×10^{-4} and cosine annealing was used to decay the learning rate over 1000 epochs. The hyper-parameters, (λ_1, λ_2) (Eq. 3.12) were set to $(10, 2)$. For numerical stability, the proposed network produces $\frac{1}{\alpha}$ instead of α. The positivity constraint on the output (for predicted α, β) is enforced by applying the ReLU at the end of the output layers in the network.

P	Methods	Cityscapes		Maps		Facade		IXI	
		AMSE (std)↓	ASSIM (std)↑	AMSE (std)↓	ASSIM (std)↑	AMSE (std)↓	ASSIM (std)↑	AMSE (std)↓	ASSIM (std)↑
	gcGAN [68]	107.83 (10.8)	0.62 (0.09)	117.21 (10.6)	0.43 (0.07)	138.21 (11.5)	0.41 (0.05)	108.32 (8.7)	0.67 (0.12)
	CUT [195]	108.34 (8.7)	0.51 (0.12)	119.32 (8.9)	0.51 (0.11)	123.22 (17.6)	0.58 (0.09)	87.12 (10.4)	0.64 (0.07)
	Cy.GAN [338]	121.32 (10.3)	0.31 (0.13)	107.32 (7.5)	0.61 (0.13)	134.23 (15.3)	0.45 (0.07)	98.14 (9.1)	0.70 (0.09)
	nCy.GAN [13]	107.76 (11.2)	0.60 (0.08)	96.14 (9.3)	0.68 (0.05)	109.32 (10.4)	0.68 (0.06)	88.36 (8.2)	0.77 (0.09)
	UGAC (ours)	80.19 (10.4)	0.78 (0.09)	72.32 (8.4)	0.82 (0.07)	95.37 (9.3)	0.77 (0.04)	68.38 (9.8)	0.87 (0.11)
	gcGAN [68]	96.76 (18.2)	0.66 (0.03)	104.83 (11.7)	0.49 (0.09)	129.54 (15.1)	0.47 (0.09)	91.45 (13.3)	0.71 (0.08)
	CUT [195]	98.45 (9.8)	0.59 (0.09)	108.21 (7.5)	0.53 (0.14)	114.45 (21.9)	0.55 (0.12)	75.31 (8.3)	0.78 (0.15)
	Cy.GAN [338]	111.17 (15.4)	0.35 (0.08)	91.47 (10.8)	0.70 (0.10)	158.57 (25.2)	0.39 (0.16)	85.24 (9.5)	0.72 (0.05)
	nCy.GAN [13]	97.89 (12.1)	0.64 (0.04)	75.97 (10.7)	0.78 (0.16)	106.79 (18.7)	0.69 (0.14)	70.89 (8.8)	0.81 (0.09)
	UGAC (ours)	63.77 (8.5)	0.83 (0.07)	51.24 (6.6)	0.88 (0.11)	92.77 (13.2)	0.78 (0.07)	43.54 (6.2)	0.89 (0.05)
	gcGAN [68]	105.64 (17.3)	0.60 (0.07)	116.55 (15.8)	0.45 (0.11)	134.56 (10.7)	0.40 (0.11)	121.31 (17.4)	0.66 (0.13)
	CUT [195]	90.56 (11.6)	0.52 (0.11)	97.21 (7.8)	0.65 (0.09)	118.89 (15.9)	0.52 (0.07)	98.66 (9.7)	0.69 (0.09)
	Cy.GAN [338]	122.48 (19.6)	0.30 (0.12)	112.38 (9.8)	0.62 (0.13)	174.65 (19.2)	0.33 (0.14)	106.16 (14.8)	0.67 (0.12)
	nCy.GAN [13]	95.78 (10.6)	0.61 (0.05)	90.17 (13.2)	0.77 (0.08)	119.89 (12.8)	0.57 (0.09)	96.91 (10.57)	0.73 (0.06)
	UGAC (ours)	78.85 (6.9)	0.80 (0.10)	66.58 (10.4)	0.86 (0.05)	103.83 (9.4)	0.72 (0.09)	70.54 (10.4)	0.85 (0.07)

Table 3.1: Evaluating methods on four datasets under Gaussian, Uniform and Impulse perturbations, evaluated with `AMSE` (lower better) and `ASSIM` (higher better) across varying noise levels. "P" = perturbation. We show results with best performing four methods.

3.5.2 Comparing with the State of the Art

Compared methods. We compare our model to seven state-of-the-art methods for unpaired image-to-image translation, including (1) distanceGAN [17] (disGAN): a uni-directional method to map different domains by maintaining a distance metric between samples of the domains. (2) geometry consistent GAN [68] (gcGAN): a uni-directional method that imposes pairwise distance and geometric constraints. (3) UNIT [159]: a bi-directional method that matches the latent representations of the two domain. (4) CUT [195]: a uni-directional method that uses contrastive learning to match the patches in the same locations in both domains. (5) CycleGAN [338] (Cy.GAN): a bi-directional method that uses cycle consistency loss. (6) guess CycleGAN [13]: a variant of CycleGAN that uses an additional guess discriminator that "guesses" at random which of the image is fake in the collection of input and reconstruction images. (7) adversarial noise CycleGAN [13] (nCy.GAN): another variant of CycleGAN that introduces noise in the cycle consistency loss. Note that both guess CycleGAN [13] and adversarial noise CycleGAN [13] improve the model robustness to noise.

Quantitative evaluation. As described in Section 7.5.1, we trained the models using the *clean* images (NL0) and evaluated them at varying noise levels (NL0, NL1, NL2, NL3), results are detailed next.

Figure 3.3 shows the quantitative results on *Cityscapes* dataset with Gaussian perturbation. When increasing the noise levels, we observe that the performance of compared methods degrade significantly, while our method remains more robust to noise – e.g., the mean `IoU.SEGM` values are changed from around 0.24 to 0.2 for our model but degrades from around 0.24 to 0.05 for the baseline Cy.GAN. Similarly, our model outperforms two strong competitors (gCy.GAN, nCy.GAN) that are built to defend noise perturbation on

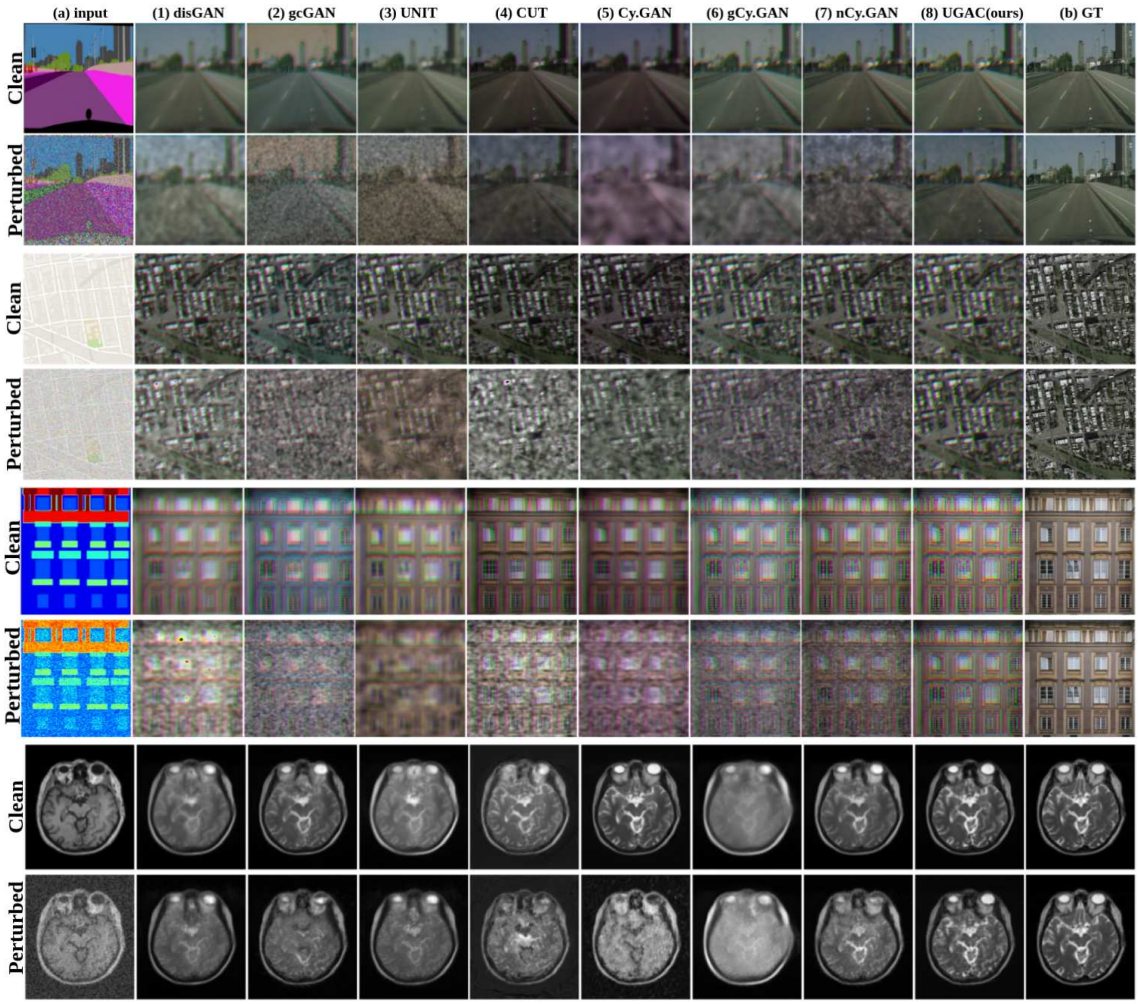

Figure 3.4: Qualitative results on Cityscapes, Google Maps, CMP Facade, and IXI. Outputs of clean image (at NL0) and perturbed image (at NL3) are shown. **(a)** input, **(1)–(7)** outputs from compared methods, and **(8)** output from UGAC, **(b)** ground-truth images. Outputs of UGAC are much closer to groundtruth images (better in quality) than the other methods in the presence of noise perturbations.

higher noise levels. Similar trends are observed for other datasets. This indicates that our model offers better translation quality at higher noise levels.

To evaluate model robustness, we tested different methods using the metrics `AMSE` and `ASSIM` to quantify the overall image quality under increasing noise levels as defined in Section 7.5.1. Table 7.1 shows the performance of all the models on different datasets for three types of perturbations, i.e., Gaussian, Uniform, and Impulse. We can see that the proposed UGAC model performs better than other methods. For instance, when adding Gaussian noise, UGAC obtains a much better `ASSIM` of 0.78/0.82/0.77/0.87 vs. 0.60/0.68/0.68/0.77 yielded by the best competitor nCy.GAN on Cityscapes/Maps/Facade/IXI. When adding Uniform noise or Impulse noise, we can also find that our model outperforms the other methods by substantial margins. Overall, the better performance of UGAC on different datasets suggests its stronger robustness towards various types of perturbations.

Qualitative results. Figure 3.4 visualizes the generated output images for Cityscapes, Google Maps, CMP Facade, and IXI datasets where all the models are trained with clean images and tested with either clean images or perturbed images. The test-time perturbation is of type Gaussian and corresponds to noise-level NL2. We see that, while all the methods generate samples of high quality when tested on unperturbed clean input; whereas when tested with perturbed inputs, we observe results with artifacts but the artifacts are imperceptible in our UCAC method.

The results on Cityscapes dataset (with the primary direction, translating from segmentation maps to real photo) demonstrate that with perturbed input, methods such as disGAN, gcGAN, UNIT generate images with high frequency artifacts (col.1 to 3), whereas methods such as CUT, Cy.GAN, gCy.GAN and nCy.GAN (col.4 to 7) generate images with low frequency artefacts. Both kinds of artefact lead to degradation of the visual quality of the output. Our method (col.8) generates output images that still preserve all the high frequency details and are visually appealing, even with perturbed input. Similar trends are observed for other datasets including Maps (with primary translation from maps to photo) and Facade (with primary translation from segmentation maps to real photo).

For the IXI dataset (with primary translation from T1 to T2 MRI scans), we observe that the other models fail to reconstruct medically relevant structures like trigeminal-nerve (in the centre) present in the input T1 MRI scans. Moreover, high-frequency details throughout the white and grey matter in the brain are missing. In contrast, our method gracefully reconstructs many of the high-frequency details. It shows that our model is capable of generating images of good quality at higher noise levels.

3.5.3 Analyzing the Model Uncertainty

Evaluating the generalized adaptive norm. We study the performance of our method by modelling the per-pixel residuals in three ways on IXI dataset. First, i.i.d Gaussian distribution, i.e., $(\alpha_{ij}, \beta_{ij})$ is manually set to $(1, 2) \forall i, j$, which is equivalent to using fixed l_2 norm at every pixel in cycle consistency loss ($\mathcal{L}_{\alpha\beta}\big|_{\alpha=1,\beta=2}$). visual quality when given perturbed input. Second, i.i.d Laplace distribution, i.e., $(\alpha_{ij}, \beta_{ij})$ is manually set to $(1, 1) \forall i, j$, which is equivalent to using fixed l_1 norm at every pixel in cycle consistency loss ($\mathcal{L}_{\alpha\beta}\big|_{\alpha=1,\beta=1}$). Third, independent but non-identically distributed generalized Gaussian distribution (UGAC), which is equivalent to using spatially varying l_q quasi-norms where q is predicted by the network for every pixel ($\mathcal{L}_{\alpha\beta}\big|_{\text{pred}}$).

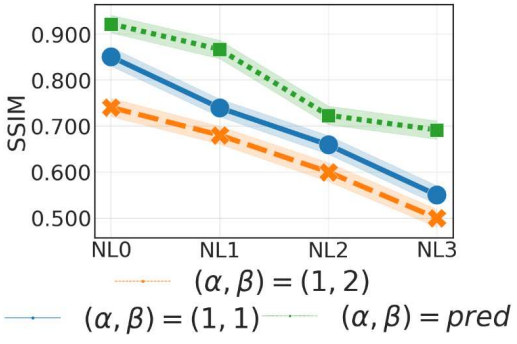

Figure 3.5: Adaptive (α, β)=pred vs. fixed (α, β)=(1, 1) and (α, β)=(1, 2) norm.

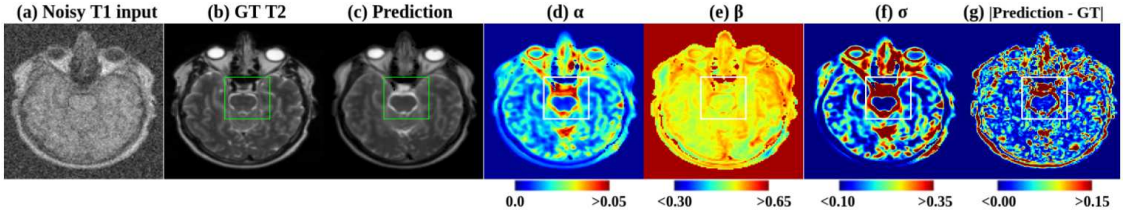

Figure 3.6: Visualization of uncertainty maps for noisy input at NL3 (sample from IXI test-set). **(a)** Noisy T1w MRI input. **(b)** Corresponding ground-truth T2w MRI. **(c)** Predicted T2w MRI. **(d)-(e)** Predicted α and β maps. **(f)** Uncertainty maps from predicted α and β maps. **(g)** Absolute residual between the prediction and the ground-truth.

Fig 3.5 shows the quantitative performance of these three variants across different noise levels for IXI datasets. We see that spatially adaptive quasi-norms perform better than fixed norms, even at higher noise levels (i.e., presence of outliers). Note that our GGD based heteroscedastic model subsumes the Gaussian ($\alpha = 1, \beta = 2$) and Laplacian ($\alpha = 1, \beta = 1$). Moreover, the heteroscedastic versions of Gaussian and Laplacian can be obtained by fixing β, i.e., for Laplacian ($\beta = 1$) and for Gaussian ($\beta = 2$), and varying α. Modeling residuals as GGD is more liberal than both homo/hetero-scedastic Gaussian/Laplacian distribution because it is able to capture all the heavier/lighter-tailed distributions (along with all possible Gaussian/Laplacian distributions) that are beyond the modeling capabilities of Gaussian/Laplacian alone.

Visualizing uncertainty maps. We visualize our uncertainty maps for the T1w MRI (domain A) to T2w MRI (domain B) translation task, on IXI dataset, with perturbations in the input (NL3). Figure 3.6-(a) shows input axial slices (T1w at NL3). The perturbations have degraded the high-frequency features (see green ROI). Figure 3.6-(b) shows the corresponding ground-truth axial slice (T2w MRI). Figure 3.6-(c) shows that our method recovers high-frequency details. However, we observe a higher contrast (compared to ground-truth) (green ROI). The subtle disparity between the contrast has been picked up by our scale-map (α) and shape-map (β) as shown in Figure 3.6-(d) and (e), respectively. Moreover, we see that, although our formulation assumes independent (but non-identically) likelihood model for the pixel level residuals, the structure in the α and the β (Figure 3.6-(d) and (e)) shows that the model learns to exploit the correlation in the neighbourhood pixels. The pixel-level variation in the α and β yields pixel-level uncertainty values in the predictions as described in Section 7.5.1.

Figure 3.6-(f) shows the uncertainty map (σ) for the predictions made by the network. We see that the disparity in the contrasts between the prediction and the ground-truth is reflected as high uncertainty in the disparity region, i.e., uncertainty is high where the reconstruction is of inferior quality, indicated by high-residual values shown in Figure 3.6-(g). The correspondence between uncertainty maps (Figure 3.6-(f)) and residual maps (Figure 3.6-(g)) suggests that uncertainty maps can be used as the proxy to residual maps (that are unavailable at the test time, as the ground-truth images will not be available) and can serve as an indicator of image quality.

Residual vs. uncertainty scores. To further study the relationship between the uncertainty maps and the residual maps across a wide variety of images, we analyze the results on IXI test set. We show the density and the scatter-plot between the residual score and uncertainty score in Figure 3.7, where every point represents a single image. For an image, the mean residual score (on the y-axis) is derived as the mean of absolute residual values for all the pixels in the image. Similarly, the uncertainty score (on the x-axis) is calculated as the mean of uncertainty values of all the pixels in that image.

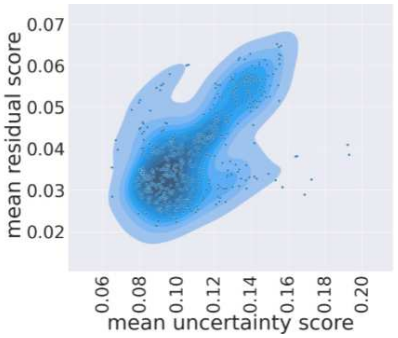

Figure 3.7: Residual scores vs. uncertainty scores.

From the plot, we see that across the test-set, the mean uncertainty score correlates positively with the mean residual score, i.e., a higher uncertainty score corresponds to a higher residual. An image with a higher residual score represents a poor-quality image. This further supports the idea that uncertainty maps derived from our method can be used as a proxy to residual that indicates the overall image quality of the output generated by our network. Therefore, the predicted uncertainty maps can potentially be used for designing a quality check triggering mechanism where poor quality predictions are evaluated by human experts.

3.6 Discussion and Conclusion

In this work, we propose an uncertainty-aware generalized cycle consistency for unpaired image translation along with uncertainty estimation. Our formulation assumes the pixel-wise independent (but non-identically) distributed likelihood model for the residuals, relaxing the i.i.d. assumption made by the previous work. However, our experiments also show that the model learns the underlying structure between the neighbourhood pixels and predicts the structured/correlated parameters for the output distribution (i.e., α, β for MRI translation shown in Figure 3.6-(d) and (e)).

We demonstrate the efficacy of the proposed method on robust unpaired image translation on various datasets spanning autonomous driving, maps, facades, and medical images consisting of MRI scans. We also demonstrate the robustness of our method by evaluating the performance in different kinds of perturbations in the input with varying severity and show that our method outperforms all the baselines by generating superior images in terms of quantitative metrics and appearance. In addition, we show that the uncertainty estimates (derived from the estimated parameters of the distribution, α and β) are faithful proxy to the residuals between the predictions and the ground truth.

It is worth noting that robustness towards various kinds of perturbation can also be achieved by data augmentation techniques that include the perturbed images in the training phase. However, this is orthogonal to the concept proposed in this work that

achieves robustness via the a new modeling technique. In principle, one could combine both the augmentation techniques and modeling techniques to obtain more robust models. In this work, we used relatively small neural networks (in terms of parameters based on UNet), while this network has not been used previously for this problem, we employ it to train our models with limited compute with reasonable training time and a lower memory footprint (details of the networks available in the Appendix A.5). This however affects the performance of the networks, and leads to images with artifacts/distortions (specially with small datasets consisting few hundred samples). Our method can be applied to deeper neural networks with more parameters/higher capacity and trained with higher resolution images, which would lead to significantly better performance, given enough compute.

An interesting avenue for further exploration is the analysis of uncertainty maps when presented with anomalous inputs, beyond perturbations, with stronger shifts between training and test data distribution which will be investigated in future.

Broader Impact

Modern deep-learning-based image translation schemes are becoming more popular. They allow the generation of synthetic datasets, e.g., for the segmentation use case in autonomous driving, faster image acquisition via algorithmic super-resolution, image enhancement in computational photography, faster and cheaper medical diagnosis by translating between different imaging modalities. However, critical areas like medical imaging and autonomous driving require methods that are robust towards various perturbations and, at the same time, can also provide uncertainty estimates in the predictions. Estimating the uncertainty in the prediction can help trigger expert intervention preventing fatal scenarios. We introduced a novel image-to-image translation model capable of estimating uncertainty along with the predictions and is shown to be beneficial in ensuring good image translation quality and good model performance on downstream tasks even in the presence of unseen noisy patterns in input images at inference time. Furthermore, our method is potentially applicable to detect ambiguities in the images. These merits could bring positive societal impacts to various critical application domains, such as medical imaging and autonomous driving.

4

BayesCap: Bayesian Identity Cap for Calibrated Uncertainty in Frozen Neural Networks

4.1 Abstract

High-quality calibrated uncertainty estimates are crucial for numerous real-world applications, especially for deep learning-based deployed ML systems. While Bayesian deep learning techniques allow uncertainty estimation, training them with large-scale datasets is an expensive process that does not always yield models competitive with non-Bayesian counterparts. Moreover, many of the high-performing deep learning models that are already trained and deployed are non-Bayesian in nature and do not provide uncertainty estimates. To address these issues, we propose `BayesCap` that learns a Bayesian identity mapping for the frozen model, allowing uncertainty estimation. `BayesCap` is a memory-efficient method that can be trained on a small fraction of the original dataset, enhancing pretrained non-Bayesian computer vision models by providing calibrated uncertainty estimates for the predictions without (i) hampering the performance of the model and (ii) the need for expensive retraining the model from scratch. The proposed method is agnostic to various architectures and tasks. We show the efficacy of our method on a wide variety of tasks with a diverse set of architectures, including image super-resolution, deblurring, inpainting, and crucial application such as medical image translation. Moreover, we apply the derived uncertainty estimates to detect out-of-distribution samples in critical scenarios like depth estimation in autonomous driving. Code is available at `https://github.com/ExplainableML/BayesCap`.

4.2 Introduction

Image enhancement and translation tasks like super-resolution [143], deblurring [134, 135], inpainting [317], colorization [326, 104], denoising [253, 200], medical image synthesis [339, 6, 40, 260, 259, 264, 246], monocular depth estimation in autonomous driving [67, 81],

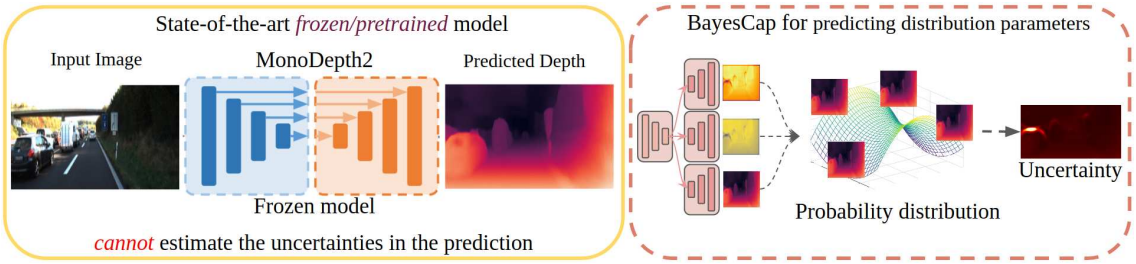

Figure 4.1: Computer vision models for image enhancements and translations deterministically map input to output without producing the uncertainty in the latter (example on the right shows depth estimation using MonoDepth2 [81]). `BayesCap` approximates the underlying distribution and adds uncertainty to the predictions of pretrained models efficiently, details in Section 4.4.3.

etc., have been effectively tackled using deep learning methods generating high-fidelity outputs. But, the respective state-of-the-art models usually learn a *deterministic* one-to-one mapping between the input and the output, without modeling the uncertainty in the prediction. For instance, a depth estimation model predicts a depth map from the input RGB image (Figure 4.1-(Left)), without providing uncertainty. In contrast, learning a probabilistic mapping between the input and the output yields the underlying distribution and provides uncertainty estimates for the predictions. This is a vital feature in safety-critical applications such as autonomous driving and medical imaging. For instance, well-calibrated uncertainty estimates can be used to trigger human/expert intervention in highly uncertain predictions, consequently preventing fatal automated decision making [39, 227, 224]. The conventional approach for obtaining uncertainty estimates is to train Bayesian models *from scratch*. However, Bayesian deep learning techniques are difficult to train and are not scalable to large volumes of high-dimensional data [50]. Moreover, they cannot be easily integrated with sophisticated deterministic architectures and training schemes tailored for specific tasks, vital to achieving state-of-the-art in vision applications [55, 80].

To address the above challenges, we enhance the predictions of pretrained state-of-the-art non-Bayesian deterministic deep models with uncertainty estimation while preserving their strong model performances. There is limited literature tackling the similar problem [278, 9, 50] but these methods do not yield well-calibrated uncertainty estimates or do not scale to high-dimensional cases such as image synthesis, translation, and enhancement.

In this work, we propose `BayesCap`, shown in Figure 4.1-(Right), an architecture agnostic, plug-and-play method to generate uncertainty estimates for pretrained models. The key idea is to train a Bayesian autoencoder over the output images of the pretrained network, approximating the underlying distribution for the output images. Due to its Bayesian design, in addition to reconstructing the input, `BayesCap` also estimates the parameters of the underlying distribution, allowing us to compute the uncertainties. `BayesCap` is highly data-efficient and can be trained on a small fraction of the original

dataset. For instance, `BayesCap` is 3-5× faster to train as compared to a Bayesian model from scratch, while still achieving uncertainty estimates that are better calibrated than the baselines.

To summarize, we make the following contributions. (1) We propose `BayesCap`, a simple method for generating post-hoc uncertainty estimates, by learning a Bayesian identity mapping, over the outputs of image synthesis/translation tasks with deterministic pretrained models. (2) `BayesCap` leads to calibrated uncertainties while retaining the performance of the underlying state-of-the-art pretrained network on a variety of tasks including super-resolution, deblurring, inpainting, and medical imaging. (3) We also show that quantifying uncertainty using `BayesCap` can help in downstream tasks such as Out-of-Distribution (OOD) detection in critical applications like autonomous driving.

4.3 Related Works

Image Enhancement and Translations. Advances in computer vision led to tackle challenging problems such as super-resolution [54, 143], denoising [253, 200], deblurring [181, 134, 135], inpainting [197, 317], depth estimation [67, 81] among others. Such problems are tackled using a diverse set of architectures and learning schemes. For instance, the popular method for super-resolution involves training a conditional *generative adversarial networks* (GANs), where the generator is conditioned with a low-resolution image and employs a pretrained VGG network [238] to enforce the content loss in the feature space along with the adversarial term from the discriminator [143]. Differently, for the inpainting task, [317] uses a conditional GAN with contextual attention and trains the network using spatially discounted reconstruction loss. In the case of monocular depth estimation, recent works exploit the left-right consistency as a cue to train the model in an unsupervised fashion [80]. While these methods are highly diverse in their architectures, training schemes, supervisory signals, etc., they typically focus on providing a deterministic one-to-one mapping which may not be ideal in many critical scenarios such as autonomous driving [170] and medical imaging [263, 264, 261]. Our `BayesCap` preserves the high-fidelity outputs provided by such deterministic pretrained models while approximating the underlying distribution of the output of such models, allowing uncertainty estimation.

Uncertainty Estimation. Bayesian deep learning models are capable of estimating the uncertainties in their prediction [138, 120]. Uncertainties can be broadly divided into two categories; (1) Epistemic uncertainty which is the uncertainty due to the model parameters [85, 24, 50, 291, 34, 138, 70]. (2) Aleatoric uncertainty which is the underlying uncertainty in the measurement itself, often estimated by approximating the per-pixel residuals between the predictions and the ground-truth using a *heteroscedastic* distribution whose parameters are predicted as the output of the network which is trained *from scratch* to maximize the likelihood of the system [120, 158, 10, 278, 284, 142]. While epistemic uncertainty is important in low-data regimes as parameter estimation becomes noisy, however, this is often not the case in computer vision settings with large scale datasets

where aleatoric uncertainty is the critical quantity [120]. However, it is expensive to train these models and they often perform worse than their deterministic counterparts [192, 212, 50]. Unlike these works, `BayesCap` is a fast and efficient method to estimate uncertainty over the predictions of a pretrained deterministic model.

Post-hoc Uncertainty Estimation. While this has not been widely explored, some recent works [50, 59] have tried to use the Laplace approximation for this purpose. However, these methods computes the Hessian which is not feasible for high-dimensional modern problems in computer vision [326, 197, 67, 134, 200]. Another line of work to tackle this problem is test-time data augmentation [278, 9] that perturbs the inputs to obtain multiple outputs leading to uncertainties. However, these estimates are often poorly calibrated [75]. It is of paramount importance that the uncertainty estimates are well calibrated [131, 87, 140, 142, 199, 323]. In many high-dimensional computer vision problems the per-pixel output is often a continuous value [197, 134, 143, 326], i.e., the problem is regression in nature. Recent works focused on *Uncertainty Calibration Error* that generalizes to high dimensional regression [131, 147, 140, 142]. Unlike prior works [9, 50, 278, 59], `BayesCap` scales to high-dimensional tasks, providing well-calibrated uncertainties.

4.4 Methodology: BayesCap - Bayesian Identity Cap

We first describe the problem formulation in Section 7.4.1, and preliminaries on uncertainty estimation in Section 4.4.2. In Section 4.4.3, we describe construction of `BayesCap` that models a probabilistic identity function capable of estimating the high-dimensional complex distribution from the frozen deterministic model, estimating calibrated uncertainty for the predictions.

4.4.1 Problem formulation

Let $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$ be the training set with pairs from domain $\mathbf{X}$ and $\mathbf{Y}$ (i.e., $\mathbf{x}_i \in \mathbf{X}, \mathbf{y}_i \in \mathbf{Y}, \forall i$), where $\mathbf{X}, \mathbf{Y}$ lies in $\mathbb{R}^m$ and $\mathbb{R}^n$, respectively. While our proposed solution is valid for data of arbitrary dimension, we present the formulation for images with applications for image enhancement and translation tasks, such as super-resolution, inpainting, etc. Therefore, $(\mathbf{x}_i, \mathbf{y}_i)$ represents a pair of images, where $\mathbf{x}_i$ refers to the input and $\mathbf{y}_i$ denotes the transformed/enhanced output. For instance, in super-resolution $\mathbf{x}_i$ is a low-resolution image and $\mathbf{y}_i$ its high-resolution version. Let $\mathbf{\Psi}(\cdot; \theta) : \mathbb{R}^m \rightarrow \mathbb{R}^n$ represent a Deep Neural Network parametrized by θ that maps images from the set $\mathbf{X}$ to the set $\mathbf{Y}$, e.g. from corrupted to the non-corrupted/enhanced output images.

We consider a real-world scenario, where $\mathbf{\Psi}(\cdot; \theta)$ has already been trained using the dataset $\mathcal{D}$ and it is in a *frozen state* with parameters set to the learned optimal parameters θ^*. In this state, given an input $\mathbf{x}$, the model returns a point estimate of the output, i.e., $\hat{\mathbf{y}} = \mathbf{\Psi}(\mathbf{x}; \theta^*)$. However, point estimates do not capture the distributions of the output ($\mathcal{P}_{\mathbf{Y}|\mathbf{X}}$) and thus the uncertainty in the prediction that is crucial in many real-world

applications [120]. Therefore, we propose to estimate $\mathcal{P}_{\mathbf{Y}|\mathbf{X}}$ for the pretrained model in a fast and cheap manner, quantifying the uncertainties of the output without re-training the model itself.

4.4.2 Preliminaries: Uncertainty Estimation

To understand the functioning of our `BayesCap` that produces uncertainty estimates for the *frozen or pretrained* neural networks, we first consider a model trained from scratch to address the target task and estimate uncertainty. Let us denote this model by $\mathbf{\Psi}_s(\cdot;\zeta) : \mathbb{R}^m \rightarrow \mathbb{R}^n$, with a set of trainable parameters given by ζ. To capture the *irreducible* (i.e., aleatoric) uncertainty in the output distribution $\mathcal{P}_{Y|X}$, the model must estimate the parameters of the distribution. These are then used to maximize the likelihood function. That is, for an input $\mathbf{x}_i$, the model produces a set of parameters representing the output given by, $\{\hat{\mathbf{y}}_i, \hat{v}_i \dots \hat{\rho}_i\} := \mathbf{\Psi}_s(\mathbf{x}_i;\zeta)$, that characterizes the distribution $\mathcal{P}_{Y|X}(\mathbf{y};\{\hat{\mathbf{y}}_i, \hat{v}_i \dots \hat{\rho}_i\})$, such that $\mathbf{y}_i \sim \mathcal{P}_{Y|X}(\mathbf{y};\{\hat{\mathbf{y}}_i, \hat{v}_i \dots \hat{\rho}_i\})$. The likelihood $\mathscr{L}(\zeta;\mathcal{D}) := \prod_{i=1}^{N} \mathcal{P}_{Y|X}(\mathbf{y}_i;\{\hat{\mathbf{y}}_i, \hat{v}_i \dots \hat{\rho}_i\})$ is then maximized in order to estimate the optimal parameters of the network. Moreover, the distribution $\mathcal{P}_{Y|X}$ is often chosen such that uncertainty can be estimated using a closed form solution $\mathscr{F}$ depending on the estimated parameters of the neural network, i.e.,

$$\{\hat{\mathbf{y}}_i, \hat{v}_i \dots \hat{\rho}_i\} := \mathbf{\Psi}_s(\mathbf{x}_i;\zeta) \tag{4.1}$$

$$\zeta^* := \underset{\zeta}{\operatorname{argmax}}\ \mathscr{L}(\zeta;\mathcal{D}) = \underset{\zeta}{\operatorname{argmax}} \prod_{i=1}^{N} \mathcal{P}_{Y|X}(\mathbf{y}_i;\{\hat{\mathbf{y}}_i, \hat{v}_i \dots \hat{\rho}_i\}) \tag{4.2}$$

$$\text{Uncertainty}(\hat{\mathbf{y}}_i) = \mathscr{F}(\hat{v}_i \dots \hat{\rho}_i) \tag{4.3}$$

It is common to use a *heteroscedastic* Gaussian distribution for $\mathcal{P}_{Y|X}$ [120, 278], in which case $\mathbf{\Psi}_s(\cdot;\zeta)$ is designed to predict the *mean* and *variance* of the Gaussian distribution, i.e., $\{\hat{\mathbf{y}}_i, \hat{\sigma}_i^2\} := \mathbf{\Psi}_s(\mathbf{x}_i;\zeta)$, and the predicted *variance* itself can be treated as uncertainty in the prediction. The optimization problem becomes,

$$\zeta^* = \underset{\zeta}{\operatorname{argmax}} \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\hat{\sigma}_i^2}} e^{-\frac{|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2}{2\hat{\sigma}_i^2}} = \underset{\zeta}{\operatorname{argmin}} \sum_{i=1}^{N} \frac{|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2}{2\hat{\sigma}_i^2} + \frac{\log(\hat{\sigma}_i^2)}{2} \tag{4.4}$$

$$\text{Uncertainty}(\hat{\mathbf{y}}_i) = \hat{\sigma}_i^2. \tag{4.5}$$

The above equation models the per-pixel residual (between the prediction and the ground-truth) as a Gaussian distribution. However, this may not always be fit, especially in the presence of outliers and artefacts, where the residuals often follow heavy-tailed distributions. Recent works such as [261, 264] have shown that heavy-tailed distributions can be modeled as a heteroscedastic generalized Gaussian distribution, in which case $\mathbf{\Psi}_s(\cdot;\zeta)$ is designed to predict the *mean* ($\hat{\mathbf{y}}_i$), *scale* ($\hat{\alpha}_i$), and *shape* ($\hat{\beta}_i$) as trainable parameters,

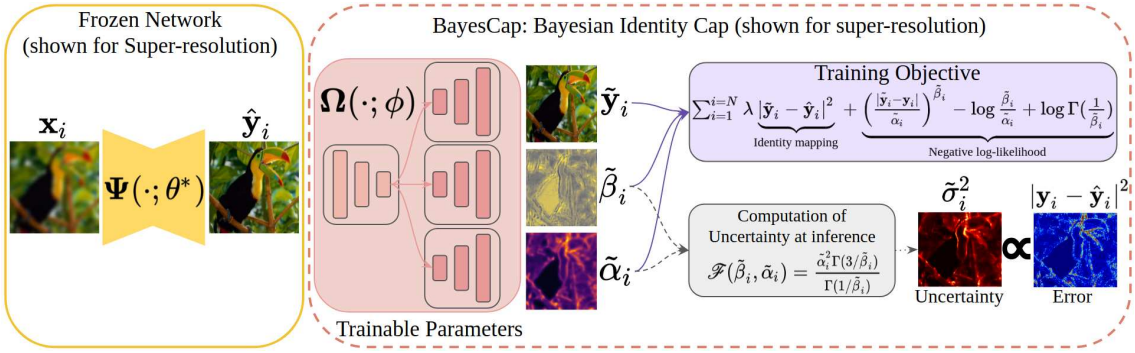

Figure 4.2: `BayesCap` ($\mathbf{\Omega}(\cdot;\phi)$) in tandem with the pretrained network with frozen parameters ($\mathbf{\Psi}(\cdot;\theta^*)$) (details in Section 4.4.3). While the pretrained network cannot estimate the uncertainty, the proposed `BayesCap` feeds on the output of the pretrained network and maps it to the underlying probability distribution that allows computation of well calibrated uncertainty estimates.

i.e., $\{\hat{\mathbf{y}}_i, \hat{\alpha}_i, \hat{\beta}_i\} := \mathbf{\Psi}_s(\mathbf{x}_i;\zeta)$,

$$\zeta^* := \underset{\zeta}{\text{argmax}}\, \mathscr{L}(\zeta) = \underset{\zeta}{\text{argmax}} \prod_{i=1}^{N} \frac{\hat{\beta}_i}{2\hat{\alpha}_i \Gamma(\frac{1}{\hat{\beta}_i})} e^{-(|\hat{\mathbf{y}}_i - \mathbf{y}_i|/\hat{\alpha}_i)^{\hat{\beta}_i}} = \underset{\zeta}{\text{argmin}} -\log \mathscr{L}(\zeta)$$

$$= \underset{\zeta}{\text{argmin}} \sum_{i=1}^{N} \left(\frac{|\hat{\mathbf{y}}_i - \mathbf{y}_i|}{\hat{\alpha}_i}\right)^{\hat{\beta}_i} - \log\frac{\hat{\beta}_i}{\hat{\alpha}_i} + \log\Gamma(\frac{1}{\hat{\beta}_i}) \tag{4.6}$$

$$\text{Uncertainty}(\hat{\mathbf{y}}_i) = \frac{\hat{\alpha}_i^2 \Gamma(\frac{3}{\hat{\beta}_i})}{\Gamma(\frac{1}{\hat{\beta}_i})}. \tag{4.7}$$

Here $\Gamma(z) = \int_0^\infty x^{z-1}e^{-x}dx, \forall z > 0$, represents the Gamma function [8]. While the above formulation (Eq. (4.4)-(4.7)) shows the dependence of various predicted distribution parameters on one another when maximizing the likelihood, it requires training the model from scratch, that we want to avoid. In the following, we describe how we address this problem through our `BayesCap`.

4.4.3 Constructing BayesCap

In the above, $\mathbf{\Psi}_s(\cdot;\zeta)$ was trained from scratch to predict all the parameters of distribution and does *not* leverage the *frozen* model $\mathbf{\Psi}(\cdot;\theta^*)$ estimating $\mathbf{y}_i$ using $\hat{\mathbf{y}}_i$ in a deterministic fashion. To circumvent the training from scratch, we notice that one only needs to estimate the remaining parameters of the underlying distribution. Therefore, to augment the frozen point estimation model , we learn a Bayesian identity mapping represented by $\mathbf{\Omega}(\cdot;\phi) : \mathbb{R}^n \rightarrow \mathbb{R}^n$, that reconstructs the output of the frozen model $\mathbf{\Psi}(\cdot;\theta^*)$ and also produces the parameters of the distribution modeling the reconstructed output. We refer to this network as `BayesCap` (schematic in Figure 6.2). As in Eq. (4.7), we use

heteroscedastic generalized Gaussian to model output distribution, i.e.,

$$\mathbf{\Omega}(\hat{\mathbf{y}}_i = \mathbf{\Psi}(\mathbf{x}_i; \theta^*); \phi) = \{\tilde{\mathbf{y}}_i, \tilde{\alpha}_i, \tilde{\beta}_i\}, \text{ with } \mathbf{y}_i \sim \frac{\tilde{\beta}_i}{2\tilde{\alpha}_i \Gamma(\frac{1}{\tilde{\beta}_i})} e^{-(|\tilde{\mathbf{y}}_i - \mathbf{y}_i|/\tilde{\alpha}_i)^{\tilde{\beta}_i}} \tag{4.8}$$

To enforce the identity mapping, for every input $\mathbf{x}_i$, we regress the reconstructed output of the `BayesCap` ($\tilde{\mathbf{y}}_i$) with the output of the pretrained base network ($\hat{\mathbf{y}}_i$). This ensures that, the distribution predicted by `BayesCap` for an input $\mathbf{x}_i$, i.e., $\mathbf{\Omega}(\mathbf{\Psi}(\mathbf{x}_i; \theta^*); \phi)$, is such that the point estimates $\tilde{\mathbf{y}}_i$ match the point estimates of the pretrained network $\hat{\mathbf{y}}_i$. Therefore, as the quality of the reconstructed output improves, the uncertainty estimated by $\mathbf{\Omega}(\cdot; \phi)$ also approximates the uncertainty for the prediction made by the pretrained $\mathbf{\Psi}(\cdot; \theta^*)$, i.e.,

$$\tilde{\mathbf{y}}_i \to \hat{\mathbf{y}}_i \implies \tilde{\sigma}_i^2 = \frac{\tilde{\alpha}_i^2 \Gamma(3/\tilde{\beta}_i)}{\Gamma(1/\tilde{\beta}_i)} \to \hat{\sigma}_i^2 \tag{4.9}$$

To train $\mathbf{\Omega}(\cdot; \phi)$ and obtain optimal parameters (ϕ^*), we minimize the fidelity term between $\tilde{\mathbf{y}}_i$ and $\hat{\mathbf{y}}_i$, along with the negative log-likelihood for $\mathbf{\Omega}(\cdot; \phi)$, i.e.,

$$\phi^* = \underset{\phi}{\text{argmin}} \sum_{i=1}^{N} \lambda \underbrace{|\tilde{\mathbf{y}}_i - \hat{\mathbf{y}}_i|^2}_{\text{Identity mapping}} + \underbrace{\left(\frac{|\tilde{\mathbf{y}}_i - \mathbf{y}_i|}{\tilde{\alpha}_i}\right)^{\tilde{\beta}_i} - \log \frac{\tilde{\beta}_i}{\tilde{\alpha}_i} + \log \Gamma(\frac{1}{\tilde{\beta}_i})}_{\text{Negative log-likelihood}} \tag{4.10}$$

Here λ represents the hyperparameter controlling the contribution of the fidelity term in the overall loss function. Extremely high λ will lead to improper estimation of the ($\tilde{\alpha}$) and ($\tilde{\beta}$) parameters as other terms are ignored. Eq. (4.10) allows `BayesCap` to estimate the underlying distribution and uncertainty.

Eq. (4.8) and (4.10) show that the construction of $\mathbf{\Omega}$ is independent of $\mathbf{\Psi}$ and the $\mathbf{\Omega}$ always performs the Bayesian identity mapping regardless of the task performed by $\mathbf{\Psi}$. This suggests that task specific tuning will have minimal impact on the performance of $\mathbf{\Omega}$. In our experiments, we employed the same network architecture for `BayesCap` with the same set of hyperparameters, across different tasks and it achieves state-of-the-art uncertainty estimation results without task specific tuning (as shown in Section 7.5), highlighting that `BayesCap` is not sensitive towards various design choices including architecture, learning-rate, etc.

4.5 Experiments

We first describe our experimental setup (i.e., datasets, tasks, and evaluation metrics) in Section 7.5.1. We compare our model to a wide variety of state-of-the-art methods quantitatively and qualitatively in Section 7.5.2. Finally in Section 4.5.3, we provide an ablation analysis along with a real world application of `BayesCap` for detecting out-of-distribution samples.

4.5.1 Tasks and Datasets

We show the efficacy of our `BayesCap` on various image enhancement and translation tasks including super-resolution, deblurring, inpainting, and MRI translation, as detailed below. In general, image enhancement and translation tasks are highly ill-posed problems as an *injective function* between input and output may not exist [160, 261], thereby necessitating the need to learn a probabilistic mapping to quantify the uncertainty and indicate poor reconstruction of the output images. For each task we choose a well established deterministic pretrained network, for which we estimate uncertainties.

Super-resolution. The goal is to map low-resolution images to their high-resolution counterpart. We choose pretrained SRGAN [143] as our base model $\mathbf{\Psi}(\cdot;\theta^*)$. The `BayesCap` model $\mathbf{\Omega}(\cdot;\phi)$ is trained on ImageNet patches sized 84×84 to perform $4\times$ super-resolution. The resulting combination of SRGAN and `BayesCap` is evaluated on the Set5 [22], Set14 [320], and BSD100 [167] datasets.

Deblurring. The goal is to remove noise from images corrupted with blind motion. We use the pretrained DeblurGANv2 [135] which shows improvements over the original DeblurGAN [134]. The `BayesCap` model is evaluated on the GoPro dataset [181], using standard train/test splits.

Inpainting. The goal is to fill masked regions of an input image. We use pretrained DeepFillv2 [316], that improves over DeepFill [317], as the base model for inpainting. Both the original base model and the `BayesCap` are trained and tested on the standard train/test split of Places365 dataset [331].

MRI Translation. We predict the T2 MRI from T1 MRI, an important problem in medical imaging as discussed in [30, 27, 102, 312, 314]. We use the pretrained deterministic UNet as base model [264, 218]. Both the base model and `BayesCap` are trained and tested on IXI datatset [214] following [264].

Baselines. For all tasks, we compare `BayesCap` against 7 methods in total, out of which 6 baselines can estimate uncertainty of a pretrained model without re-training and one baseline modifies the base network and train it from *scratch* to estimate the uncertainty. The first set of baselines belong to *test-time data augmentation* (`TTDA`) technique [9, 281, 278], where we generate multiple perturbed copies of the input and use the set of corresponding outputs to compute the uncertainty. We consider three different ways of perturbing the input, (i) per-pixel noise perturbations `TTDAp` [9, 281], (ii) affine transformations `TTDAa` [9, 281] and (iii) random corruptions from Gaussian blurring, contrast enhancement, and color jittering (`TTDAc`) [9, 281]. As additional baseline, we also consider `TTDApac` that generates the multiple copies by combining pixel-level perturbations, affine transformations, and corruptions as described above.

Another set of baselines uses dropout [244, 118, 70, 141, 164] before the final predictions. This is possible even for the models *that are not originally trained with dropout*. We refer to this model as `DO`. In addition, we consider a baseline that combines dropout with test-time data augmentation (`DOpac`). Finally, we also compare against a model trained from scratch

to produce the uncertainty as described in [120]. We refer to this as `Scratch`.

Metrics. We evaluate the performance of various models on two kinds of metrics (i) image reconstruction quality and (ii) predicted uncertainty calibration quality. To measure reconstruction quality, we use SSIM [289] and PSNR. For inpainting we also show mean ℓ_1 and ℓ_2 error, following the convention in original works [317, 316]. We emphasize that all the methods, *except* `Scratch`, can use the output of the pretrained base model and only derive the uncertainty maps using different estimation techniques described above. Therefore image reconstruction quality metrics like SSIM, PSNR, mean ℓ_1 and ℓ_2 error remain the same as that of base network. However, `Scratch` method *does not* have access to the pretrained model, therefore it has to use its own predicted output and uncertainty estimates.

To quantify the quality of the uncertainty, we use the *uncertainty calibration error* (UCE) as described in [140, 87] for regression tasks. It measures the discrepancy between the predictive error and predictive uncertainty, given by, $\texttt{UCE} := \sum_{m=1}^{M} |B_m|N|\texttt{err}(B_m) - \texttt{uncer}(B_m)|$, where B_m is one of the uniformly separated bins, $\texttt{err}(B_m) := 1|B_m| \sum_{i \in B_m} ||\hat{\mathbf{y}}_i - \mathbf{y}_i||^2$, and $\texttt{uncer}(B_m) := 1|B_m| \sum_{i \in B_m} \hat{\sigma}_i^2$. We also use *correlation coefficient* (C.Coeff.) between the error and the uncertainty, as high correlation is desirable.

Implementation Details. We optimize Eq. 4.10 using the Adam optimizer [122] and a batch size of 2 with images that are resized to 256×256. During training we exponentially anneal the hyperparameter λ that is initially set to 10. This guides the `BayesCap` to learn the identity mapping in the beginning, and gradually learn the optimal parameters of the underlying distribution via maximum likelihood estimation.

4.5.2 Results

Super-resolution. Table 4.1 shows the image reconstruction performance along with the uncertainty calibration performance on the Set5, Set14, and the BSD100 datasets for all the methods. We see that `BayesCap` significantly outperforms all other methods in terms of UCE and C.Coeff while retaining the image reconstruction performance of the base model. For instance, across all the 3 sets, the correlation coefficient is always greater than 0.4 showing that the error and the uncertainty are correlated to a high degree. The model trained from scratch has a correlation coefficient between 0.22-0.31 which is lower than `BayesCap`. The baselines based on dropout and test-time data augmentation show nearly no correlation between the uncertainty estimates and the error (C.Coeff. of 0.03-0.17). A similar trend can be seen with the UCE where `BayesCap` has the best UCE scores followed by the model trained from scratch, while the test-time data augmentation and the dropout baselines have a very high UCE (between 0.33 and 0.83) suggesting poorly calibrated uncertainty estimates.

Qualitatively, Figure 7.3 shows the prediction of the pretrained SRGAN along with the predictions of the `BayesCap` showing per-pixel estimated distribution parameters along with the uncertainty map on a sample from Set5 dataset. High correlation between

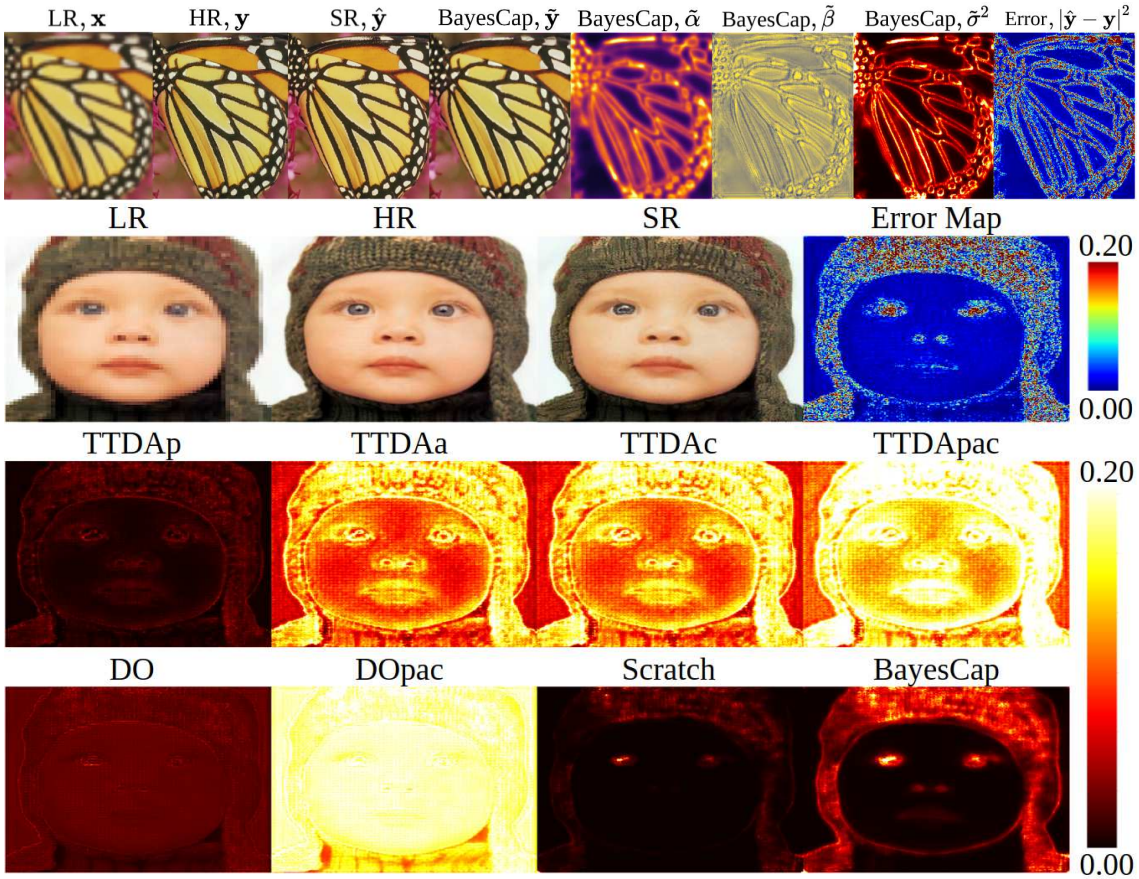

Figure 4.4: Qualitative example showing the results of the pre-trained SRGAN model along with the uncertainty maps produced by `BayesCap` and the other methods. Uncertainty derived from `BayesCap` has better correlation with the error.

the per-pixel predictive error of SRGAN and uncertainties from `BayesCap` suggests that `BayesCap` produces well-calibrated uncertainty estimates. Moreover, Figure 4.4 shows that uncertainty produced by other baselines are not in agreement with the error (e.g., `TTDAp` does not show high uncertainty within the eye, where error is high) indicating that they are poorly calibrated.

Deblurring. We report the results on the GoPro dataset in Table 4.2. DeblurGANv2 achieves significantly better results than `Scratch` (29.55 vs 26.16 PSNR). In terms of UCE, `BayesCap` outperforms all the methods by achieving a low score of 0.038. While the `Scratch` is close and achieves a UCE of 0.076, all the other methods have a UCE that is nearly 10 times higher suggesting that `BayesCap` estimates the most calibrated uncertainty. This is also visible in Figure 4.5-(left) where the uncertainties provided by `BayesCap` is correlated with the error (C.Coeff. of 0.32) unlike methods from `TTDA` and `DO` class that have very low correlation between the uncertainty and the error (C.Coeff of 0.03 - 0.17). While `Scratch` achieves a reasonable score, second only to `BayesCap`, in terms of UCE (0.076 vs. 0,038) and C.Coeff (0.21 vs. 0.32), it has much poorer image reconstruction output with a PSNR of 26.16 and SSIM of 0.8136. The poor reconstruction also justifies

D	Metrics	SRGAN	TTDAp	TTDAa	TTDAc	TTDApac	DO	DOpac	Scratch	BayesCap
Set5	PSNR↑	29.40	29.40	29.40	29.40	29.40	29.40	29.40	27.83	29.40
	SSIM↑	0.8472	0.8472	0.8472	0.8472	0.8472	0.8472	0.8472	0.8166	0.8472
	UCE↓	**NA**	0.39	0.40	0.42	0.47	0.33	0.36	0.035	**0.014**
	C.Coeff↑	**NA**	0.17	0.13	0.08	0.03	0.05	0.07	0.28	**0.47**
Set14	PSNR↑	26.02	26.02	26.02	26.02	26.02	26.02	26.02	25.31	26.02
	SSIM↑	0.7397	0.7397	0.7397	0.7397	0.7397	0.7397	0.7397	0.7162	0.7397
	UCE↓	**NA**	0.57	0.63	0.61	0.69	0.48	0.52	0.048	**0.017**
	C.Coeff↑	**NA**	0.07	0.04	0.04	0.06	0.08	0.04	0.22	**0.42**
BSD100	PSNR↑	25.16	25.16	25.16	25.16	25.16	25.16	25.16	24.39	25.16
	SSIM↑	0.6688	0.6688	0.6688	0.6688	0.6688	0.6688	0.6688	0.6297	0.6688
	UCE↓	**NA**	0.72	0.77	0.81	0.83	0.61	0.64	0.057	**0.028**
	C.Coeff↑	**NA**	0.13	0.09	0.11	0.09	0.10	0.08	0.31	**0.45**

Table 4.1: Quantitative results showing the performance of pretrained SRGAN in terms of PSNR and SSIM, along with the quality of of uncertaintiy maps obtained by `BayesCap` and other baselines, in terms of UCE and Correlation Coefficient (C.Coeff). All results on 3 datasets including Set5, Set14, and BSD100.

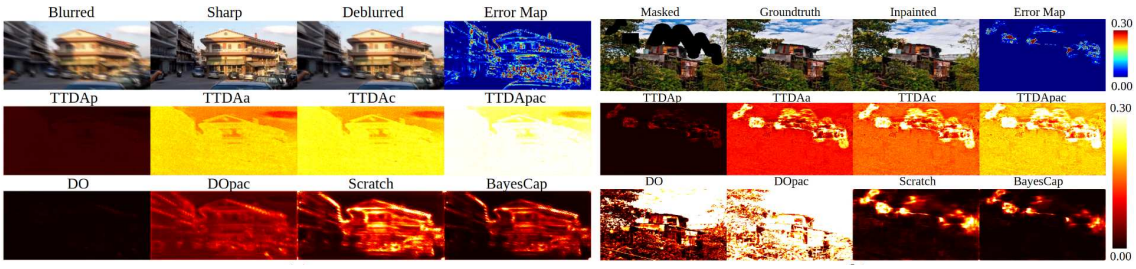

Figure 4.5: Qualitative example showing the results of the pretrained DeblurGANv2 and DeepFillv2 on image deblurring (left) and inpainting (right) tasks along with the uncertainty maps produced by different methods.

D	Metrics	DeblurGANv2	TTDAp	TTDAa	TTDAc	TTDApac	DO	DOpac	Scratch	BayesCap
GoPro	PSNR↑	29.55	29.55	29.55	29.55	29.55	29.55	29.55	26.16	29.55
	SSIM↑	0.9340	0.9340	0.9340	0.9340	0.9340	0.9340	0.9340	0.8136	0.9340
	UCE↓	**NA**	0.44	0.45	0.49	0.53	0.52	0.59	0.076	**0.038**
	C.Coeff↑	**NA**	0.17	0.13	0.08	0.03	0.05	0.07	0.21	**0.32**

Table 4.2: Results showing the performance of pretrained DeblurGANv2 in terms of PSNR and SSIM, along with the quality of of uncertaintiy obtained by `BayesCap` and other methods, in terms of UCE and C.Coeff on GoPro dataset.

the relatively higher uncertainty values for `Scratch` (as seen in Figure 4.5-(left)) when compared to `BayesCap`.

Inpainting. Table 4.3 shows the results on Places365 dataset. The pretrained base model DeepFillv2 [316] achieves a mean L1 error of 9.1%, however `Scratch` is much worse, achieving a mean L1 error of 15.7%. This again demonstrates that training a Bayesian model from scratch often does not replicate the performance of deterministic

D	Metrics	DeepFillv2	TTDAp	TTDAa	TTDAc	TTDApac	DO	DOpac	Scratch	BayesCap
Places365	m. ℓ_1 err.↓	9.1%	9.1%	9.1%	9.1%	9.1%	9.1%	9.1%	15.7%	9.1%
	m. ℓ_2 err.↓	1.6%	1.6%	1.6%	1.6%	1.6%	1.6%	1.6%	5.8%	1.6%
	PSNR↑	18.34	18.34	18.34	18.34	18.34	18.34	18.34	17.24	18.34
	SSIM↑	0.6285	0.6285	0.6285	0.6285	0.6285	0.6285	0.6285	0.6032	0.6285
	UCE↓	**NA**	0.63	0.88	0.87	0.93	1.62	1.49	0.059	**0.011**
	C.Coeff↑	**NA**	0.26	0.11	0.12	0.08	0.09	0.12	0.44	**0.68**

Table 4.3: Performance of pretrained DeepFillv2 in terms of mean ℓ_1 error, mean ℓ_2 error, PSNR and SSIM, along with the quality of uncertainty obtained by `BayesCap` and other methods, in terms of UCE and C.Coeff on Places365 dataset.

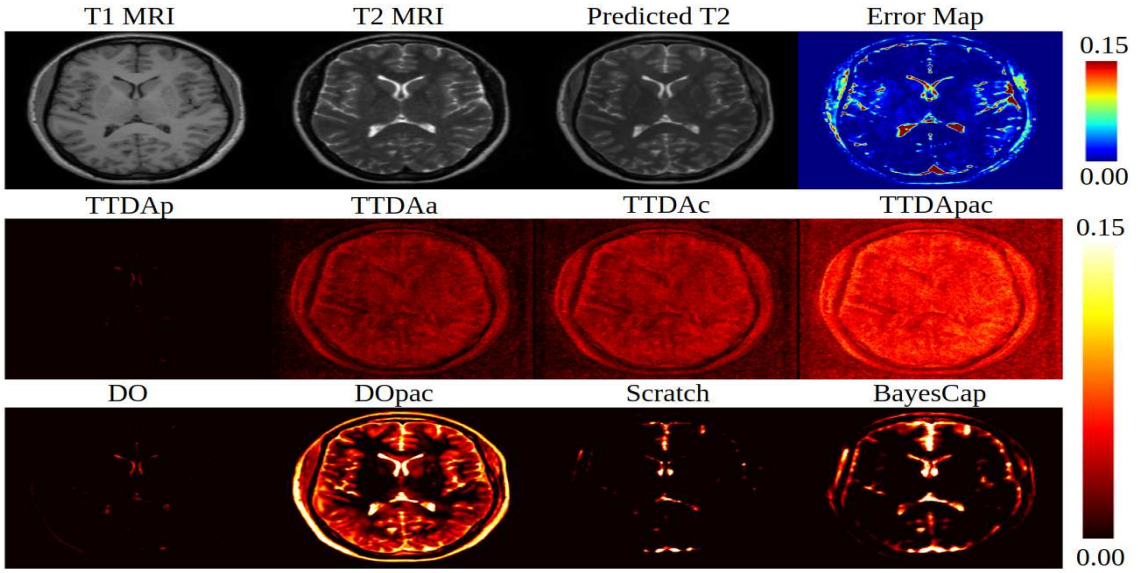

Figure 4.6: Qualitative example showing the results of the pretrained UNet for T1 to T2 MRI translation along with the uncertainty produced by different methods.

counterparts. Also, `BayesCap` retains the reconstruction performance of DeepFillv2 [316] and provides well-calibrated uncertainties, as demonstrated by highest C.Coeff. (0.68), and the lowest UCE (0.011). Methods belonging to `TTDA` and `DO` classes are unable to provide good uncertainties (C.Coeff. of 0.08-0.26). The example in Figure 4.5-(right) also illustrates an interesting phenomenon. Although, the uncertainties are predicted for the entire image, we see that `BayesCap` automatically learns to have extremely low uncertainty values outside the masked region which is perfectly reconstructed. Within the masked region, uncertainty estimates are highly correlated with the error.

MRI Translation. We perform T1 to T2 MRI translation as described in [264, 261] which has an impact in clinical settings by reducing MRI acquisition times [30, 27, 102, 312, 314]. The quantitative results on the IXI dataset are show in Table 4.4. The pretrained base model employing U-Net architecture, achieves a SSIM score of 0.9272 which is nearly matched by `Scratch` (0.9169). However, we see that `BayesCap` performs better than `Scratch` in terms of UCE (0.036 vs. 0.029) and C.Coeff (0.52 vs. 0.58). Other methods are poorly calibrated,

D	Metrics	UNet	TTDAp	TTDAa	TTDAc	TTDApac	DO	DOpac	Scratch	BayesCap
IXI	PSNR↑	25.70	25.70	25.70	25.70	25.70	25.70	25.70	25.50	25.70
	SSIM↑	0.9272	0.9272	0.9272	0.9272	0.9272	0.9272	0.9272	0.9169	0.9272
	UCE↓	**NA**	0.53	0.46	0.41	0.44	0.38	0.40	0.036	**0.029**
	C.Coeff↑	**NA**	0.05	0.14	0.16	0.08	0.13	0.47	0.52	**0.58**

Table 4.4: Performance of pretrained UNet for MRI translation in terms of PSNR and SSIM, along with the quality of of uncertainty obtained by `BayesCap` and other methods, in terms of UCE and C.Coeff on IXI Dataset.

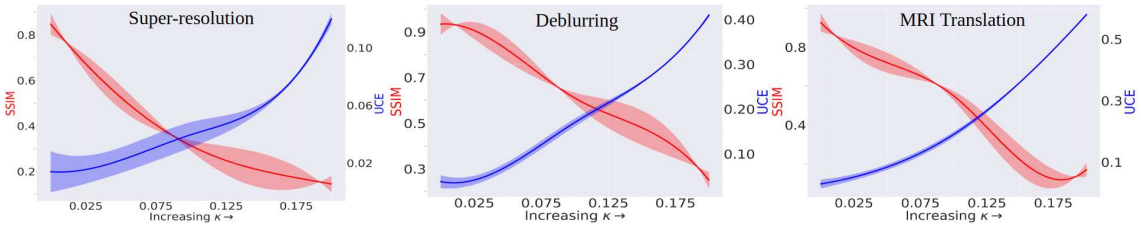

Figure 4.7: *Impact of the identity mapping*. Degrading the quality of the identity mapping (SSIM) at inference, leads to poorly calibrated uncertainty (UCE). κ represents the magnitude of noise used for degrading the identity mapping.

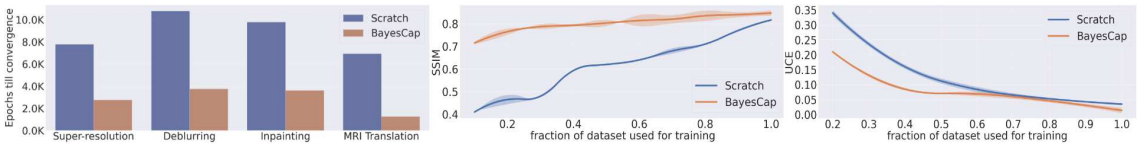

Figure 4.8: `BayesCap` can be trained to achieve optimal performance in fewer epochs (left), while being more data-efficient (achieves better results with fewer samples) as compared to `Scratch` (middle and right), shown for super-resolution.

as indicated by high UCE and low C.Coeff. This is also evident from Figure 4.6, indicating high correlation between the `BayesCap` uncertainty and error, while low correlation of the same for other methods.

4.5.3 Ablation Studies

As discussed in Section 4.4.3, `BayesCap` can help in estimating uncertainty for the frozen model only if `BayesCap` provides perfect reconstruction. To study this, we design an experiment that deteriorates the identity mapping learned by `BayesCap`, leading to poor uncertainty estimates. Moreover, we also demonstrate that `BayesCap` is much more data and compute efficient when compared to `Scratch` (i.e., model capable of estimating the uncertainty, trained from scratch).

Figure 4.7 shows that preserving the identity mapping is essential to providing well-calibrated post-hoc uncertainty estimates using `BayesCap`. Here, we gradually feed increasingly noisy samples (corrupted by zero-mean Gaussian with variance given by κ^2) to the model that leads to poor reconstruction by `BayesCap` and as a result degrades the identity mapping. As the quality of the identity mapping degrades (decreasing SSIM),

we see that quality of uncertainty also degrades (increasing UCE). For instance, for super-resolution, with zero noise in the input the reconstruction quality of the `BayesCap` is at its peak (SSIM, 0.8472) leading to almost identical mapping. This results in well-calibrated uncertainty maps derived from `BayesCap` (UCE, 0.014). However, with $\kappa = 0.15$, the reconstruction quality decreases sharply (SSIM of 0.204) leading to poorly calibrated uncertainty (UCE of 0.0587), justifying the need for the identity mapping.

We also show that `BayesCap` is more efficient than training a Bayesian model from scratch both in terms of time and data required to train the model in Figure 4.8-(Left). On all the 4 tasks, `BayesCap` can be trained 3-5× faster than `Scratch`. For super-resolution, we show that `BayesCap` can achieve competitive results even when trained on a fraction of the dataset, whereas the performance of `Scratch` reduces sharply in low data regime, as shown in Figure 4.8-(Middle and Right). For instance, with just 50% of training data, `BayesCap` performs 33% better than `Scratch` in terms of SSIM. This is because `BayesCap` learns the autoencoding task, whereas `Scratch` learns the original translation task.

4.5.4 Application: Out-of-Distribution Analysis

The proposed `BayesCap` focuses on estimating well-calibrated uncertainties of pretrained image regression models in a post-hoc fashion. This can be crucial in many critical real-world scenarios such as autonomous driving and navigation [302, 120, 286, 174]. We consider monocular depth estimation (essential for autonomous driving) and show that the estimated uncertainty can help in detecting *out-of-distribution* (OOD) samples. We take MonoDepth2 [81] that is trained on the KITTI dataset [76]. The resulting model with `BayesCap` is evaluated on the validation sets of the KITTI dataset, the India Driving Dataset(IDD) [271] as well as the Places365 dataset [331]. The KITTI dataset captures images from German cities, while the India Driving Dataset has images from Indian roads. Places365 consists of images for scene recognition and is vastly different from driving datasets. Intuitively, both IDD and Places365 represent OOD as there is a shift in distribution when compared to training data (KITTI), this is captured in Figure 4.9-(a,b,c,d), representing degraded depth and uncertainty on OOD images. The uncertainties also reflect this with increasingly higher values for OOD samples as also shown in the bar plot (Figure 4.9-(d)). This suggests that mean uncertainty value for an image can be used to detect if it belongs to OOD.

To quantify this, we plot the ROC curve for OOD detection on a combination of the KITTI (in-distribution) samples and IDD and Places365 (out-of-distribution) samples in Figure 4.9-(e). Additionally, we compare using uncertainty for OOD detection against (i) using intermediate features from the pretrained model [211] and (ii) using features from the bottleneck of a trained autoencoder [332]. Samples are marked OOD if the distance between the features of the samples and mean feature of the KITTI validation set is greater than a threshold. We clearly see that using the mean uncertainty achieves far superior results (AUROC of 0.96), against the pretrained features (AUROC of 0.82)

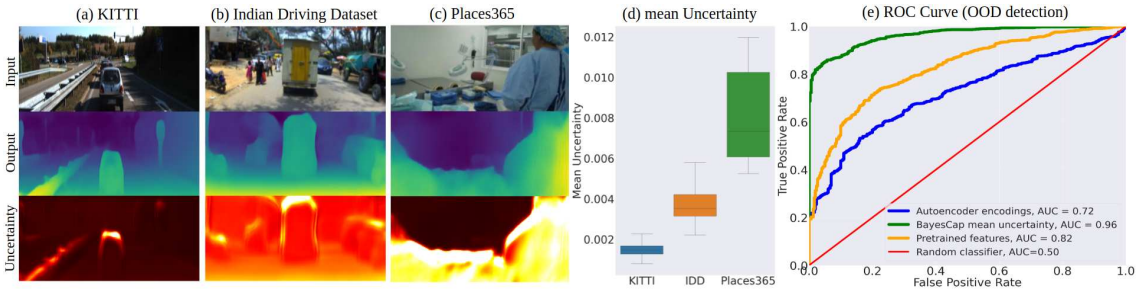

Figure 4.9: `BayesCap` with MonoDepth2 [81] for depth estimation in autonomous driving. Trained on KITTI and evaluated on (a) KITTI, (b) Indian Driving Dataset, and (c) Places365. (d) and (e) Plots show mean uncertainty values and ROC curve for OOD detection respectively, as described in Section 4.5.4.

and autoencoder based approach (AUROC of 0.72). Despite not being specifically tailored for OOD detection, `BayesCap` achieves strong results. This indicates the benefits of its well-calibrated uncertainty estimates on downstream tasks.

4.6 Conclusion

We proposed `BayesCap`, a fast and cheap post-hoc uncertainty estimation method for pretrained deterministic models. We show that our method consistently produces well-calibrated uncertainty estimates across a wide variety of image enhancement and translation tasks without hampering the performance of the pretrained model. This is in sharp contrast to training a Bayesian model from scratch that is more expensive and often not competitive with deterministic counterparts. We demonstrate that derived calibrated uncertainties can be used in critical scenarios for detecting OOD and helping decision-making. One limitation of our model is the assumption that the input is sufficiently contained in the output of the target task. Future works may address this limitation, as well as extending `BayesCap` to discrete predictions.

5

USIM-DAL: Uncertainty-aware Statistical Image Modeling-based Dense Active Learning for Super-resolution

5.1 Abstract

Dense regression is a widely used approach in computer vision for tasks such as image super-resolution, enhancement, depth estimation, etc. However, the high cost of annotation and labeling makes it challenging to achieve accurate results. We propose incorporating active learning into dense regression models to address this problem. Active learning allows models to select the most informative samples for labeling, reducing the overall annotation cost while improving performance. Despite its potential, active learning has not been widely explored in high-dimensional computer vision regression tasks like super-resolution. We address this research gap and propose a new framework called *USIM-DAL* that leverages the statistical properties of colour images to learn informative priors using probabilistic deep neural networks that model the heteroscedastic predictive distribution allowing uncertainty quantification. Moreover, the aleatoric uncertainty from the network serves as a proxy for error that is used for active learning. Our experiments on a wide variety of datasets spanning applications in natural images (visual genome, BSD100), medical imaging (histopathology slides), and remote sensing (satellite images) demonstrate the efficacy of the newly proposed *USIM-DAL* and superiority over several dense regression active learning methods.

5.2 Introduction

The paradigm of dense prediction is very important in computer vision, given that pixel-level regression tasks like super-resolution, restoration, depth estimation etc., help in holistic scene understanding. A common example of a pixel-level (i.e., dense) regression

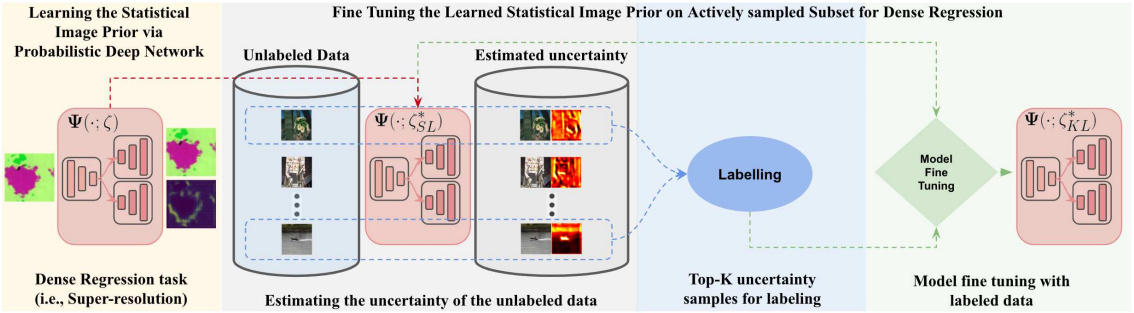

Figure 5.1: The proposed framework *USIM-DAL*. (Left-to-right) We train a probabilistic deep network for a dense regression task (e.g., super-resolution) on synthetic samples obtained from statistical image models as described in Section 6.4. The pre-trained model is used to identify the high-uncertainty samples from the domain-specific unlabeled set. Top-K highly uncertain samples are chosen for labeling on which the pre-trained network is further fine-tuned.

task is *Image super-resolution* (SR) is the process of recovering high-resolution (HR) images from their low-resolution (LR) versions. It is an important class of image processing techniques in computer vision, deep learning, and image processing and offers a wide range of real-world applications, such as medical imaging [151], satellite imaging [273], surveillance [29] and security [82], and remote sensing [304], to name a few. The well-performing techniques for super-resolution often rely on deep learning-based methods that are trained in a supervised fashion, requiring high-resolution data as groundtruth. However, the acquisition of high-resolution imaging data (to be served as labels) for many real-world applications may be infeasible. Consider the example of histopathology microscopy from medical imaging, where the typical digital microscope takes significantly longer to acquire the high-resolution scans (i.e., at high magnification) image of the slide than low-magnification [1, 90]. Moreover, the acquired high-resolution scans also have a significantly larger memory footprint leading to an increase in storage resources [18]. Similarly, acquiring high spatial resolution images from satellites for remote sensing requires expensive sensors and hardware and has significantly higher operating costs [46, 45]. In such scenarios, generating a large volume of training samples is infeasible.

As a remedy, concepts like zero-shot SR or single-image SR have been proposed. Nevertheless, zero-shot SR still requires ample supervision from the test image patches [236] to learn the transferrable model for novel scenarios with divergent distributions [240], and the performance of the single-image SR models is still affected by the lack of sufficient labeled data [153]. Notwithstanding these discussions, there are situations where there are restrictions on dealing with training samples within a pre-defined budget. For example, in histopathology microscopy, the constraint on available resources may allow high-resolution acquisition for only a limited number of patients/microscopy slides. One of the viable solutions in this regard is to select a subset of highly representative training samples from the available training set while respecting the budget and deploying them to train the SR model. This corresponds to the notion of active learning for subset

selection. However, selecting the subset is challenging considering the fact that we need a quantitative measurement for the eligibility of a given training LR-HR pair to be selected. Many works have explored different *query functions* to select a subset to label from a larger dataset [16, 84, 220]. However, most of them have been applied to classification or low-dimensional regression problems [109], and there still exists a gap on how to address this for dense regression tasks (e.g., super-resolution). Active learning technique to label those points for which the current model is least certain has been studied well in the context of classification [309]. While there are recent advances in uncertainty estimation using neural networks for dense regression [121, 267], it is yet to be studied if they can be leveraged in active learning for dense regression.

In summary, our contributions are as follows: (i) We show how statistical image models can help alleviate the need for a large volume of high-resolution imaging data. (ii) We show that probabilistic deep networks, along with the statistical image models, can be used to learn informative prior about niche domain datasets that may allow limited access to high-resolution data. (iii) Our probabilistic deep network trained with the statistical image models allows us to estimate the uncertainty for the sample in a niche domain that can be leveraged for active learning as illustrated in Figure 6.2.

5.3 Related Work

Active Learning. These are a set of techniques that involve selecting a minimal data subset to be annotated, representing the entire dataset, and providing maximum performance gains. Querying strategies for active learning can be broadly categorized into three categories: heterogeneity-based, performance-based, and representativeness-based models. Uncertainty sampling [16, 84, 283, 220, 57], a type of heterogeneity-based model, is a standard active learning strategy where the learner aims to label those samples which have the most uncertain labelings. Non-Bayesian approaches[28, 288] dealing with entropy, distance from decision boundary, etc., also exist but are not scalable for deep learning [229]. Representation-based methods that aim at increasing the diversity in a batch[109] have also been studied. However, most of these works have been studied in the context of classification or low-dimensional regression problems, and the literature on dense regression is still sparse.

Statistical Image models. The $n \times n$ RGB images occupy the space of $\mathbb{R}^{3n^2}$. However, the structured images occupy a small region in that space. The statistical properties of the samples in this small structured space can be leveraged to generate synthetic data that have similar statistics to real-world structured images. For instance, the observation that natural images follow a power law with respect to the magnitude of their Fourier Transform (FT) formed the basis for Wiener image denoising[237], Dead Leaves models [145] and fractals as image models [208, 117]. Similarly, works like [63, 237, 129] showed that outputs of zero mean wavelets to natural images are sparse and follow a generalized Laplacian distribution. Works like [94, 202] showed statistical models capable of producing

realistic-looking textures. The recent work [12] takes this research a step closer to realistic image generation by learning from procedural noise processes and using the generated samples for pre-training the neural networks. However, it is only applied to classification.

Super-resolution. This consists of CNN-based methods to enhance the resolution of the image [144, 285, 260, 259]. Attention mechanism has proven to be ubiquitous, with [296] introducing channel and spatial attention modules for adaptive feature refinement. Transformers-based endeavors such as [152], achieve state-of-the-art results using multi-head self-attention for SR. [223] uses a probabilistic diffusion model and performs SR through an iterative denoising process. Works like [236, 25] use internal and external recurrence of information to get superior SR performance during inference. However, these works do not consider the problem of super-resolution in the active learning context, leaving a gap in the literature.

Uncertainty Estimation. Quantifying uncertainty in machine learning models is crucial for safety-critical applications [182, 246, 263, 265, 261]. Uncertainty can be broadly categorized into two classes: (i) Epistemic uncertainty (i.e., uncertainty in model weights [24, 50, 85, 121]). (ii) Aleatoric uncertainty (i.e., noise inherent in the observations) [10, 278]. The dense predictive uncertainty may be considered as a proxy for error and can be used for active learning purposes [142].

5.4 Method

We first formulate the problem in Section 6.4.1, and present preliminaries on active learning, statistical image models, and uncertainty estimation in Section 5.4.2. In Section 5.4.3, we describe the construction of *USIM-DAL* that learns a prior via statistical image modeling, which is later used to select the most informative samples from the unlabeled set for labeling and further improving the model.

5.4.1 Problem formulation

Let $\mathcal{D}_U = \{\mathbf{x}_i\}_{i=1}^N$ be the unlabeled set of input images from domain $\mathbf{X}$ (i.e., $\mathbf{x}_i \in \mathbf{X} \forall i$). We consider the task where images ($\mathbf{x}$) are to be mapped to another set of dense continuous labels ($\mathbf{y}$, e.g., other images, such that $\mathbf{y}_i \in \mathbf{Y} \forall i$). We want to learn a mapping $\mathbf{\Psi}$ for the same, i.e., $\mathbf{\Psi} : \mathbf{X} \rightarrow \mathbf{Y}$. However, we want to learn it under the constraint that we do not have sufficient *budget* to "label" all the N samples in $\mathcal{D}_U$ (i.e., acquire all the corresponding $\mathbf{y}$), but we do have a budget to label a significantly smaller subset of $\mathcal{D}_U$ with $K << N$ samples, say $\mathcal{D}_U^K$. This is a real-world constraint, as discussed in Section 5.3. In this work, we focus on the problem of super-resolution where the domain $\mathbf{Y}$ consists of high-resolution images (corresponding to the low-resolution images in domain $\mathbf{X}$).

We tackle the problem of choosing the set of $K << N$ samples ($\mathcal{D}_U^K$) that are highly representative of the entire unlabeled training set $\mathcal{D}_U$, such that the learned mapping $\mathbf{\Psi}$ on unseen data from a similar domain performs well.

5.4.2 Preliminaries

Active Learning. As discussed above, given a set of N unlabeled images $\mathcal{D}_U$, we want to choose a set of $K << N$ samples ($\mathcal{D}_U^K$) that are highly representative of the entire unlabeled training set $\mathcal{D}_U$. This is the problem of active learning, which consists of *query strategies* that maps the entire unlabeled set $\mathcal{D}_U$ to its subset. That is, the query strategy (constrained to choose K samples and parameterized by ϕ) is given by, $Q_{K,\phi} : \mathcal{D}_U \rightarrow \mathcal{D}_U^K$. Many works explore designing the query strategy $Q_{K,\phi}$ [16, 84, 283]. However, they seldom attempt to design such a strategy for dense regression.

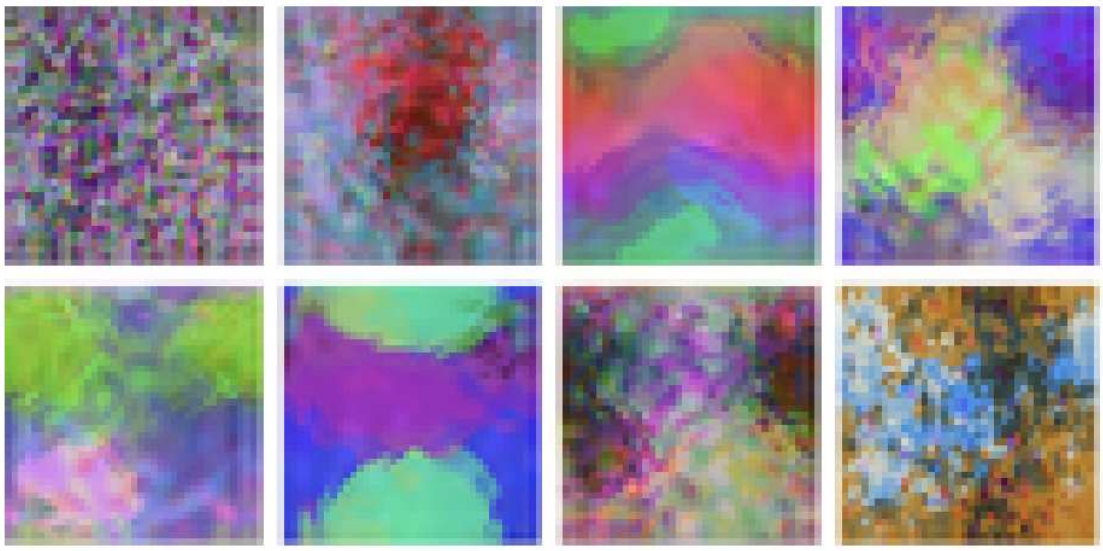

Figure 5.2: Samples generated from Statistical Image Models (combination of Spectrum + WMM + Color histogram). The abstract images generated from such a model capture the Fourier, Wavelet, and color histogram properties of the color natural images.

Statistical Image Models (SIM). As discussed in [12], the statistical properties of RGB images can be exploited to generate synthetic images that can serve as an excellent pre-training learning signal. The generative model (based on statistical properties of RGB images) is described as $\mathcal{G}(\cdot;\theta_G) : \mathbf{z} \rightarrow \mathbf{x}$ where $\mathbf{z}$ is a stochastic latent variable and $\mathbf{x}$ is an image. The image generation is modelled as a hierarchical process in which, first, the parameters of a model are sampled. Then the image is sampled given these parameters and stochastic noise. Previous works [12] highlight the following statistical models. (i) **Spectrum:** based on the magnitude of the Fourier transform (FT). The FT of many natural images follows a power law, i.e., $\frac{1}{|f|^\alpha}$, where $|f|$ is the magnitude of frequency f, and α is a constant close to 1. For generative models, the sampled images are constrained to be random noise images that have FT magnitude following $\frac{1}{|f_x|^a+|f_y|^b}$ with a and b being two random numbers uniformly sampled as detailed in [12]. (ii) **Wavelet-marginal model (WMM):** Generates the texture by modeling their histograms of wavelet coefficient as discussed in [237, 129]. (iii) **Color histograms:** As discussed in [12], this

generative model follows the color distribution of the dead-leaves model [12]. Combining all these different models allows for capturing colour distributions, spectral components, and wavelet distributions that mimic those typical for natural images. Figure 5.2 shows examples of generated samples from such models.

Uncertainty Estimation. Various works [138, 121] have proposed different methods to model the uncertainty estimates in the predictions made by DNNs for different tasks. Interestingly recent works [121, 267] have shown that for many real-world vision applications, modeling the aleatoric uncertainty allows for capturing erroneous predictions that may happen with out-of-distribution samples. To estimate the uncertainty for the regression tasks using deep network (say $\mathbf{\Psi}(\cdot;\zeta) : \mathbf{X} \rightarrow \mathbf{Y}$), the model must capture the output distribution $\mathcal{P}_{Y|X}$. This is often done by estimating $\mathcal{P}_{Y|X}$ with a parametric distribution and learning the parameters of the said distribution using the deep network, which is then used to maximize the likelihood function. That is, for an input $\mathbf{x}_i$, the model produces a set of parameters representing the output given by, $\{\hat{\mathbf{y}}_i, \hat{\nu}_i \dots \hat{\rho}_i\} := \mathbf{\Psi}(\mathbf{x}_i;\zeta)$, that characterizes the distribution $\mathcal{P}_{Y|X}(\mathbf{y}; \{\hat{\mathbf{y}}_i, \hat{\nu}_i \dots \hat{\rho}_i\})$, such that $\mathbf{y}_i \sim \mathcal{P}_{Y|X}(\mathbf{y}; \{\hat{\mathbf{y}}_i, \hat{\nu}_i \dots \hat{\rho}_i\})$. The likelihood $\mathscr{L}(\zeta;\mathcal{D}) := \prod_{i=1}^{N} \mathcal{P}_{Y|X}(\mathbf{y}_i; \{\hat{\mathbf{y}}_i, \hat{\nu}_i \dots \hat{\rho}_i\})$ is then maximized to estimate the optimal parameters of the network. Typically, the parameterized distribution is chosen to be *heteroscedastic* Gaussian distribution, in which case $\mathbf{\Psi}(\cdot;\zeta)$ is designed to predict the *mean* and *variance* of the Gaussian distribution, i.e., $\{\hat{\mathbf{y}}_i, \hat{\sigma}_i^2\} := \mathbf{\Psi}(\mathbf{x}_i;\zeta)$. The optimization problem becomes,

$$\zeta^* = \underset{\zeta}{\operatorname{argmin}} \sum_{i=1}^{N} \frac{|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2}{2\hat{\sigma}_i^2} + \frac{\log(\hat{\sigma}_i^2)}{2} \tag{5.1}$$

With Uncertainty$(\hat{\mathbf{y}}_i) = \hat{\sigma}_i^2$. An important observation from Equation 5.1 is that, ignoring the dependence through ζ, the solution to Equation 5.1 decouples estimation of $\hat{\mathbf{y}}_i$ and $\hat{\sigma}_i$. That is, for minimizing with respect to $\hat{\mathbf{y}}_i$ we need,

$$\frac{\partial \left(\sum_{i=1}^{N} \frac{|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2}{2\hat{\sigma}_i^2} + \frac{\log(\hat{\sigma}_i^2)}{2} \right)}{\partial \hat{\mathbf{y}}_i} = 0 \tag{5.2}$$

$$\frac{\partial^2 \left(\sum_{i=1}^{N} \frac{|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2}{2\hat{\sigma}_i^2} + \frac{\log(\hat{\sigma}_i^2)}{2} \right)}{\partial \hat{\mathbf{y}}_i^2} > 0 \tag{5.3}$$

Equation 5.2 & 5.3 lead to $\hat{\mathbf{y}}_i = \mathbf{y}_i \; \forall i$. Similarly for minimizing with respect to $\hat{\sigma}_i$ we need,

$$\frac{\partial \left(\sum_{i=1}^{N} \frac{|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2}{2\hat{\sigma}_i^2} + \frac{\log(\hat{\sigma}_i^2)}{2} \right)}{\partial \hat{\sigma}_i} = 0 \tag{5.4}$$

$$\frac{\partial^2 \left(\sum_{i=1}^{N} \frac{|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2}{2\hat{\sigma}_i^2} + \frac{\log(\hat{\sigma}_i^2)}{2} \right)}{\partial \hat{\sigma}_i^2} > 0 \tag{5.5}$$

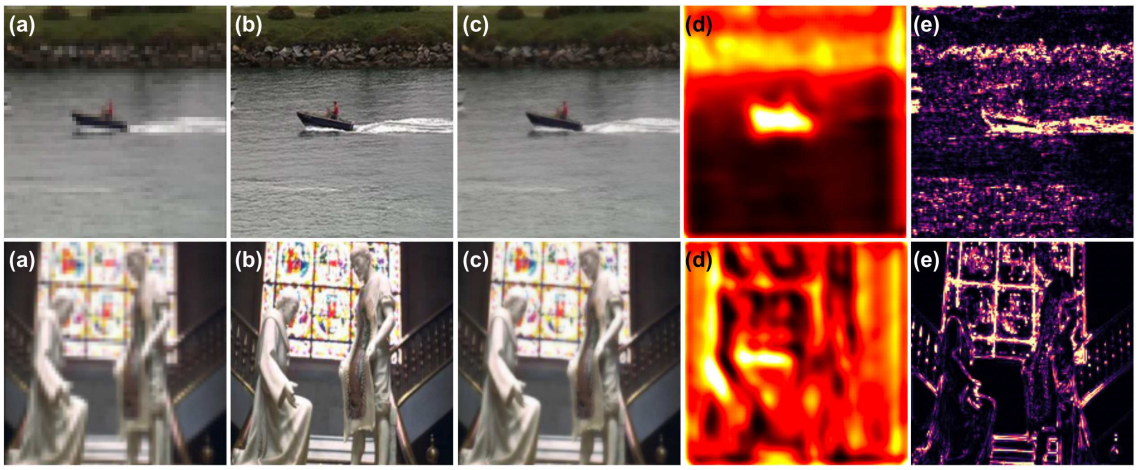

Figure 5.3: Output of the pre-trained probabilistic deep network (which is trained using synthetic images sampled from statistical image models) on samples from *unseen* natural image datasets. (a) LR input, (b) HR groundtruth, (c) Predicted output, SR, from the network, (d) Predicted uncertainty from the network, (e) Error between SR and groundtruth.

Equation 5.4 & 5.5 lead to $\hat{\sigma}_i^2 = |\hat{\mathbf{y}}_i - \mathbf{y}_i|^2 \; \forall i$. That is, the estimation $\hat{\sigma}_i^2$ should perfectly reflect the squared error. Therefore, a higher $\hat{\sigma}_i^2$ indicates higher error. We leverage this observation to design our dense active learning framework as described in Section 5.4.3.

5.4.3 Constructing *USIM-DAL*

To tackle the problem mentioned in Section 6.4.1 (i.e., choosing a small subset), we leverage the fact that even before training the model with the labelled set, we can train a model based on the samples that we get from statistical image model as described above, which can then be used to make inference on the unlabeled domain-specific dataset identifying the high-uncertainty samples. The high-uncertainty samples can then be labelled and used to fine-tune the model.

We constraint the generative process for statistical image models as, Similar to [12], we treat image generation as a hierarchical process in which first the parameters of a model, θ_G, are sampled. Then the image is sampled given these parameters and stochastic noise, i.e.,

$$\theta_G \sim prior(\theta_G) \text{ and } \mathbf{z} \sim prior(\mathbf{z}) \tag{5.6}$$

$$\mathbf{x} = \mathcal{G}(\mathbf{z}; \theta_G) \tag{5.7}$$

In particular, for super-resolution, we create a large (synthetic) labelled dataset using the samples from the statistical image models, say $\mathcal{D}_{SL} = \{(\texttt{low}(\mathbf{x}_{s,i}), \mathbf{x}_{s,i})\}_{i=1}^{M}$. Where $\mathbf{x}_{s,i}$ are generated samples from statistical image model and $\texttt{low}(\cdot)$, is the 4× down-sampling operation. We then train the network $\mathbf{\Psi}(\cdot; \zeta)$ on $\mathcal{D}_{SL}$ using Equation 5.1, leading to the optimal parameter ζ_{SL}^*, as shown in Figure 6.2. The trained model $\mathbf{\Psi}(\cdot; \zeta_{SL}^*)$ is then run in inference mode on all the samples of the unlabeled set $\mathcal{D}_U$ and gather the top uncertain

samples for labeling, that is,

$$\{\hat{\mathbf{y}}_i, \hat{\sigma}_i\} := \mathbf{\Psi}(\mathbf{x}_i; \zeta^*_{SL}) \; \forall \mathbf{x}_i \in \mathcal{D}_U \tag{5.8}$$

$$\mathcal{D}^K_U := \{\mathbf{x}_j\} \forall j \in \texttt{topK}\left(\{\langle\hat{\sigma}_i\rangle\}^N_{i=1}\right) \tag{5.9}$$

Where, $\langle\cdot\rangle$ represents the mean operation, and $\texttt{topK}\left(\{\langle\hat{\sigma}_i\rangle\}^N_{i=1}\right)$ returns the indices of "top-K" most uncertain samples (i.e., mean uncertainty is high). We then acquire the labels for the samples in $\mathcal{D}^K_U$, giving us, $\mathcal{D}^K_{UL} = \{(\mathbf{x}_j, \mathbf{y}_j)\}$. As discussed in Section 5.4.2, the input samples in $\mathbf{D}^K_{UL}$ serve as a proxy to the set of K samples that would have the highest error between the prediction made by the model $\mathbf{\Psi}(\cdot; \zeta^*_{SL})$ and the ground truth. That leads to better fine-tuning. The model $\mathbf{\Psi}(\cdot; \zeta^*_{SL})$ is then fine-tuned on $\mathcal{D}^K_{UL}$ via Equation 5.1, leading to the final state of the model $\mathbf{\Psi}(\cdot; \zeta^*_{KL})$ (shown in Figure 6.2) that can be used for inferring on the new sample.

USIM-DAL models the aleatoric uncertainties in the prediction. Still, it is crucial to note that it leverages the Statistical Image Modeling (SIM)-based synthetic images for pertaining and learning important priors for color images that broadly capture different niche domains such as medical images, satellite images, etc. Therefore, the initial model, capable of estimating the aleatoric uncertainty (trained on SIM-based synthetic images), can reasonably capture the uncertainty as a proxy for reconstruction error for domain-specific images that are not necessarily out-of-distribution images. Moreover, picking samples with high reconstruction errors for subsequent fine-tuning of the model yields better performance on similar highly erroneous cases, iteratively improving the model. Furthermore, in high-dimensional regression cases, the aleatoric and epistemic uncertainty often influence each other and are not independent [121, 267, 327].

5.5 Experiments and Results

We provide an overview of the experiments performed and the results obtained. In Section 5.5.1, we describe the task and various methods used for comparison. Section 5.5.3 analyzes the performance of various dense active learning algorithms for super-resolution and shows that our proposed method *USIM-DAL* can help greatly improve the performance when constrained with a limited budget.

5.5.1 Tasks, Datasets, and Methods

We present the results of all our experiments on the super-resolution task. We demonstrate our proposed framework using a probabilistic SRGAN (which is the adaptation of SR-GAN [144] that estimates pixel-wise uncertainty as described in [121]) model. We evaluate the performance of various models on a wide variety of domains like (i) Natural Images (with Set5, Set14, BSD100, and Visual Genome dataset [144, 166, 130]). (ii) Satellite Images (with PatternNet dataset [337]). (ii) Histopathology Medical Images (with Camelyon dataset [157]). The evaluation protocol is designed to constraint all the training domain

datasets to be restricted by a small fixed number of images (also called *training budget*). We used different training budgets of 500, 1000, 2000, 3000 and 5000 images for natural and satellite domains. For both natural and satellite images, the input image resolution was set to 64×64. For natural images the training dataset was obtained from Visual Genome (separate from the test-set). Similarly, for the histopathology medical images, the input image resolution was set to 32×32 and we used training budgets of 4000, 8000, 12000, and 16000.

We compare the super-resolution performance in terms of metrics MSE, MAE, PSNR, and SSIM [289] for the following methods on respective test sets: (i) SRGAN model trained from scratch with a randomly chosen subset satisfying the training budget from the entire training data (called *Random*). (ii) SRGAN model trained from scratch on a large synthetically generated dataset via statistical image modeling (as described in Section 5.4.2). This model is called *SIM*. (iii) SRGAN model trained from scratch on a large synthetically generated dataset via statistical image modeling and then fine-tuned on a randomly chosen subset satisfying the training budget from the entire training data, called *SIM+Random*. (iv) SRGAN model trained from scratch on a large synthetically generated dataset via statistical image modeling and then fine-tuned on a subset chosen using uncertainty estimates, satisfying the training budget from the entire training data, called *USIM-DAL*.

5.5.2 Dense Active Learning via Uncertainty Estimation

Our method proposes to utilize a probabilistic network that is learned from synthetic images sampled from statistical image models (i.e., $\mathbf{\Psi}(\cdot; \zeta^*_{SL})$ mentioned in Section 5.4.3). Figure 5.3 shows the output of probabilistic SRGAN trained on synthetic images evaluated on samples from natural images. We observe that (i) The predicted super-resolved images (Figure 5.3-(c)) are still reasonable. (ii) The uncertainty estimates (Figure 5.3-(d)) still resemble the structures from the images and are a reasonable proxy to the error maps (Figure 5.3-(e)) between the predictions and the ground truth, even though the model has never seen the natural images.

We use the predicted uncertainty from this model to identify the samples from the real-world domain that would lead to

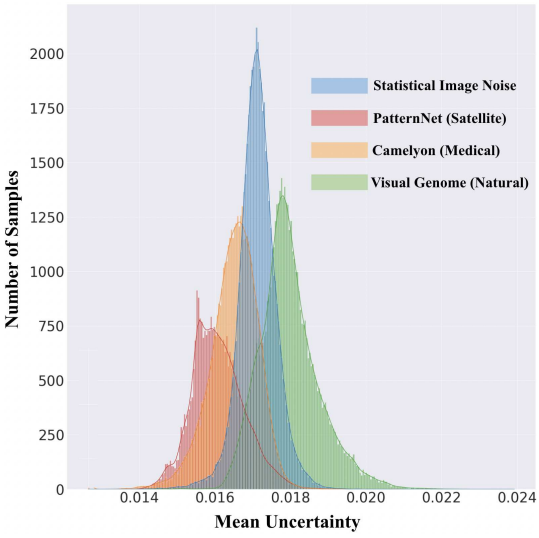

Figure 5.4: Distribution of mean uncertainty for samples in Statistical Image Noise, PatternNet (satellite), Camelyon (medical), Visual Genome (natural) datasets.

D	Methods	Budgets (Number of images)					
		500	1000	2000	3000	5000	
		MSE / MAE / PSNR / SSIM	MSE / MAE / PSNR / SSIM	MSE / MAE / PSNR / SSIM	MSE / MAE / PSNR / SSIM	MSE / MAE / PSNR / SSIM	
		$\times 10^3$ $\times 10^2$ $\times 10^0$ $\times 10^2$	$\times 10^3$ $\times 10^2$ $\times 10^0$ $\times 10^2$	$\times 10^3$ $\times 10^2$ $\times 10^0$ $\times 10^2$	$\times 10^3$ $\times 10^2$ $\times 10^0$ $\times 10^2$	$\times 10^3$ $\times 10^2$ $\times 10^0$ $\times 10^2$	
Set5	Random	4.129 / 3.854 / 24.784 / 7.232	3.898 / 3.720 / 24.957 / 7.319	3.660 / 3.588 / 25.271 / 7.422	3.586 / 3.529 / 25.334 / 7.465	3.500 / 3.420 / 25.514 / 7.539	
	SIM	3.431 / 3.524 / 25.641 / 7.541	3.431 / 3.524 / 25.641 / 7.541	3.431 / 3.524 / 25.641 / 7.541	3.431 / 3.524 / 25.641 / 7.541	3.431 / 3.524 / 25.641 / 7.541	
	SIM + Random	2.976 / 3.139 / 26.283 / 7.839	2.958 / 3.099 / 26.377 / 7.872	2.941 / 3.081 / 26.435 / 7.896	2.934 / 3.088 / 26.436 / 7.910	2.912 / 3.056 / 26.546 / 7.935	
	USIM-DAL	**2.926** / **3.088** / **26.484** / **7.869**	**2.884** / **3.069** / **26.550** / **7.894**	**2.848** / **3.027** / **26.619** / **7.931**	**2.843** / **3.029** / **26.644** / **7.944**	**2.831** / **3.025** / **26.699** / **7.943**	
Set14	Random	6.254 / 4.750 / 22.535 / 6.333	6.111 / 4.669 / 22.576 / 6.382	5.942 / 4.564 / 22.701 / 6.468	5.862 / 4.539 / 22.616 / 6.488	5.800 / 4.450 / 22.886 / 5.594	
	SIM	4.852 / 4.303 / 22.897 / 6.383	4.852 / 4.303 / 22.897 / 6.383	4.852 / 4.303 / 22.897 / 6.383	4.852 / 4.303 / 22.897 / 6.383	4.852 / 4.303 / 22.897 / 6.383	
	SIM + Random	4.488 / 3.907 / 23.748 / **7.016**	4.485 / 3.871 / 23.787 / **7.082**	4.444 / 3.828 / 24.106 / 7.159	4.426 / 3.828 / 24.162 / 7.179	4.396 / 3.798 / 24.090 / 7.198	
	USIM-DAL	**4.376** / **3.836** / **23.810** / 6.984	**4.366** / **3.816** / **23.818** / 7.000	**4.331** / **3.767** / **24.288** / **7.177**	**4.317** / **3.749** / **24.422** / **7.208**	**4.292** / **3.728** / **24.553** / **7.227**	
BSD100	Random	4.857 / 4.338 / 23.357 / 6.072	4.778 / 4.294 / 23.427 / 6.098	4.670 / 4.226 / 23.583 / 6.160	4.630 / 4.207 / 23.598 / 6.187	4.600 / 4.160 / 23.703 / 6.214	
	SIM	3.526 / 3.738 / 24.805 / 6.713	3.526 / 3.738 / 24.805 / 6.713	3.526 / 3.738 / 24.805 / 6.713	3.526 / 3.738 / 24.805 / 6.713	3.526 / 3.738 / 24.805 / 6.713	
	SIM + Random	3.362 / 3.578 / 25.007 / 6.786	3.352 / 3.559 / 25.043 / 6.794	3.328 / 3.539 / 25.092 / 6.812	3.323 / 3.540 / 25.085 / 6.816	3.305 / 3.519 / 25.137 / 6.834	
	USIM-DAL	**3.299** / **3.520** / **25.174** / **6.826**	**3.293** / **3.520** / **25.191** / **6.830**	**3.282** / **3.504** / **25.207** / **6.838**	**3.277** / **3.496** / **25.212** / **6.844**	**3.262** / **3.486** / **25.263** / **6.854**	
Visual Genome	Random	4.442 / 3.946 / 23.935 / 6.853	4.346 / 3.892 / 24.033 / 6.889	4.231 / 3.818 / 24.200 / 6.954	4.182 / 3.797 / 24.216 / 6.983	4.120 / 3.718 / 24.353 / 7.032	
	SIM	4.310 / 3.963 / 24.055 / 6.826	4.310 / 3.963 / 24.055 / 6.826	4.310 / 3.963 / 24.055 / 6.826	4.310 / 3.963 / 24.055 / 6.826	4.310 / 3.963 / 24.055 / 6.826	
	SIM + Random	4.038 / 3.721 / 24.396 / 7.036	4.026 / 3.690 / 24.423 / 7.056	3.993 / 3.663 / 24.496 / 7.088	3.977 / 3.661 / 24.515 / 7.101	3.943 / 3.631 / 24.563 / 7.126	
	USIM-DAL	**3.966** / **3.668** / **24.543** / **7.056**	**3.949** / **3.657** / **24.570** / **7.069**	**3.925** / **3.623** / **24.624** / **7.109**	**3.908** / **3.608** / **24.656** / **7.126**	**3.880** / **3.593** / **24.721** / **7.143**	

Table 5.1: Evaluating different methods on natural image datasets (Set5, Set14, BSD100, Visual Genome) using MSE, MAE, PSNR, SSIM. Lower MSE/MAE is better. Higher PSNR/SSIM is better. "D": Datasets. Best results are in bold.

high errors. Figure 5.4 shows the distribution of mean uncertainty values for samples in (i) Statistical Noise (ii) Natural (ii) Satellite (iii) Medical image datasets. We notice that the model trained on synthetic images leads to a gaussian distribution for the mean uncertainty values on the synthetic image datasets. We obtain similar distributions for other datasets from different domains. This further emphasizes that uncertainty estimates obtained from $\mathbf{\Psi}(\cdot; \zeta^*_{SL})$ can be used as a proxy to identify the highly uncertain (therefore erroneous) samples from different domains (i.e., the samples close to the right tail of the distributions).

5.5.3 *USIM-DAL* for Super-resolution

Table 5.1 shows the performance of different methods on multiple natural image datasets, including Set5, Set14, BSD100, and Visual Genome (VG). We observe that with the smallest training budget of 500 images, *USIM-DAL* performs the best with a PSNR/-MAE of 25.174/0.035 (Table 5.1 shows the results with a scaling factor for better accommodation) compared to *SIM+Random* with PSNR/MAE of 25/0.039 and *SIM* with

PSNR/MAE of 24.8/0.037. We also notice that at this budget, choosing the random subset of the training dataset to train the model from scratch performs the worst with PSNR/MAE of 23.36/0.043. As the budget increases (left to right in Tabel 5.1), the performances of all the methods also improve. However, a similar trend is observed where the *USIM-DAL* performs better than *SIM+Random*, *SIM*, and *Random*. We observe a similar trend for other natural image datasets. This allows us to make the following observations: (i) Using a synthetic training image dataset (sampled from the statistical image model, discussed in Section 5.4.2) leads to better performance than using a small random subset of training images from the original domain (i.e., *SIM* better than *Random*). (ii) Using the above synthetic training image dataset to train a model and later fine-tuning it with domain-specific samples lead to further improvements (i.e., both *USIM-DAL* and *SIM+Random* better than *SIM*). (iii) With a limited budget, fine-tuning a model (pre-trained on synthetic training image dataset) using high-uncertainty samples from the training set (as decided by the *USIM-DAL*) is better than using the random samples from the training set (i.e., *USIM-DAL* better than *SIM+Random*). We perform a similar set of experiments with other imaging domains, namely, (i) Satellite imaging (using PatternNet dataset) and (ii) Medical imaging (using Camelyon histopathology dataset). We observe a similar (to natural images) trend in these domains. Figure 5.5 shows the performance (measured using PSNR) for different methods on these two domains, with varying training budgets. For satellite imaging, at the lowest training budget of 500 images, *USIM-DAL* with PSNR of 23.5 performs better than *SIM+Random* with PSNR of 23.4 and *SIM* with a PSNR of 23.2. We observe that as the training budget increases to 2000 images, *USIM-DAL* (with PSNR of 23.6) outperforms *SIM+Random* (with PSNR of 23.35) with an even higher margin. As we increase the training budget further, the *SIM+Random* model starts performing similarly to *USIM-DAL*. With a budget of 5000 samples, *USIM-DAL* has a performance of 23.62, and *SIM+Random* has a performance of 23.60. Given a domain with large (specific to datasets) training budgets, the performance achieved from random sampling and active learning strategies will converge.

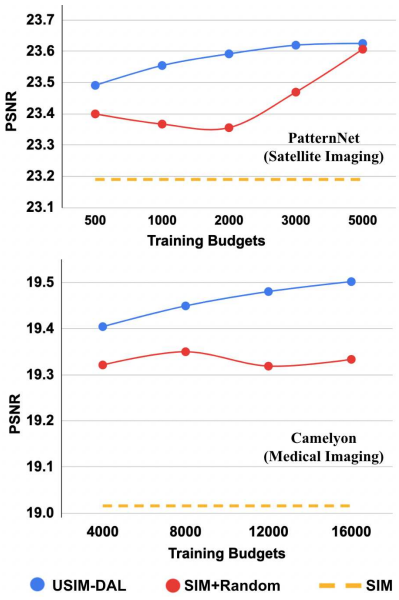

Figure 5.5: Evaluation of various methods on histopathology medical domain (i.e., Camelyon dataset) and satellite imaging domain (i.e., PatternNet dataset) at various fine-tuning budgets. The yellow curve is the *SIM* baseline. The red curve is the SIM model fine-tuned with random samples (i.e., *SIM+Random*). The blue curve is the SIM model fine-tuned with the highest uncertain samples (i.e., *USIM-DAL*).

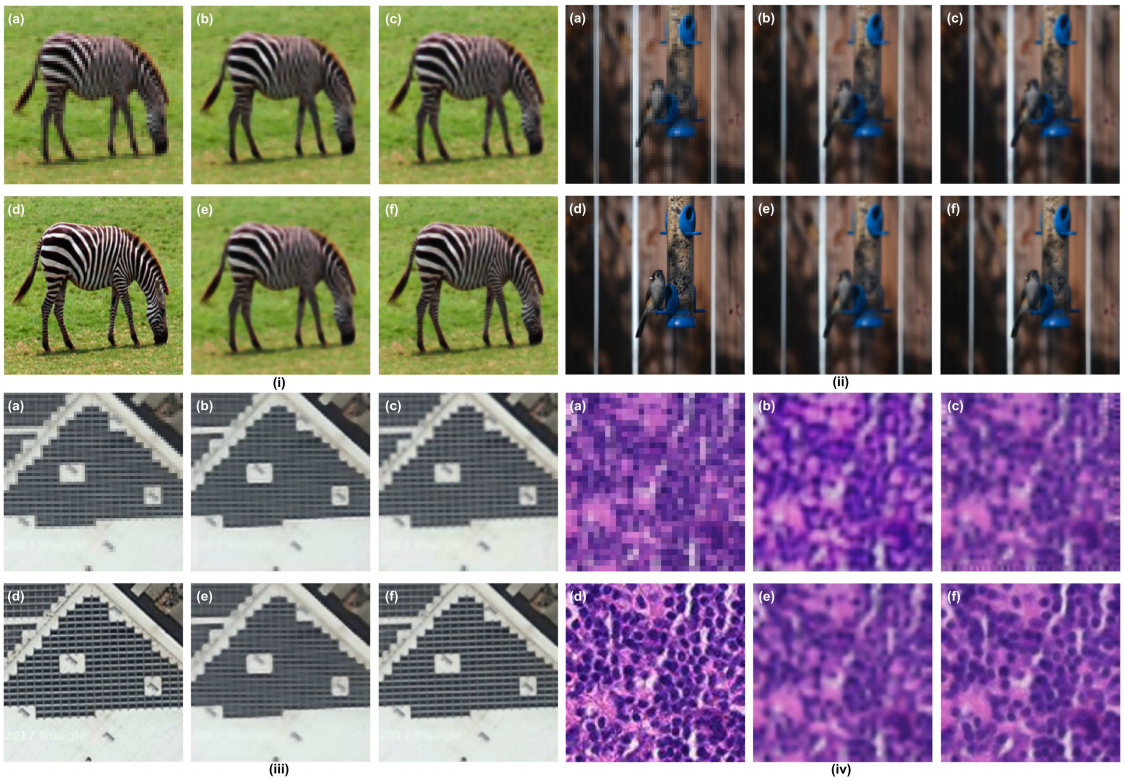

Figure 5.7: Qualitative results from different methods (performing 4× super-resolution) including (b) *Random*, (c) *SIM*, (e) *SIM+Random*, (f) *USIM-DAL* on (i) BSD100, (ii) Visual Genome, (iii) PatternNet, and (iv) Camelyon datasets. (a) LR input, and (d) HR groundtruth. Input resolution for BSD100, Visual Genome, and PatternNet is 64×64, and for Camelyon is 32×32. (f) *USIM-DAL* produces the most visually appealing outputs.

For Camelyon dataset, we use the input image resolution of 32×32. We observe that *USIM-DAL* performs the best across all budgets when compared to *SIM+Random* and *SIM*. We also note that high-frequency features that are typically present in high-resolution scans (i.e., obtained at 20× or 40× magnification from the histopathology microscope) make the super-resolution problem harder and require more data to achieve good performance. Figure 5.6 summarizes the performance gain (in terms of PSNR) by using *USIM-DAL* (i.e., uncertainty-based active learning strategy for dense regression) compared to *SIM+Random* (i.e., no active learning, randomly choosing a subset from real training domain), relative to *SIM* (i.e., no real samples used from the domain) at best performing limited budgets. That is, the relative percentage boost in performance is

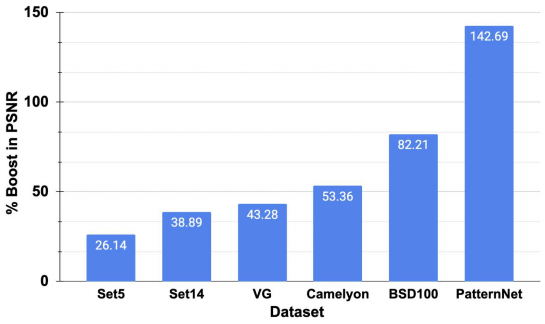

Figure 5.6: Relative % boost in PSNR of *USIM-DAL* relative to *SIM+Random* over *SIM* baseline (Equation 5.10) at optimal budget for six datasets across three domains.

reported as:

$$\frac{(\text{PSNR}_{\text{USIM-DAL}} - \text{PSNR}_{\text{SIM+Random}}) * 100}{\text{PSNR}_{\text{SIM+Random}} - \text{PSNR}_{\text{SIM}}} \tag{5.10}$$

We note that *USIM-DAL* consistently performs better than *SIM+Random*, with the relative percentage boost in PSNR of 26.14% for Set5 to 142.69% for PatternNet. Figure 5.7 shows the qualitative outputs of different models on multiple datasets. On all the datasets, we notice that the output obtained by *USIM-DAL* is better than the output of *SIM+Random* that is better than *SIM* and *Random*.

5.6 Conclusion

In this work, we presented a novel framework called *USIM-DAL* that is designed to perform active learning for dense-regression tasks, such as image super-resolution. Dense-regression tasks, such as super-resolution, are an important class of problem for which deep learning offers a wide range of solutions applicable to medical imaging, security, and remote sensing. However, most of these solutions often rely on supervision signals derived from high-resolution images. Due to the time-consuming acquisition of high-resolution images or expensive sensors, hardware, and operational costs involved, it is not always feasible to generate large volumes of high-resolution imaging data. But in real-world scenarios, a limited budget for acquiring high-resolution data is often available. This calls for active learning that chooses a subset from large unlabeled set to perform labeling to train the models. While multiple querying strategies (in the context of active learning) exist for the classification tasks, the same for dense regression tasks are seldom discussed. Our work paves the way for using modern uncertainty estimation techniques for active learning in dense regression tasks. We show that a large synthetic dataset acquired using statistical image models can be used to learn informative priors for various domains, including natural images, medical images, satellite images, and more. The learned prior can then be used to choose the subset consisting of high-uncertainty samples that can then be labeled and used to fine-tune the prior further. Through extensive experimentation, we show that our approach generalizes well to a wide variety of domains, including medical and satellite imaging. we show that active learning performed by proposed querying strategy (i.e., *USIM-DAL*) leads to gains of upto 140% / 53% with respect to a random selection strategy (i.e., SIM+Random) relative to no dataset-specific fine-tuning (i.e., *SIM*) on satellite/medical imaging.

6

ProbVLM: Probabilistic Adapter for Frozen Vision-Language Models

6.1 Abstract

Large-scale vision-language models (VLMs) like CLIP successfully find correspondences between images and text. Through the standard deterministic mapping process, an image or a text sample is mapped to a single vector in the embedding space. This is problematic: as multiple samples (images or text) can abstract the same concept in the physical world, deterministic embeddings do not reflect the inherent ambiguity in the embedding space. We propose `ProbVLM`, a probabilistic adapter that estimates probability distributions for the embeddings of pre-trained VLMs via inter/intra-modal alignment in a post-hoc manner without needing large-scale datasets or computing. On four challenging datasets, i.e., COCO, Flickr, CUB, and Oxford-flowers, we estimate the multi-modal embedding uncertainties for two VLMs, i.e., CLIP and BLIP, quantify the calibration of embedding uncertainties in retrieval tasks and show that `ProbVLM` outperforms other methods. Furthermore, we propose active learning and model selection as two real-world downstream tasks for VLMs and show that the estimated uncertainty aids both tasks. Lastly, we present a novel technique for visualizing the embedding distributions using a large-scale pre-trained latent diffusion model.

6.2 Introduction

Recently, large vision-language models (VLMs) [205, 177, 150, 239, 4, 112] have become exceedingly popular due to their ability to align images and text. These models such as CLIP [205] and BLIP [150] are trained on large-scale datasets such as LAION-400M [225] and YFCC-100M [252] and have shown strong performance when evaluated in a zero-shot fashion (i.e without requiring fine-tuning on specific datasets) for a variety of downstream tasks. One of the most popular applications of VLMs is cross-modal retrieval [276, 282] i.e retrieving images (text) for a queried text (images). However, image-to-text matching (and vice-versa) is fundamentally ill-posed due to the inherent ambiguity in

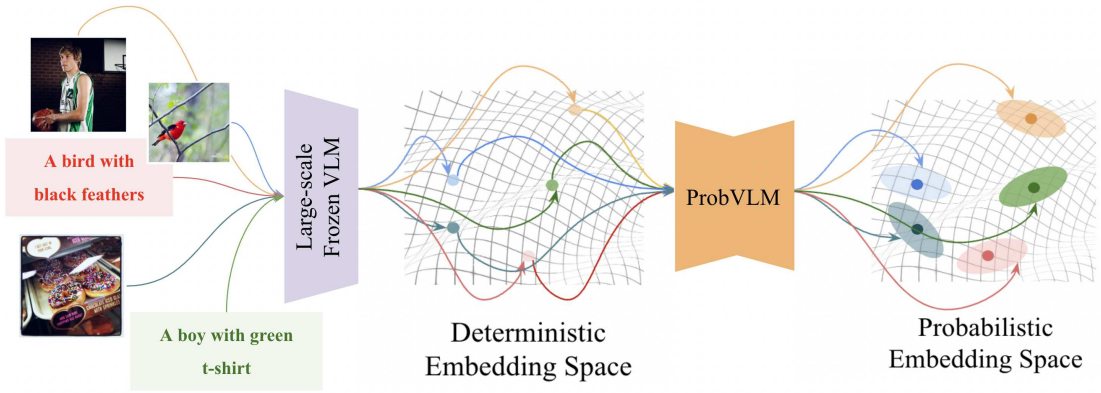

Figure 6.1: We provide probabilistic embeddings for deterministic pre-trained vision-language models that are *frozen*. By capturing the ambiguity inherently present in the inputs, we obtain well-calibrated uncertainty estimates.

either modality [305], i.e. the same caption (or image) can be valid for multiple images (or captions). Therefore, it becomes essential to model the ambiguity inherently present in the various modalities, and combinations thereof.

Instead of mapping inputs to embeddings, probabilistic embedding methods [190, 37] learn to map input samples to distributions. This is achieved by parameterizing the distributions of the embeddings and training a deep neural network to maximize its likelihood. Although they model ambiguities in the embedding space, such probabilistic models require training deep networks from scratch. This requires access to the large-scale datasets and the computational resources of the recent VLMs [205, 112, 177, 239, 150].

We propose `ProbVLM`, a post-hoc probabilistic adapter, the first method to convert the deterministic embeddings provided by a *frozen* large-scale vision-language models into probabilistic ones, as shown in Figure 6.1. This enables us to efficiently retain the benefits of large-scale pre-training while learning distributions that model the inherent ambiguities in the different modalities. Our `ProbVLM` models the embedding distribution as a heteroscedastic probability distribution and is trained using a combination of intra-modal and cross-modal alignment objectives and provides well-calibrated uncertainty estimates, useful for several tasks.

We demonstrate on two large vision-language datasets, i.e., COCO [154] and Flickr [201], and on two fine-grained image datasets, i.e., CUB [275] and Oxford-Flowers [187] with sentences from [209], that `ProbVLM` learns calibrated uncertainties without requiring large-scale models to be trained from scratch. This sharply contrasts previous works on probabilistic embeddings [190, 37] that train new models from scratch. We perform a series of analyses to understand the impact of the training objective and to study the properties of the resulting uncertainties. Furthermore, we demonstrate that our uncertainty estimates can be used to select the optimal model from a set of finetuned vision-language models on an unlabeled target dataset. They can also be used to choose the most suitable samples for fine-tuning the model in an active learning setup. Finally, with the help of a pretrained

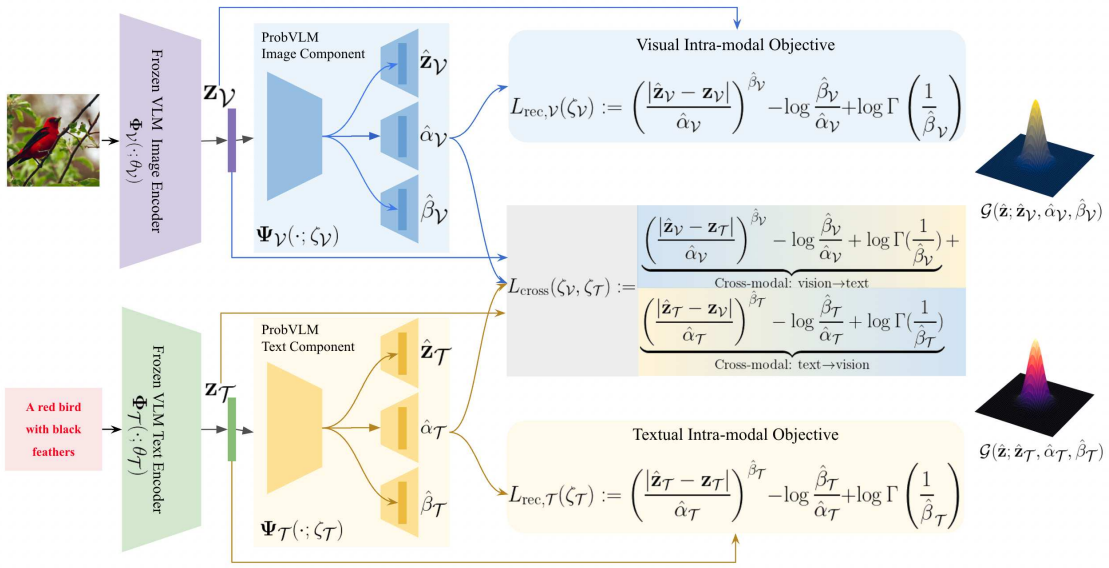

Figure 6.2: Proposed framework (`ProbVLM`) takes an existing vision-language model and introduces a probabilistic adapter over the image and text encoders. These adapters predict the parameters of a parameterized distribution for a given embedding. Models are trained by minimizing an objective consisting of intra/cross-modal supervision as detailed in Section 6.4.

latent diffusion model [216], i.e., *Stable Diffusion*, we decode sampled embeddings from predicted distribution to visualize the predicted embedding distributions. We show that the predicted embedding distributions indeed capture meaningful modes of variation, that may be interpretable.

6.3 Related Work

Vision-Language Models. Such models [205, 177, 239, 4, 150, 162, 149, 310, 315, 290] have become ubiquitous in recent times due to their various applications in image classification [333, 71, 334, 175], cross-modal retrieval [11], as well as open-vocabulary semantic segmentation [78, 301]. The most notable among these is CLIP [205], which consists of an image and text encoder trained on 400M image-text pairs with a contrastive objective [89, 191]. As a result, the model is able to project images and text to a shared embedding space. In this paper, we focus on using the shared embedding space for the task of cross-modal retrieval [201, 154]. Recent works have predominantly relied on large-scale pre-training [205, 177, 239, 4, 325, 225, 226] to project images and text to the same metric space. However, it is essential to note that all of these vision-language models [205, 177, 150, 239, 4] provide deterministic mappings that do not model the inherent ambiguity in the inputs. In this work, we turn a deterministic model (i.e., CLIP) into a probabilistic one, without the need of a large-scale dataset.

Probabilistic Embeddings. These methods [190, 37, 148] provide an elegant solution to estimate the ambiguity present in the inputs [124]. The key idea here is to map inputs to

probability distributions in the embedding space, as opposed to point estimates, thereby modeling the inherent ambiguity present in the input. In the context of cross-modal retrieval, this was done by optimizing a probabilistic analog of the contrastive objective to learn distributions for the image and text inputs [37]. Other works have further improved the performance [148, 194, 111], extended this formulation to achieve compositional retrieval [185], and have applied it to other tasks such as video retrieval [194, 61] and tasks like pose estimation [247]. However, most of these works focus on training a model from scratch, thereby not leveraging the power of the pre-trained models that are widely present. The notable exception to this is Probabilistic Face Embedding (PFE) [234] that proposed to learn a probabilistic embedding while retaining a deterministic pre-trained model for the task of learning face embeddings. However, this was done in a unimodal setting using only images. In this work, we aim to utilize pre-trained vision-language models while providing probabilistic embeddings for both modalities. The probabilistic embeddings derived from our proposed `ProbVLM` are consistent with cross-modal learning at the core of pretrained vision-language models.

Uncertainty Estimation. These techniques have been widely explored for different tasks in computer vision [120, 24, 139, 142, 188, 318, 268, 184, 256, 88, 221, 319, 207, 261, 246, 263]. Uncertainties can be broadly categorized into aleatoric [120, 74, 10, 284, 48, 9, 278, 188, 299] and epistemic [85, 24, 139, 291, 70, 108, 64, 65] uncertainties. Uncertainty estimation has been used for a variety of tasks, such as identifying model failure [58, 20, 19, 292] and is extensively used in active learning to select the best samples to train the model [230, 125, 206, 232, 309, 308, 203, 179]. While several of these methods focus on training a new Bayesian model from scratch for quantifying the uncertainties in the prediction, some recent works like [268, 318, 96] have proposed methods to estimate the uncertainties for the pre-trained frozen models. However, these works tackle data from a single modality. This work efficiently estimates the uncertainty for the pre-trained frozen large-scale vision-language model.

6.4 Method

We first describe the problem formulation in Section 6.4.1. In Section 6.4.2, we describe our proposed method `ProbVLM` that estimates the complex probability distributions for the embeddings of the frozen deterministic vision-langue encoders, quantifying the uncertainties for their predictions.

6.4.1 Problem Formulation

Let $\mathcal{D} = (\mathcal{I}, \mathcal{C})$ denote a vision and language dataset, where $\mathcal{I}$ is a set of images and $\mathcal{C}$ a set of captions The two sets are connected via ground-truth matches where multiplicity is plausible. For a caption $c \in \mathcal{C}$ (respectively an image $i \in \mathcal{I}$), the set of corresponding images (respectively captions) is given by $\kappa(c) \subseteq \mathcal{I}$ (respectively $\kappa(i) \subseteq \mathcal{C}$). Recent

advances in cross-modal vision-language models [205, 177, 239] often involve learning a shared embedding space, $\mathcal{Z} \subseteq \mathbb{R}^D$ (D-dimensional space), for images and texts. This allows quantifying the similarity between cross-modal elements based on their distances in the shared embedding space. The shared embedding space is learned via a set of two encoders: $\mathbf{\Phi}_{\mathcal{V}}(\cdot;\theta_{\mathcal{V}}) : \mathcal{I} \rightarrow \mathcal{Z}$ for the images and $\mathbf{\Phi}_{\mathcal{T}}(\cdot;\theta_{\mathcal{T}}) : C \rightarrow \mathcal{Z}$ for the texts, where $\theta_{\mathcal{V}}$ and $\theta_{\mathcal{T}}$ are the parameters for the respective mapping functions.

We consider a real-world scenario where the above set of encoders have already been trained on vast datasets using large models with high computational cost, e.g., CLIP [205], SLIP [177], Flava [239] and BLIP [150], are in *frozen state*, i.e., we have $\mathbf{\Phi}_{\mathcal{V}}(\cdot;\theta^*_{\mathcal{V}})$ and $\mathbf{\Phi}_{\mathcal{T}}(\cdot;\theta^*_{\mathcal{T}})$, where $\theta^*_{\mathcal{V}}, \theta^*_{\mathcal{T}}$ represents the parameters of the pretrained frozen encoders. These encoders are *deterministic* and map an image/text to vectors in the shared space, i.e., given a sample image $\mathbf{x}_{\mathcal{V}}$ (and similarly sample text $\mathbf{x}_{\mathcal{T}}$), the encoder provides an embedding $\mathbf{z}_{\mathcal{V}} := \mathbf{\Phi}_{\mathcal{V}}(\mathbf{x}_{\mathcal{V}};\theta^*_{\mathcal{V}})$ (and similarly, $\mathbf{z}_{\mathcal{T}} := \mathbf{\Phi}_{\mathcal{T}}(\mathbf{x}_{\mathcal{T}};\theta^*_{\mathcal{T}})$). However, the point estimates, $\mathbf{z}$, do not capture the ambiguity inherent to these embeddings [190, 37, 61] that are better represented by the probability distribution $P_{\mathbf{z}|\mathbf{x}}$. Therefore, we propose to estimate $P_{\mathbf{z}|\mathbf{x}}$ for the pretrained model efficiently, using `ProbVLM`, quantifying the uncertainties of the output without re-training the encoders.

6.4.2 Building `ProbVLM`

Despite being deterministic, large-scale *frozen* encoders already provide high-quality point estimates. Our proposed method leverages this fact, using the embeddings $\mathbf{z}$ as estimates for the mean of the desired distribution $P_{\mathbf{z}|\mathbf{x}}$, and estimating the remaining parameters. $P_{\mathbf{z}|\mathbf{x}}$ can be modeled as a parametric distribution $P_{\mathbf{z}|\mathbf{x}}(\mathbf{z}|\{\hat{\mathbf{z}}, \hat{v}...\hat{\rho}\})$ where the parameters can be estimated using a deep neural network [70, 120, 139]. Therefore, we introduce `ProbVLM`,

$$\mathbf{\Psi}(\cdot;\zeta) := (\mathbf{\Psi}_{\mathcal{V}}(\cdot;\zeta_{\mathcal{V}}), \mathbf{\Psi}_{\mathcal{T}}(\cdot;\zeta_{\mathcal{T}})) \tag{6.1}$$

where $\mathbf{\Psi}_{\mathcal{V}}$ and $\mathbf{\Psi}_{\mathcal{T}}$ represents the vision and text encoders parameterized by $\zeta_{\mathcal{V}}$ and $\zeta_{\mathcal{T}}$, respectively. Also, $\zeta := \zeta_{\mathcal{V}} \cup \zeta_{\mathcal{T}}$ represents the overall parameters for `ProbVLM`. that learns to estimate the parameters $\{\hat{\mathbf{z}}, \hat{v}...\hat{\rho}\}$ with the help of frozen encoders $\mathbf{\Phi}_{\mathcal{V}}(\cdot;\theta^*_{\mathcal{V}})$ and $\mathbf{\Phi}_{\mathcal{T}}(\cdot;\theta^*_{\mathcal{T}})$. The functions $\mathbf{\Psi}_{\mathcal{V}}(\cdot;\zeta_{\mathcal{V}})$ and $\mathbf{\Psi}_{\mathcal{T}}(\cdot;\zeta_{\mathcal{T}})$ operate on image and text embeddings respectively, but during training depend on both modalities, as discussed later. We design the learning scheme for $\mathbf{\Psi}(\cdot;\zeta)$ such that: (i) Estimated parameter $\hat{\mathbf{z}}$ should remain faithful to the original unimodal embedding $\mathbf{z}$ (*intra-modal alignment*), this makes the uncertainty of the `ProbVLM` serve as a good proxy for the uncertainty of frozen encoders. (ii) Estimated parameters $\{\hat{v}...\hat{\rho}\}$ should capture the ambiguities and uncertainties present within and across modalities (*cross-modal alignment*). Figure 6.2 depicts `ProbVLM` in tandem with the frozen VLM.

Intra-modal Alignment. To ensure that the mean of the distribution estimated by $\mathbf{\Psi}(\cdot;\zeta)$ reflects the point estimates provided by the frozen encoders, we set up a probabilistic reconstruction problem for the embeddings within the modalities. That is, for a

given sample $\mathbf{x}$ (either from image or text modality), we obtain the embedding from the frozen encoder $\mathbf{z} = \mathbf{\Phi}(\mathbf{x}; \theta)$ (using the appropriate encoder), then the modality-specific component of $\mathbf{\Psi}(\cdot; \zeta)$ learns to reconstruct the $\mathbf{z}$ (let the reconstruction be called $\hat{\mathbf{z}}$). The modality-specific component of $\mathbf{\Psi}(\cdot; \zeta)$ is designed to (i) relax the i.i.d constraints by assuming independent but *not* identically distributed residuals and (ii) learn the *heteroscedasticity* for the residuals at the time of reconstruction that may follow the heavy-tailed distributions [268, 266, 133, 132, 100]. The modality-specific component is learned by maximizing the likelihood, $\mathcal{L}(\zeta; \{\mathbf{z}_i\}_{i=1}^N)$ for the embeddings of N samples in the datasets. That is, the modality-specific optimal parameters are given by,

$$\zeta^* := \underset{\zeta}{\operatorname{argmax}}\ \mathcal{L}(\zeta; \{\mathbf{z}_i\}_{i=1}^N) = \prod_{i=1}^{N} \frac{\hat{\beta}_i e^{-(|\hat{\mathbf{z}}_i - \mathbf{z}_i|/\hat{\alpha}_i)^{\hat{\beta}_i}}}{2\hat{\alpha}_i \Gamma(1/\hat{\beta}_i)} \tag{6.2}$$

In the above equation, $\frac{\hat{\beta}_i e^{-(|\hat{\mathbf{z}}_i - \mathbf{z}_i|/\hat{\alpha}_i)^{\hat{\beta}_i}}}{2\hat{\alpha}_i \Gamma(1/\hat{\beta}_i)}$ represents the *generalized Gaussian distribution* (GGD, represented by $\mathcal{G}$) that is capable of modeling heavy-tailed distributions (note the Gaussian and Laplace are special cases of $\mathcal{G}$ with $\alpha = 1, \beta = 2$ and $\alpha = 1, \beta = 1$, respectively). The variables $\hat{\mathbf{z}}_i, \hat{\alpha}_i, \hat{\beta}_i$ are the predicted mean, scale, and shape parameters of $\mathcal{G}$ from our modality-specific components for the given input $\mathbf{z}_i$. We obtain modality-specific optimal parameters by minimizing negative log-likelihood (equivalent to Equation 6.2). Given $\mathbf{z}$ and predicted $\hat{\mathbf{z}}, \hat{\alpha}, \hat{\beta}$, loss is given by,

$$L_{\text{rec}}(\zeta) := \left(\frac{|\hat{\mathbf{z}} - \mathbf{z}|}{\hat{\alpha}}\right)^{\hat{\beta}} - \log \frac{\hat{\beta}}{\hat{\alpha}} + \log \Gamma(\frac{1}{\hat{\beta}}) \tag{6.3}$$

Therefore, the vision-specific component of `ProbVLM`, $\mathbf{\Psi}(\cdot; \zeta_{\mathcal{V}})$), is trained by minimizing the Eqation 6.3 using image embeddings, we denote this loss as $L_{\text{rec}}^{\mathcal{V}}(\zeta_V)$. Similarly the text-specific component, $\mathbf{\Psi}(\cdot; \zeta_{\mathcal{T}})$, is trained by minimizing $L_{\text{rec}}^{\mathcal{T}}(\zeta_T)$. As discussed next, we also enforce cross-modal alignment so that the predicted distribution of `ProbVLM` captures the uncertainties across modalities from one-to-many correspondences for an embedding.

Cross-modal Alignment. While the intra-modal alignment seeks to match the means of the output distribution from `ProbVLM` to the embeddings derived from frozen vision-language encoders, we also enforce the image and text embedding output distribution (from `ProbVLM`) belonging to similar concepts to remain close to each other. That is, given an image and text embedding pair $(\mathbf{z}_{\mathcal{V}}, \mathbf{z}_{\mathcal{T}})$ (from frozen model) representing similar concepts, the output distributions from $\mathbf{\Psi}(\cdot; \zeta)$, $\mathcal{G}(\mathbf{z}; \hat{\mathbf{z}}_{\mathcal{V}}, \hat{\alpha}_{\mathcal{V}}, \hat{\beta}_{\mathcal{V}})$ and $\mathcal{G}(\mathbf{z}; \hat{\mathbf{z}}_{\mathcal{T}}, \hat{\alpha}_{\mathcal{T}}, \hat{\beta}_{\mathcal{T}})$ (later referred to as $\mathcal{G}_{\mathcal{V}}(\mathbf{z})$) and $\mathcal{G}_{\mathcal{T}}(\mathbf{z})$) should match. This can be measured directly from the likelihood as, $p(\mathbf{z}_v = \mathbf{z}_u)$, where $\mathbf{z}_v \sim \mathcal{G}_{\mathcal{V}}(\mathbf{z})$ and $\mathbf{z}_u \sim \mathcal{G}_{\mathcal{T}}(\mathbf{z})$ as in [234] , i.e.,

$$p(\mathbf{z}_v = \mathbf{z}_u) := \iint \mathcal{G}_{\mathcal{V}}(\mathbf{z}_v) \mathcal{G}_{\mathcal{T}}(\mathbf{z}_u) \delta(\mathbf{z}_v - \mathbf{z}_u) d\mathbf{z}_v d\mathbf{z}_u \tag{6.4}$$

where $\delta(\cdot)$ refers to the *Dirac-delta distribution*. The above integral can be simplified further by defining $\Delta\mathbf{z} = \mathbf{z}_v - \mathbf{z}_u$ and seeking $p(\Delta\mathbf{z}) = 0$. As both $\mathbf{z}_v$ and $\mathbf{z}_u$ are GGD random

variables, $\Delta\mathbf{z}$ follows the distribution based on the *Bivariate Fox H-function* [243, 165, 168] given by,

$$\Delta z \sim \frac{1}{2\Gamma(1/\hat{\beta}_{\mathcal{V}}),\Gamma(1/\hat{\beta}_{\mathcal{T}})} \times \int \mathcal{H}_{1,2}^{1,1}\left[At^2\middle|\begin{matrix}(1-\frac{1}{\hat{\mathbf{z}}_{\mathcal{V}}},\frac{1}{\hat{\mathbf{z}}_{\mathcal{T}}})\\(0,1)(\frac{1}{2},1)\end{matrix}\right]\mathcal{H}_{1,2}^{1,1}\left[Bt^2\middle|\begin{matrix}(1-\frac{1}{\hat{\mathbf{z}}_{\mathcal{T}}},\frac{1}{\hat{\mathbf{z}}_{\mathcal{T}}})\\(0,1)(\frac{1}{2},1)\end{matrix}\right]\cos t(\mu-z)dt \tag{6.5}$$

Where $A = \frac{\hat{\alpha}_{\mathcal{V}}^2\Gamma(1/\hat{\beta}_{\mathcal{V}})}{4\Gamma(3/\hat{\beta}_{\mathcal{V}})}$, $B = \frac{\hat{\alpha}_{\mathcal{T}}^2\Gamma(1/\hat{\beta}_{\mathcal{T}})}{4\Gamma(3/\hat{\beta}_{\mathcal{T}})}$, $\mu = \hat{\mathbf{z}}_v - \hat{\mathbf{z}}_u$, and $\mathcal{H}$ is the *Fox H function* [243, 165, 168]. Equation B.2 does not provide a scalable objective function suitable for training deep neural networks. Hence, we propose an approximation that is easily scalable for deep-learning models given by,

$$\begin{aligned} p(\mathbf{z}_v = \mathbf{z}_u) &= \iint \mathcal{G}_{\mathcal{V}}(\mathbf{z}_v)\mathcal{G}_{\mathcal{T}}(\mathbf{z}_u)\delta(\mathbf{z}_v - \mathbf{z}_u)d\mathbf{z}_v d\mathbf{z}_u \\ &\approx \int \frac{1}{2}\left(\mathcal{G}_{\mathcal{V}}(\mathbf{z})\delta(\mathbf{z}-\mathbf{z}_{\mathcal{T}}) + \mathcal{G}_{\mathcal{T}}(\mathbf{z})\delta(\mathbf{z}-\mathbf{z}_{\mathcal{V}})\right)d\mathbf{z} \end{aligned} \tag{6.6}$$

The appendix shows details of the above equation. The first term in the integral, $\int \mathcal{G}_{\mathcal{V}}(\mathbf{z})\delta(\mathbf{z}-\mathbf{z}_{\mathcal{T}})d\mathbf{z}$, is the likelihood of the text embedding $\mathbf{z}_{\mathcal{T}}$ under the predicted distribution, $\mathcal{G}_{\mathcal{V}}(\mathbf{z})$, for the visual embedding. Similarly, the second term is the likelihood of the visual embedding $\mathbf{z}_{\mathcal{V}}$ under the predicted distribution, $\mathcal{G}_{\mathcal{T}}(\mathbf{z})$, for the text embedding. Negative log of Equation B.4 leads to a scalable objective function that can be used to learn the optimal parameters for vision and text components of `ProbVLM` ($\mathbf{\Psi}_{\mathcal{V}}(\cdot;\zeta_{\mathcal{V}})$ and $\mathbf{\Psi}_{\mathcal{T}}(\cdot;\zeta_{\mathcal{T}})$),

$$\begin{aligned} L_{\text{cross}}(\zeta_{\mathcal{V}},\zeta_{\mathcal{T}}) := \underbrace{\left(\frac{|\hat{\mathbf{z}}_{\mathcal{V}}-\mathbf{z}_{\mathcal{T}}|}{\hat{\alpha}_{\mathcal{V}}}\right)^{\hat{\beta}_{\mathcal{V}}} - \log\frac{\hat{\beta}_{\mathcal{V}}}{\hat{\alpha}_{\mathcal{V}}} + \log\Gamma(\frac{1}{\hat{\beta}_{\mathcal{V}}})+}_{\text{Cross-modal: vision}\rightarrow\text{text}} \\ \underbrace{\left(\frac{|\hat{\mathbf{z}}_{\mathcal{T}}-\mathbf{z}_{\mathcal{V}}|}{\hat{\alpha}_{\mathcal{T}}}\right)^{\hat{\beta}_{\mathcal{T}}} - \log\frac{\hat{\beta}_{\mathcal{T}}}{\hat{\alpha}_{\mathcal{T}}} + \log\Gamma(\frac{1}{\hat{\beta}_{\mathcal{T}}})}_{\text{Cross-modal: text}\rightarrow\text{vision}} \end{aligned} \tag{6.7}$$

The overall objective used for `ProbVLM` is designed to be,

$$L_{\text{ProbVLM}}(\zeta_{\mathcal{V}},\zeta_{\mathcal{T}}) = L_{\text{rec}}^{\mathcal{V}}(\zeta_{\mathcal{V}}) + L_{\text{rec}}^{\mathcal{T}}(\zeta_{\mathcal{T}}) + \lambda_{\text{cross}}L_{\text{cross}}(\zeta_{\mathcal{V}},\zeta_{\mathcal{T}}) \tag{6.8}$$

where λ_{cross} is a hyperparameter controlling the relative contribution of inter-intra modality terms.

Uncertainty Quantification. Given embedding $\mathbf{z}$ from a frozen encoder, predicted distributions from the trained `ProbVLM` (output from the appropriate component) allows aleatoric uncertainty estimation as $\hat{\sigma}^2_{\text{aleatoric}} = \frac{\hat{\alpha}^2\Gamma(3/\hat{\beta})}{\Gamma(1/\hat{\beta})}$. Moreover, we design both $\mathbf{\Psi}_{\mathcal{V}}$ and $\mathbf{\Psi}_{\mathcal{T}}$ to be simple 3-layer MLPs with dropout layers (with dropout probability set to 0.1 during training). Activating dropouts during inference, with multiple forward passes (say M), allows estimating the epistemic uncertainty, $\hat{\sigma}^2_{\text{epistemic}} = \frac{1}{M}\sum_{m=1}^{M}(\hat{\mathbf{z}}_m - \frac{1}{M}\sum_{j=1}^{M}\hat{\mathbf{z}}_j)^2$. We estimate total uncertainty as,

$$\hat{\sigma}^2_{\text{total}} = \hat{\sigma}^2_{\text{epistemic}} + \hat{\sigma}^2_{\text{aleatoric}} \tag{6.9}$$

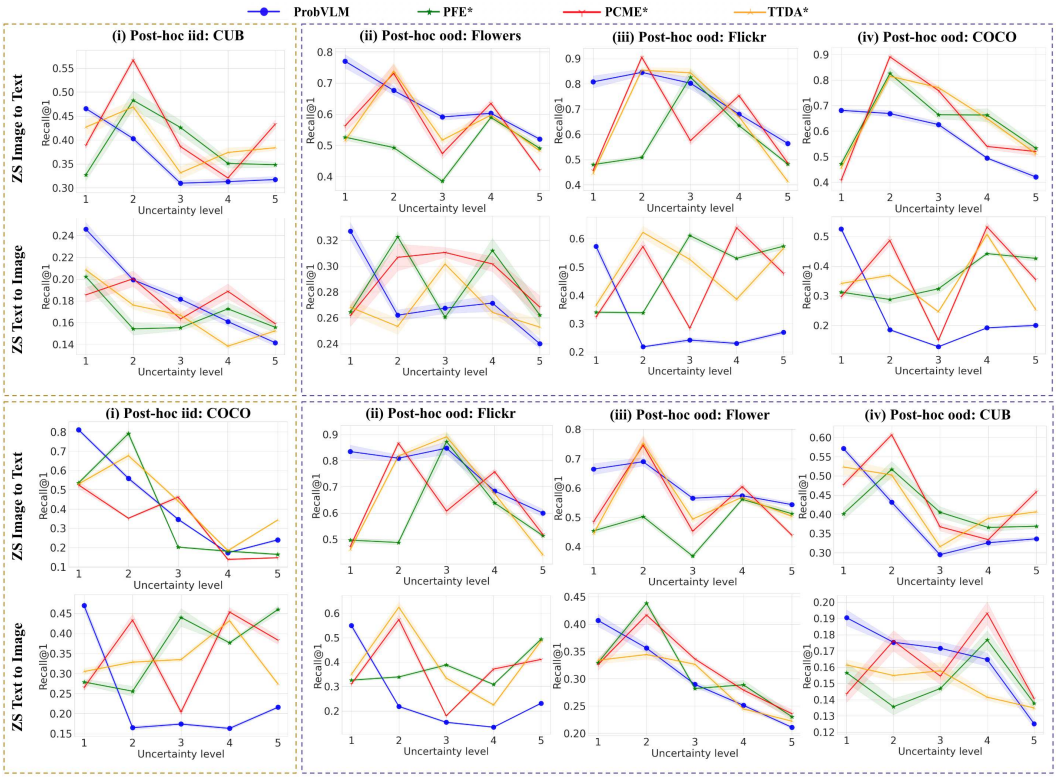

Figure 6.3: Measuring the calibration with various post-hoc method for Image-to-Text and Text-to-Image retrieval when trained on (top) CUB and (bottom) COCO, and evaluated on CUB, COCO, Flickr, FLO.

6.4.3 Latent Diffusion for Probabilistic Embeddings

For a given text embedding $\mathbf{z}_{\mathcal{T}}$, the distribution estimated via `ProbVLM`, $\mathcal{G}(\mathbf{z}; \hat{\mathbf{z}}_{\mathcal{T}}, \hat{\alpha}_{\mathcal{T}}, \hat{\beta}_{\mathcal{T}})$ can be visualized by drawing samples from the predicted distribution of vectors (say, $\{\hat{\mathbf{z}}_{\mathcal{T},i}\}_{i=1}^{Q}$) and passing them through a latent diffusion model, e.g., *Stable Diffusion* (say, $\mathbf{\Omega}(\cdot; \theta_{\Omega}^{*})$) using CLIP text encoder, to synthesize the set of samples (say, J) from the corresponding distribution of images, i.e.,

$$J := \{\mathbf{\Omega}(\hat{\mathbf{z}}_i; \theta_{\Omega})\}_{i=1}^{Q} \tag{6.10}$$

Section 6.5.4 uses this to visualize the predicted distributions.

6.5 Experiments and Results

We start by highlighting our tasks, datasets, and evaluation metrics. We also compare our model to various state-of-the-art methods quantitatively and qualitatively in Section 6.5.1. In Section 6.5.2, we provide an ablation analysis, and Section 6.5.3 demonstrates some real-world applications of `ProbVLM` for model selection and active learning.

Datasets, Metrics, and Baselines. We use the MS-COCO [154], Flickr-30k [201], and the CUB [275] as they are widely used for cross-modal retrieval [37, 60, 242].

Furthermore, we adapt the Oxford-Flowers 102 (FLO) dataset [187] similar to [37] as an additional benchmark for cross-modal retrieval in a fine-grained setting. We evaluate the performance of both Image-to-Text retrieval and Text-to-Image Retrieval using the Recall@k (R@k) metric. To evaluate the calibration of the uncertainty estimates, we define uncertainty levels [37] and use the Spearman rank correlation (denoted by S) between the uncertainty level and the Recall@k for retrieval. For an ideal model, performance would decrease monotonically with increasing uncertainty levels, leading to a score of -1. We also compute the R^2 for the regression fit between the uncertainty levels and R@1 performances to measure if the drop in performance follows a linear trend. Finally, we also measure the product of these two scores (as a unified metric), i.e., $-SR^2$, which should be 1.0 for an ideal model.

			i2t				t2i			
VL	M	Metrics	COCO	Flickr	FLO	CUB	COCO	Flickr	FLO	CUB
CLIP	ProbVLM	S ↓	**-0.99**	**-0.70**	**-0.90**	**-0.60**	**-0.30**	**-0.70**	**-0.99**	**-0.89**
		R^2 ↑	**0.93**	**0.71**	**0.62**	**0.67**	**0.35**	**0.50**	**0.99**	**0.70**
		$-SR^2$ ↑	**0.93**	**0.49**	**0.56**	**0.40**	**0.10**	**0.35**	**0.99**	**0.63**
	PFE*[234]	S ↓	-0.79	-0.19	0.60	**-0.60**	0.79	0.30	-0.89	-0.10
		R^2 ↑	0.59	0.01	0.30	0.28	0.74	0.44	0.52	0.00
		$-SR^2$ ↑	0.47	0.00	-0.18	0.17	-0.59	-0.13	0.47	-0.00
	PCME*[37]	S ↓	-0.89	-0.30	-0.30	**-0.60**	0.30	0.09	-0.70	0.30
		R^2 ↑	0.75	0.07	0.07	0.20	0.16	0.01	0.57	0.01
		$-SR^2$ ↑	0.68	0.02	0.02	0.12	-0.05	-0.00	0.40	-0.00
	TTDA[9]	S ↓	-0.79	-0.30	0.00	**-0.60**	-0.10	-0.19	-0.89	-0.50
		R^2 ↑	0.69	0.09	0.00	0.41	0.26	0.071	0.80	0.15
		$-SR^2$ ↑	0.55	0.03	0.00	0.24	0.00	0.01	0.73	0.07
BLIP	ProbVLM	S ↓	**-0.87**	**-0.79**	**-0.74**	**-0.66**	**-0.43**	**-0.38**	**-0.31**	**-0.22**
		R^2 ↑	**0.92**	**0.83**	**0.68**	**0.61**	**0.52**	**0.48**	**0.45**	**0.38**
		$-SR^2$ ↑	**0.80**	**0.66**	**0.50**	**0.40**	**0.22**	**0.18**	**0.14**	**0.08**
	PFE*[234]	S ↓	-0.82	-0.74	-0.63	-0.63	-0.39	-0.32	-0.28	-0.18
		R^2 ↑	0.72	0.76	0.62	0.44	0.48	0.38	0.39	0.37
		$-SR^2$ ↑	0.58	0.57	0.39	0.27	0.19	0.12	0.11	0.07
	PCME*[37]	S ↓	-0.76	-0.53	-0.60	-0.44	-0.28	-0.26	-0.28	-0.21
		R^2 ↑	0.81	0.56	0.60	0.53	0.50	0.34	0.44	0.36
		$-SR^2$ ↑	0.62	0.29	0.36	0.23	0.14	0.09	0.12	**0.08**
	TTDA[9]	S ↓	-0.44	-0.33	-0.74	-0.60	-0.19	-0.26	-0.21	-0.21
		R^2 ↑	0.66	0.56	0.42	0.55	0.49	0.23	0.35	0.36
		$-SR^2$ ↑	0.29	0.18	0.31	0.33	0.10	0.06	0.07	**0.08**

Table 6.1: Metrics to evaluate the calibration of the uncertainty estimates for both CLIP [205] and BLIP [150] Vision-Language models for all considered methods, trained on COCO and evaluated on COCO, Flickr, CUB, and FLO.

Since there is *no prior work* to estimate probabilistic embeddings from a deterministic model in a cross-modal setting, we adapt a few existing ideas for the task. The first baseline is adapted from PFE [234], where we learn the covariances for the heteroscedastic Gaussian distribution while keeping the mean fixed to the embeddings derived from the frozen encoders in each modality. The second is to use the soft-contrastive objective of PCME[37] to train a probabilistic adapter in a post-hoc fashion. Finally, we also have a baseline that performs perform Test-Time Data Augmentation (TTDA) on the inputs [9, 278]. This is done by perturbing the images and masking out words in the text. While TTDA does not require additional training, we train our `ProbVLM` and other baselines on datasets like COCO, Flickr, CUB, and FLO.

Implementation Details. Our `ProbVLM` consists of a Multi-Layer Perceptron (MLP) for both the image and text encoder, each consisting of an input layer going from embedding dimension to 256, a hidden layer of size 256, and an output layer going from 256 to embedding dimensions. This is trained for 100 epochs with a learning rate of $1e^{-4}$. More details are available in the supplementary.

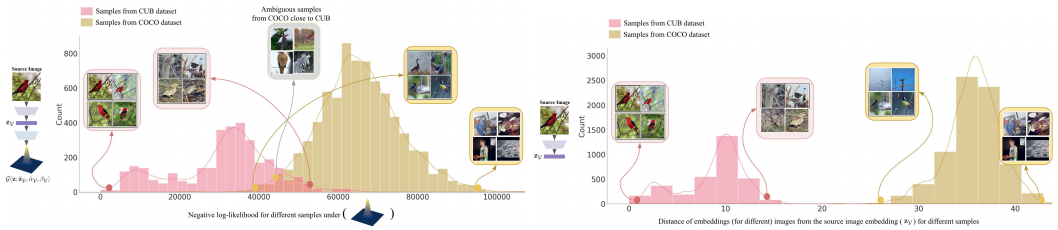

Figure 6.4: Visualizing the uncertainties of the vision encoder captured by ProbVLM. Fixing an image from CUB, we obtain the predicted embedding distribution and compute the likelihood of all other samples in CUB and COCO. We observe that the images in COCO are similar/ambiguous to CUB overlap (Top). However, deterministic embeddings lead to a separation between the two datasets (Bottom).

6.5.1 Calibrated Uncertainty via ProbVLM

We investigate the calibration of the uncertainty derived from ProbVLM for the cross-modal retrieval task. All models trained on CUB and COCO were evaluated on all four datasets. Calibration plots are illustrated in Figure 6.3. We observe that the R@1 performance consistently drops for ProbVLM as we increase the uncertainty levels, whereas the baselines rarely see a monotonic drop in performance. We quantify this performance in Table 6.1. The highest score of 0.93 for $-SR^2$ (i2t) on the COCO dataset indicates a decreasing performance trend with increasing uncertainty. Notably, the uncertainty estimates indicate the average performance in different bins even when ProbVLM is evaluated on datasets that are different from the train set. In some cases, we see that ProbVLM even achieves a nearly perfect score ($-SR^2$ of 0.99, with CLIP VLM on FLO, after training on COCO for Image-to-Text Retrieval). On the contrary, we find that the baselines often achieve poor scores. It is important to note that all these models use the same underlying embeddings and achieve the same performance on the retrieval task. Of all the considered methods, ProbVLM provides the most calibrated uncertainty estimates. We see similar trends for ProbVLM with BLIP [150], where ProbVLM achieves a $-SR^2$ of 0.80, when trained on COCO and evaluated on COCO, compared to other methods like PFE* (0.58), PCME* (0.62), and TTDA (0.29).

Figure 6.4-(Top) visualizes the ambiguities captured by ProbVLM on the visual embeddings. We take a bird image (source) from the CUB dataset and obtain the probability distribution for the visual embedding of that sample; we then compute the likelihood of the visual embeddings (i.e., point estimates derived from CLIP) for the other samples of CUB and COCO datasets, under the source distribution. We notice that within the CUB dataset, the bird images similar to the source image have a higher likelihood compared to other bird images. Also, the images from the COCO dataset tend to have a lower likelihood. However, some images from the COCO dataset have a likelihood similar to the samples from CUB. We visualize these samples and discover them to be bird images.

Moreover, the overlapping region between CUB and COCO has samples from the COCO dataset that are ambiguous and related to bird images as they have similar backgrounds, etc. On the contrary, when a similar analysis is performed using the CLIP (by measuring the distance between the embeddings instead of likelihood, Figure 6.4-(Bottom)), we notice that the two datasets are well separated and ambiguities are not captured.

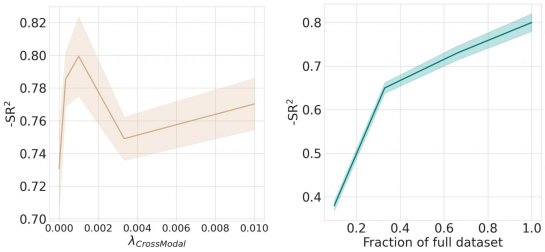

Figure 6.5: Plot indicating (left) necessity of the cross-modal alignment and (right) data required to train `ProbVLM`.

6.5.2 Ablations

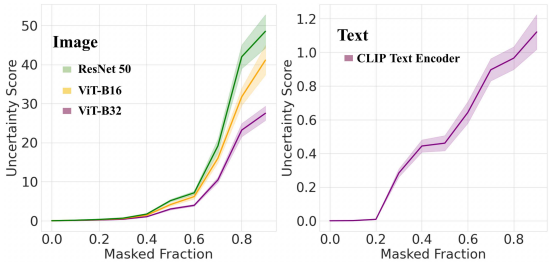

Figure 6.6: Uncertainty increases with increased masking of the input images (Left) and texts (Right). Results with three vision encoders and one language encoder from CLIP.

We ablate different components of our proposed `ProbVLM`, to provide a deeper understanding of its workings. First, we perform a sensitivity analysis on the cross-modal reconstruction objective, as shown in Figure 6.5-(Left), for `ProbVLM` on BLIP using the COCO dataset. We need a non-zero coefficient of the cross-modal loss to ensure that `ProbVLM` learns meaningful uncertainties that capture the ambiguities across modalities and are well-correlated with its performance on the downstream retrieval task. Similarly, having a large co-efficient for the cross-modal loss could hinder learning a faithful identity reconstruction, thereby hampering the performance of the downstream evaluation.

Next, we investigate the amount of data that is required to train `ProbVLM` in Figure 6.5-(Right). We get satisfactory calibration of the uncertainty estimates while using only 50% of the dataset (shown for `ProbVLM` on BLIP using COCO), indicating that `ProbVLM` is highly data-efficient.

Further, we investigate the uncertainties by masking out increasing portions of the input image/text in Figure 6.6. We use three different CLIP backbones for the images, ViT-B/32, ViT-B/16, ResNet50, and GPT-based language encoder from CLIP [205, 204]. The mean uncertainty steadily increases as we mask increasing amounts of input.

6.5.3 Applications

We study the utility of the uncertainty estimates derived from `ProbVLM` on two critical applications not well reviewed for VLMs: active learning and model selection.

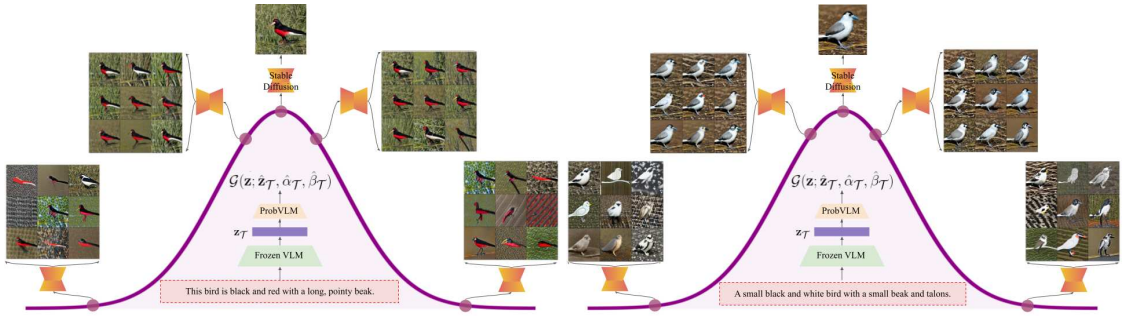

Figure 6.8: Visualizing the predicted embedding distributions from ProbVLM using a large-scale pre-trained diffusion model, i.e., *Stable Diffusion*. The example is shown for two different captions from CUB dataset, for which the point-estimate embedding vector is obtained via CLIP, and the distribution is obtained via ProbVLM.

Active Learning. Here, we choose a small subset of the unlabeled dataset to fine-tune the model [41]. In this case, we wish to finetune the CLIP model on the FLO dataset while using a limited amount of labeled data. To achieve this, we estimate the uncertainty of the image embeddings using ProbVLM (trained using a diverse dataset like COCO). We then select the top-k samples sorted by their mean uncertainty in the visual embeddings and fine-tune the CLIP model using them with a contrastive objective [205]. Results with varying budgets are shown in Figure 6.7. Selecting samples based on uncertainty scores significantly outperforms random sampling for all considered visual backbones.

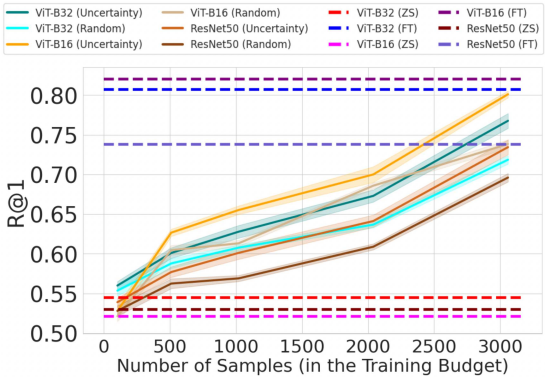

Figure 6.7: Results for active learning, with different vision encoders and varying training budgets. For a given encoder, uncertainty-based sampling outperforms random sampling.

Pretrained Model Selection. We are given a set of models trained on different data distributions. We aim to select the best model for the target distribution for which we have unlabeled samples. This has been explored mostly in the context of classification previously [73, 86, 33, 36, 51, 52].

We consider the specific case of having the CLIP models fine-tuned on three datasets, and the fourth dataset is held out, for which we only have the images. We compute the mean uncertainty on these images using ProbVLM whose weights are interpolated from all the source datasets [297, 298, 106, 105]. This is to ensure that the uncertainties on all these models are comparable. The results for this experiment are shown in Table 6.2. On CUB, Flickr, and COCO, the source model with the lowest uncertainty has the best performance on the target dataset, and on FLO dataset, the model with the least uncertainty has the 2nd best performance (R@1 of 47.9 vs 49.5 for the best model). This indicates that the

uncertainties provided by ProbVLM can be used as a signal to predict the performance on unlabelled samples for retireval.

6.5.4 Latent Diffusion for Embedding Uncertainty

To further understand the semantics of the predicted embedding distribnutions from the ProbVLM, we visualize the text embedding distributions by sampling the embedding vectors from the predicted distribution for a caption (converted to embedding vector using CLIP) and passing it through the pre-trained latent diffusion model using CLIPs text encoder, *stable diffusion*, as shown in Figure 6.8 and described in details in Section 6.4.3. We observe from Figure 6.8 that the samples obtained closer to the mean (i.e., sampled embedding vector similar to the one generated by CLIP for the caption) lead to meaningful variations in the generated images, e.g., for the left caption, close to the mean of the distribution, the generated samples show variations in the shape and colour of the beak, wings, and feet. Whereas far away from the mean of the distributions, i.e., on the tails, we start seeing images with strong artifacts that no longer preserves the semantics of the caption. We observe this for another example as well shown in Figure 6.8-(Right).

		Metrics			
D	**Models**	**Uncertainty score**	**R@1**	**R@5**	**R@10**
CUB	CLIP-ViT32-COCO	11.92	31.5	61.0	75.8
	CLIP-ViT32-Flickr	**9.37**	**32.4**	**64.2**	**76.9**
	CLIP-ViT32-FLO	15.43	22.8	49.8	64.9
FLO	CLIP-ViT32-COCO	**11.83**	47.9	79.2	88.5
	CLIP-ViT32-Flickr	13.55	**49.5**	**84.6**	**93.9**
	CLIP-ViT32-CUB	18.39	37.7	69.4	82.8
Flickr	CLIP-ViT32-COCO	**9.61**	**88.8**	**97.8**	**99.8**
	CLIP-ViT32-CUB	16.49	25.8	47.4	55.6
	CLIP-ViT32-FLO	13.67	52.8	77.8	85.2
COCO	CLIP-ViT32-Flickr	**7.28**	**58.1**	**80.9**	**88.2**
	CLIP-ViT32-CUB	15.37	8.8	21.7	29.8
	CLIP-ViT32-FLO	12.44	23.9	46.6	58.8

Table 6.2: Results for the model selection experiment. ProbVLM accurately identifies the best performing source model using only unlabeled samples of the target dataset.

6.6 Conclusion

We introduce ProbVLM, a post-hoc method for estimating the embedding distribution for a frozen large-scale deterministic vision-language model. We efficiently estimate calibrated uncertainties using our framework and show that such calibrated estimates have a variety of applications in downstream tasks such as model selection and active learning. Furthermore, we perform experiments to interpret embedding distribution predicted by ProbVLM using a large-scale pre-trained latent diffusion model (i.e., *Stable Diffusion*). We hope our work highlights and inspires future work on efficient methods for probabilistic embeddings.

7

Likelihood Annealing: Fast Calibrated Uncertainty for Regression

7.1 Abstract

Recent advances in deep learning have shown that uncertainty estimation is becoming increasingly important in applications such as medical imaging, natural language processing, and autonomous systems. However, accurately quantifying uncertainty remains a challenging problem, especially in regression tasks where the output space is continuous. Deep learning approaches that allow uncertainty estimation for regression problems often converge slowly and yield poorly calibrated uncertainty estimates that can not be effectively used for quantification. Recently proposed post hoc calibration techniques are seldom applicable to regression problems and often add overhead to an already slow model training phase. This work presents a fast calibrated uncertainty estimation method for regression tasks called *Likelihood Annealing*, that consistently improves the convergence of deep regression models and yields calibrated uncertainty without any post hoc calibration phase. Unlike previous methods for calibrated uncertainty in regression that focus only on low-dimensional regression problems, our method works well on a broad spectrum of regression problems, including high-dimensional regression. Our empirical analysis shows that our approach is generalizable to various network architectures, including multilayer perceptrons, 1D/2D convolutional networks, and graph neural networks, on five vastly diverse tasks, i.e., chaotic particle trajectory denoising, physical property prediction of molecules using 3D atomistic representation, natural image super-resolution, and medical image translation using MRI.

7.2 Introduction

Uncertainty estimation is an essential building block to provide interpretability and secure reliability in modern machine learning systems [231, 126, 272, 99] that offer intelligent

solutions for numerous real-world applications, ranging from medical analytics [146, 79, 263] to autonomous driving [302, 231, 21]. Recent advances have explored various formulations to provide accurate predictions and uncertainty estimates for deep neural networks, as represented by Bayesian approaches [70, 120, 164], ensembles [138], pseudo-ensembles [171, 65], and quantile regression [215, 303, 62] methods. However, these existing methods are often computationally expensive – e.g., slow convergence rate during training or inefficient inference cost due to multiple forward passes – while being poorly calibrated for uncertainty estimates. Moreover, some of these methods are proposed for low-dimensional regression tasks [38, 336, 31] (i.e., regressing a scalar value) and do not scale for high-dimensional regression (i.e., regressing large matrices or tensors). This paper presents a unified formulation to resolve these issues for estimating fast, well-calibrated uncertainty in deep regression models for a wide spectrum of regression problems, including chaotic particle trajectory denoising, physical property prediction of molecules using 3D atomistic representation, natural image super-resolution, and medical image translation using MRI.

We propose to revisit deep regression models trained via maximum likelihood estimation (MLE), which assumes a Gaussian distribution over the regression output and optimizes the negative log-likelihood to estimate the target and uncertainty. Although such models can ensure low regression error (i.e., high accuracy) and encapsulate the predictive uncertainty, they often converge slowly at the beginning of training due to a flat gradient landscape. Further, they may even risk gradient explosion caused by a steep gradient landscape when reaching the optima (detailed in Section 7.4.1), leading to poorly calibrated uncertainty estimates that do not offer credible interpretability for the model and cannot be used for downstream applications.

To reshape the aforementioned ill-posed gradient landscape that causes slow convergence and poorly calibrated uncertainty, we propose a novel *Likelihood Annealing* (LIKA) scheme for deep regression models that alters the original gradients by formulating a temperature-dependent improper likelihood to be optimized during the learning phase. In contrast to the standard likelihood for regression that enforces a fixed Gaussian distribution on the target, we introduce a temperature hyperparameter to impose an evolving distribution.

The proposed temperature-dependent likelihood brings crucial properties to regression uncertainty. First, the multimodal distribution on the regression target ensures that at high residuals (between output and ground truth, occurring in the initial learning phase), the gradients are much larger than the standard unimodal Gaussian distribution (explained in detail in Section 7.4 and Figure 7.1) leading to faster convergence at the beginning of the learning phase. Second, we also anneal the learning rate over the course of training along with the temperature that avoids gradient explosion towards the end of the learning phase, a problem with the standard heteroscedastic Gaussian-based likelihood distribution with sharp gradients at lower errors. Third, we construct the temperature-dependent likelihood such that the predicted uncertainty is *encouraged* to be calibrated at

every step, by being close to the error between the prediction and ground truth.

The standard unimodal distribution faces slow convergence in the beginning and potential gradient explosion towards the end of the learning phase and provides poorly calibrated uncertainty estimates. In contrast, our LIKA method allows faster convergence and offers well-calibrated uncertainty estimates for a broad spectrum of regressions. This also differs from uncertainty regression methods that estimate the full quantile as they are often shown to be effective on low-dimensional regression.

Contributions. We introduce a temperature-dependent likelihood annealing scheme for deep regression models with uncertainty estimation that leads to faster model convergence and offers better-calibrated uncertainty (detailed in Section 7.4.3). We conduct a comprehensive evaluation on various datasets, including chaotic particle trajectory denoising, physical property prediction of molecules using 3D atomistic representation, image super-resolution, and medical image translation using MRI images, presented in Section 7.5.

7.3 Related Work

Deep neural networks (DNNs) typically estimate inaccurate uncertainty due to their deterministic form that is insufficient for characterizing the accurate confidence [69, 87]. Bayesian inference has been widely studied to effectively estimate uncertainty. Directly performing Bayesian inference on deep nonlinear networks is infeasible due to intractable computations. Hence, approximate inference has been explored by variational inference [85, 24, 50, 164] or MCMC-based approximation [291, 34]. However, due to its approximation, the estimated uncertainty may fail to follow the true uncertainty quantification [138]. Moreover, compared with typical DNNs, approximate Bayesian inference is computationally more expensive and has slower convergence in practice. Non-Bayesian methods have been proposed as an alternative. For instance, [120, 138] modeled two terms, i.e. predictive mean and variance, as an output of DNN to estimate the uncertainty directly from the network's output. Another line of work estimates the uncertainty in the prediction in a non-parametric manner by estimating different quantiles for a given input [155, 31, 336, 38]. Moreover, there are also works from conformal predictions that quantify uncertainty by constructing prediction intervals, which are sets of possible outcomes that are believed to contain the true value with a certain probability [293, 173, 322].

In general, there are two broad types of uncertainties in deep learning: (i) Aleatoric and (ii) Epistemic. Aleatoric uncertainty is the uncertainty that arises from the inherent randomness in the data. In contrast, Epistemic uncertainty is the uncertainty that arises due to a lack of knowledge or information about the data. In real-world scenarios with access to large datasets, aleatoric uncertainty is often critical because it is directly related to the variability in the data, which is essential to modeling real-world scenarios [176, 178, 9]. For example, in medical imaging, different patients may have different degrees of variability in their images due to different factors such as the presence of diseases, body

types, or imaging equipment [278, 269, 53]. By modeling aleatoric uncertainty, we can better capture this variability and improve the accuracy of the model. On the other hand, epistemic uncertainty can be reduced by acquiring more data or improving the model architecture [35, 120, 248]. This work focuses on estimating the aleatoric uncertainty in deep regression problems.

Calibrating the inaccurate uncertainty is another way to estimate accurate uncertainty [87]. In the regression task, calibration was first defined in a quantile manner [131]. That is, the estimated credible interval with confidence level α (e.g. 95%) is calibrated if α% of the ground-truth target is covered in that interval. There are post-processing methods for regression calibration [131, 198, 249]. For instance, [131] introduced an auxiliary model to adjust the output of the pre-trained model based on Platt-scaling, while others use Gaussian process [241] or maximum mean discrepancy [47]. However, an auxiliary model with enough capacity will always be able to recalibrate, even if the predicted uncertainty is completely uncorrelated with the real uncertainty [142]. Recently, [147] extended the definition of calibration where a regressor is well calibrated if the predicted error is equal to the difference between the ground truth and the predicted mean. Using this definition, [142] proposed unbiasing the predicted error by optimizing a scaling factor in the post-processing step. However, such methods often add overhead to an already slow model training phase.

7.4 Methodology: Likelihood Annealing

Our framework called Likelihood Annealing (LIKA) belongs to the family of models that are designed to predict a distribution for the outputs [120, 142, 128, 264, 261, 267] and the model is trained via a loss function derived from maximum likelihood estimation (MLE). We describe the problem formulation and related methods along with their limitations in Section 7.4.1. We present LIKA that constructs temperature-dependent likelihood to learn faster, better-calibrated regression uncertainty in Section 7.4.2, and analyze the effects of temperature annealing in Section 7.4.3.

7.4.1 Background and Motivation

Let $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{i=N}$ be the dataset that comprises of samples from domain $\mathbf{X}$ and $\mathbf{Y}$ (i.e., $\mathbf{x}_i \in \mathbf{X}, \mathbf{y}_i \in \mathbf{Y}, \forall i$), where $\mathbf{X}, \mathbf{Y}$ lies in $\mathbb{R}^m$ and $\mathbb{R}^n$, respectively. The goal of a regression task is to learn a function $\mathbf{\Psi}(\cdot; \theta) : \mathbb{R}^m \rightarrow \mathbb{R}^n$ (parameterized by θ) that maps the input $\mathbf{x}$ to the output $\mathbf{y}$. Let $\mathbf{y}_i := \mathbf{\Psi}(\mathbf{x}_i; \theta)$ be the estimate for the $\mathbf{y}_i$ and $\epsilon_i := \mathbf{y}_i - \mathbf{y}_i$ be the residual between the prediction and the ground-truth. The optimal parameters (θ^*) are learned by minimizing the error (e.g., ℓ_1 or ℓ_2 loss) between the prediction and ground truth using the labeled dataset. The ℓ_1/ℓ_2 loss function to train regression models originate by treating the residuals (i.e., ϵ_i) as following the i.i.d Laplace/Gaussian distribution. However, the i.i.d

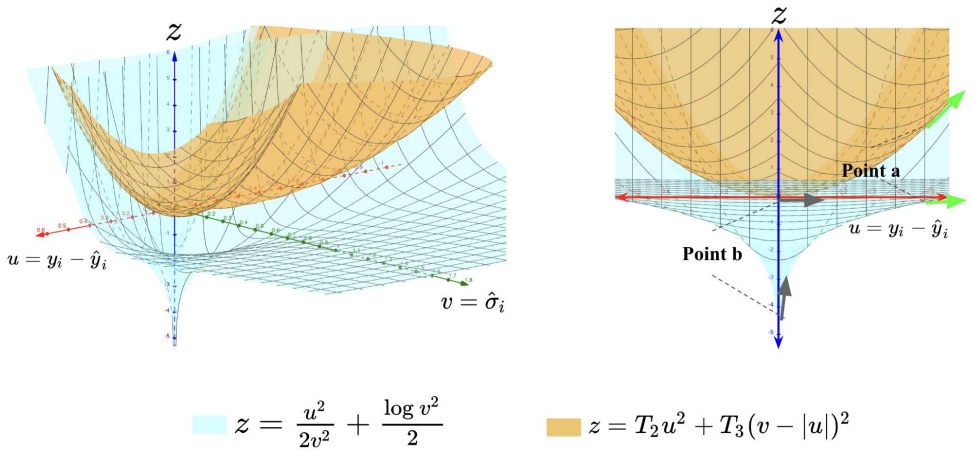

Figure 7.1: (Left) Objective function based on negative log-likelihood of standard heteroscedastic Gaussian distribution (**blue**) and temperature-dependent regularizer (**orange**) from Equation 7.4 as a function of residual and the estimated standard deviation. (Right) The 2D plot showing surfaces for a fixed predicted variance. The error and predicted variance are high at the beginning of the learning phase. The gradient of the temperature-dependent regularizer is higher (**orange**) than the gradient for the standard objective (**blue**), see **Point a** on both curves. Towards the end of training (with small error, predicted variance, and low temperatures), the objective from Equation 7.4 is dominated by the negative log-likelihood of standard heteroscedastic Gaussian with non-zero gradients. While gradients from the regularizer are zero, see **Point b**.

assumption will not capture the heteroscedasticity, and will allow uncertainty estimation with the limiting assumption of identical, i.e., homoscedastic, uncertainty values.

To estimate the uncertainty, the existing works [120] relax the i.i.d assumption and learn to model the heteroscedasticity as well. Such models are learned by maximizing the likelihood. Assuming that residuals follow Gaussian distribution, i.e., $\epsilon_i \sim \mathcal{N}(0, \hat{\sigma}_i)$, the likelihood, $P(\mathcal{D}|\theta)$, is a factored Gaussian distribution, $P(\mathcal{D}|\theta) = \prod_{i=1}^{i=N} \frac{1}{\sqrt{2\pi\hat{\sigma}_i^2}} \exp(-\frac{|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2}{2\hat{\sigma}_i^2})$. the MLE estimates for the parameters are obtained by minimizing the negative-log likelihood,

$$-\log P(\mathcal{D}|\theta) = \sum_{i=1}^{i=N} \frac{\log \hat{\sigma}_i^2}{2} + \frac{|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2}{2\hat{\sigma}_i^2} + Const. \tag{7.1}$$

The DNN is modified to output both the prediction (i.e., the mean of Gaussian) as well as the uncertainty estimate (i.e., the variance of Gaussian) learned using the above equation, i.e., $\mathbf{\Psi}(\mathbf{x}_i; \theta) = \{\hat{\mathbf{y}}_i, \hat{\sigma}_i\}$. While this method allows predicting the uncertainty estimates in single forward pass post training, it has several downsides, as discussed in the following. The blue surface in Figure 7.1-(Left) shows the loss from Equation 7.1 (which is derived by taking the negative log of Gaussian likelihood). It consists of two variables: the residual $\mathbf{y}_i - \hat{\mathbf{y}}_i$ (denoted by u) and the standard deviation $\hat{\sigma}_i$ (denoted by v). At the beginning of the training phase, the residual between the prediction and the ground truth is large along with significantly large predicted variance. Still, the corresponding gradient at that point is small (see **Point a** on the blue curve in Figure 7.1-(Right)), leading to slower convergence towards optima. As the learning progresses, the residual between prediction

and ground truth reduces substantially and so does the predicted variance, which leads to very high gradients potentially causing gradient explosion, a phenomenon often observed in practice (see **Point b** on the blue curve in Figure 7.1-(Left)). Together, this leads to slower model convergence as gradients, in the beginning, are too small. At the same time, the learning rate would also have to be substantially smaller to avoid gradient explosion later. Moreover, the works in [142, 147, 199] have shown that this method requires an additional post hoc calibration phase to tackle miscalibration.

7.4.2 Constructing Temperature Dependent *Improper* Likelihood

To tackle the slow convergence issue while providing well-calibrated uncertainty estimates, we formulate a temperature-dependent likelihood function that facilitates faster convergence with the help of temperature annealing. Our formulation imposes an explicit condition on the uncertainty estimates, keeping them calibrated throughout the learning phase, leading to calibrated uncertainty estimates without any post-hoc calibration phase. While Equation 7.1 denotes the negative log-likelihood for the standard Gaussian distribution, We formulate a new improper likelihood distribution on the network output given by,

$$P(\mathcal{D}|\theta) = \prod_{i=1}^{i=N} \frac{e^{\frac{-|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2}{(2\hat{\sigma}_i^2)}}}{\sqrt{2\pi\hat{\sigma}_i^2}} \times e^{-T_2(|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2)} \times e^{-T_3 \left\{ \begin{array}{l} |\hat{\mathbf{y}}_i - (\mathbf{y}_i + \hat{\sigma}_i)|^2, \hat{\mathbf{y}}_i \geq \mathbf{y}_i \\ |\hat{\mathbf{y}}_i - (\mathbf{y}_i - \hat{\sigma}_i)|^2, \hat{\mathbf{y}}_i < \mathbf{y}_i \end{array} \right\}} \tag{7.2}$$

Where, T_2, T_3 are hyper-parameters that we refer to as temperature. We then use the improper maximum likelihood estimator, as also used in [44, 43, 2] to derive an objective function. We do this by taking the negative log of improper likelihood from Equation 7.2, leading to the following objective (*omitting the constants for clarity and simplification*):

$$\sum_{i=1}^{i=N} \frac{\log \hat{\sigma}_i^2}{2} + \frac{|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2}{2\hat{\sigma}_i^2} + T_2(|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2) + T_3 \left\{ \begin{array}{l} |\hat{\mathbf{y}}_i - (\mathbf{y}_i + \hat{\sigma}_i)|^2, \hat{\mathbf{y}}_i \geq \mathbf{y}_i \\ |\hat{\mathbf{y}}_i - (\mathbf{y}_i - \hat{\sigma}_i)|^2, \hat{\mathbf{y}}_i < \mathbf{y}_i \end{array} \right\}. \tag{7.3}$$

We note that the above equation can be re-written as,

$$\sum_{i=1}^{i=N} \frac{\log \hat{\sigma}_i^2}{2} + \frac{|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2}{2\hat{\sigma}_i^2} + T_2(|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2) + T_3(|\hat{\sigma}_i - |\hat{\mathbf{y}}_i - \mathbf{y}_i||^2). \tag{7.4}$$

Equation 7.4 has two additional terms (i.e., $T_2(|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2)$ and $T_3(|\hat{\sigma}_i - |\hat{\mathbf{y}}_i - \mathbf{y}_i||^2)$) compared to Equation 7.1. To understand the effects of our proposed temperature-dependent improper likelihood, we first, look at the newly introduced temperature-dependent regularizers, represented by $\mathcal{L}_{\text{reg}}$ given by,

$$\mathcal{L}_{\text{reg}} = T_2(|\hat{\mathbf{y}} - \mathbf{y}|^2) + T_3(|\hat{\sigma} - |\hat{\mathbf{y}} - \mathbf{y}||^2). \tag{7.5}$$

Figure 7.1-(Left) shows the surface corresponding to $\mathcal{L}_{\text{reg}}$ in **orange** for substantially large temperature values. We notice that at the beginning of the training phase, with temperature hyper-parameters set to high values, Equation 7.4 is dominated by $\mathcal{L}_{\text{reg}}$. As shown in Figure 7.1-(Right), the corresponding gradient at the beginning of the training (dominated by $\mathcal{L}_{\text{reg}}$) is much higher (see **Point a** on the orange curve). This encourages faster convergence at the beginning of the training phase, unlike the Equation 7.1.

To further understand the effects of the newly introduced temperature-dependent regularizers, we look at the conceptual schematic, shown in Figure 7.2, that illustrates the soft constraint imposed by the regularizers, represented by $\mathcal{L}_{\text{reg}}$. As discussed above, we propose to start with high values for the T_2 and T_3 hyper-parameters and gradually decrease them during the course of training. We observe that at high temperatures (i.e., at the beginning of the training phase), the objective function from Equation 7.4 is dominated by the last two terms that are controlled by T_2 and T_3. We show these two terms (i.e., $\mathcal{L}_{\text{reg}}$) in Figure 7.2 as a function of $\hat{\mathbf{y}}$ for a fixed ground truth $\mathbf{y}$ and a fixed $\hat{\sigma}$, and note that minimizing $\mathcal{L}_{\text{reg}}$ encourages the prediction $\hat{\mathbf{y}}$ to be close to the ground truth $\mathbf{y}$, while also ensuring that the discrepancy between the prediction and ground truth $|\hat{\mathbf{y}} - \mathbf{y}|^2$ is close to the predicted variance $\hat{\sigma}^2$, encouraging calibration of the predicted variance without the need of post-hoc techniques (orange **bold** curve in Figure 7.2).

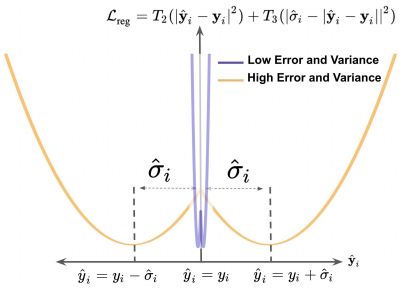

Figure 7.2: Schematic of the temperature-dependent regularizer characterized by $\{\mathbf{y}, \hat{\mathbf{y}}, \hat{\sigma}\}$. This enforces the prediction to be close to ground truth and the uncertainty estimate to be close to the error, i.e., calibrated (shown in **orange**). When the predicted variance is small, all the optimums come close to each other (shown in **blue**).

Moreover, as the training progresses and the temperature decreases, $\hat{\mathbf{y}}$ comes closer to $\mathbf{y}$ and the predicted variance $\hat{\sigma}$ also decreases, we notice that this leads to the local optimums coming closer, and eventually collapsing at $\mathbf{y} = \mathbf{y}$ in the limit (blue **bold** curve in Figure 7.2). Throughout the early phase of training (with high temperature), the regularizer encourages the prediction $\hat{\mathbf{y}}$ to be close to ground truth $\mathbf{y}$ and the predicted variance $\hat{\sigma}^2$ to be close to error $|\hat{\mathbf{y}} - \mathbf{y}|^2$. This way, the regularizer imposes a soft constraint for calibration of the predicted uncertainty estimate throughout the training.

7.4.3 Effects of Temperature Annealing

The temperature-dependent improper likelihood in Equation 7.2 leads to objective in Equation 7.4 that allows us to control the contribution of individual terms by changing the temperature hyper-parameters T_2, T_3. As described in Section 7.4.2, annealing the

temperature hyperparameters allow faster convergence of the uncertainty-aware regression with better-calibrated uncertainty methods. We start by initializing T_2, T_3 with a high value of 100 and progressively reduce them according to the training epochs using exponential annealing – referred to as *temperature annealing*. At higher temperatures, the

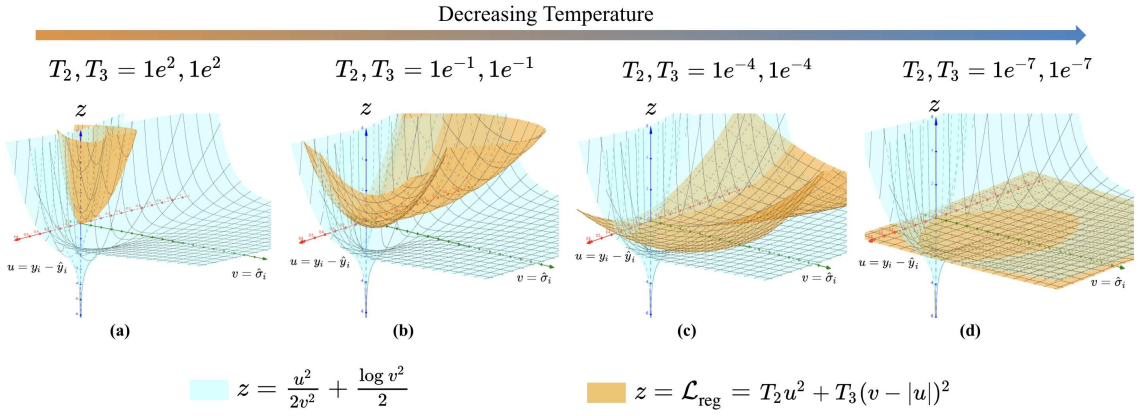

Figure 7.3: Effects of temperature annealing. As we anneal the temperature in Equation 7.4, the proposed temperature-dependent regularizer $\mathcal{L}_{\text{reg}}$ from Equation 7.5 (shown in **orange**) gradually changes from (a), (b), (c) to (d), which provides faster convergence at the beginning of training while ensuring convergence to the same optima as the standard objective function as described in Equation 7.1 (shown in **blue**).

overall objective is dominated by the temperature-dependent terms ($\mathcal{L}_{\text{reg}}$). Figure 7.3-(a)) shows the loss surface for the negative log-likelihood derived from standard Gaussian (i.e., Equation 7.1) and the newly introduced temperature-based regularize $\mathcal{L}_{\text{reg}}$. As the temperatures decrease, the overall loss is close to the standard loss function. This can also be seen from Figure 7.3-(b,c), where the surface corresponding to $\mathcal{L}_{\text{reg}}$ flattens out at lower temperature, eventually coming close to plane surface as temperatures approach 0 as shown in Figure 7.3-(d)). Note that, when temperatures are zero $\mathcal{L}_{\text{reg}} = 0$ and Equation 7.4 reduces to Equation 7.1. This dynamic contribution from different terms allows the network to converge faster in the beginning (as gradients from the temperature-dependent loss terms are higher than the standard loss term), and ensures stable convergence to the same optima as the standard loss, thus leading to faster, better-calibrated uncertainty.

7.4.4 Normalizing the improper Likelihood

We further study our proposed improper likelihood (presented at Equation 7.2) and convert it to proper likelihood by normalizing Equation 7.2. Let the normalizing constant be Z_i. Then the proper likelihood is,

$$P(\mathcal{D}|\theta) = \prod_{i=1}^{i=N} Z_i e^{\frac{-|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2}{(2\hat{\sigma}_i^2)}} \times e^{-T_2(|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2)} \times e^{-T_3 \left\{ \begin{array}{l} |\hat{\mathbf{y}}_i - (\mathbf{y}_i + \hat{\sigma}_i)|^2, \hat{\mathbf{y}}_i \geq \mathbf{y}_i \\ |\hat{\mathbf{y}}_i - (\mathbf{y}_i - \hat{\sigma}_i)|^2, \hat{\mathbf{y}}_i < \mathbf{y}_i \end{array} \right\}} \tag{7.6}$$

Where, $Z_i = \frac{2\sqrt{\pi}\hat{\sigma}\exp\left(-\frac{\hat{\sigma}^2 T_3\left(2\hat{\sigma}^2 T_2+1\right)}{2\hat{\sigma}^2(T_2+T_3)+1}\right)\left(\operatorname{erf}\left(\frac{2\hat{\sigma}^2 T_3}{\sqrt{4\hat{\sigma}^2(T_2+T_3)+2}}\right)+1\right)}{\sqrt{4\hat{\sigma}^2(T_2+T_3)+2}}$. The NLL of Equation 7.6 leads to,

$$\mathcal{L}_{\text{norm}} = \sum_{i=1}^{i=N} -\left(\frac{\hat{\sigma}_i^2 T_3\left(2\hat{\sigma}_i^2 T_2+1\right)}{2\hat{\sigma}_i^2\left(T_2+T_3\right)+1}\right) + \log\left(\operatorname{erf}\left(\frac{2\hat{\sigma}_i^2 T_3}{\sqrt{4\hat{\sigma}_i^2\left(T_2+T_3\right)+2}}\right)+1\right) - 0.5\log\hat{\sigma}_i^2 + \frac{|\hat{\mathbf{y}}_i-\mathbf{y}_i|^2}{2\hat{\sigma}_i^2} + T_2(|\hat{\mathbf{y}}_i-\mathbf{y}_i|^2) + T_3(|\hat{\sigma}_i-|\hat{\mathbf{y}}_i-\mathbf{y}_i||^2) \tag{7.7}$$

7.5 Experiments

We first provide a detailed description of our experimental setup, including the datasets used for training and evaluation, the evaluation metrics employed to assess the performance of our model in Section 7.5.1. We compare our model to a wide variety of state-of-the-art methods quantitatively and qualitatively in Section 7.5.2. Finally, we also provide an ablation analysis in Section 7.5.2 to study the rationale of our model formulation.

7.5.1 Experimental Setup

Datasets and Tasks. We conduct experiments on five datasets (three small scale problems, two large scale problems) to solve the regression task and provide uncertainty estimation. We choose the following three low-dimensional regression problems. They highlight the different complexities and network architectures that are required to solve them. In *Chaotic System using Lorenz Attractor* (referred to as *Lorenz Attractor*), the Lorenz equations describe non-linear chaotic systems given by, $\frac{\partial z_1}{\partial t} = 10(z_2 - z_1)$, $\frac{\partial z_2}{\partial t} = z_1(28 - z_3) - z_2$, $\frac{\partial z_3}{\partial t} = z_1 z_2 - 8z_3/3$. Similar to [72], to generate a trajectory we run the Lorenz equations with a $\partial t = 10^{-5}$ from which we sample with a time step of $t = 0.05$. Each point is then perturbed with Gaussian noise of standard deviation 0.5 to produce pairs of noisy and clean trajectories split into non-overlapping train/validation/test sets. We use a 1D CNN to map the noisy input to clean output. The *Physical Properties of Molecules (Atom3D)* [255] is a 3D molecular structure dataset aiming to predict the physical property such as the dipole moment given the 3D atomistic representation. We use the standard Graph Neural Network (GNN) for this task. The *House Price Prediction (Boston-housing)* [91, 15] dataset is used to predict the house prices using various attributes using Multi Layer Perceptrons (MLPs).

To show the generalization of our method to high-dimensional regression problems, we use the following two datasets. In *Super-resolution of Natural Images (Super-resolution)*, we learn mapping from low-resolution to high-resolution images using CNNs, using DIV2K dataset [254, 103]. We do 4x downsampling to create the corresponding low-resolution

images. The dataset is split into 800/100/100 images for training/val/test sets. In *Medical Image Translation (MRI Translation)*, We translate one imaging modality to another, i.e., T1 MRI to T2 MRI images. As T1 and T2 MRI from the same patient in the same orientation are often not available and T2 takes longer to acquire, learning a mapping from T1 to T2 is desirable. As in [261], we use T1 and T2 MRI of 500 patients from IXI dataset [214] (200/100/200 for training/val/test) in a 2D CNN based on U-Net [218].

Evaluation Metrics. To measure the quality of regression output, we adopt the standard metrics: mean absolute error (`MAE`) and mean square error (`MSE`). In addition, for the super-resolution and medical image translation tasks, we use `PSNR` and `SSIM` to measure the structural similarity between two images [289]. To measure the quality of uncertainty estimates ($\hat{\sigma}^2$), we compute (i) the correlation coefficient (`Corr. Coeff.`) between uncertainty estimates ($\hat{\sigma}^2$) and the error ($|\hat{\mathbf{y}} - \mathbf{y}|^2$). (ii) *Uncertainty calibration error* (`UCE`) for regression tasks [142, 147]. Following [87], the uncertainty output $\hat{\sigma}^2$ of a deep model is partitioned into M bins with equal width (each represented by B_m for $\forall m \in \{1, 2...M\}$). A weighted average of the difference between the predictive error and uncertainty is used, UCE $= \sum_{m=1}^{M} \frac{|B_m|}{N} |\text{err}(B_m) - \text{uncer}(B_m)|$. Where, $\texttt{err}(B_m) := \frac{1}{|B_m|} \sum_{i \in B_m} ||\hat{\mathbf{y}}_i - \mathbf{y}_i||^2$ and $\texttt{uncer}(B_m) := \frac{1}{|B_m|} \sum_{i \in B_m} \hat{\sigma}_i^2$. (iii) UCE for the re-calibrated uncertainty estimates (`R.UCE`). We use post-hoc calibration technique introduced in [142], called σ-scaling, that optimizes for the scaling factor (s), post training to produce uncertainty estimates ($\hat{\sigma}^2$) and predictions ($\hat{\mathbf{y}}$) using, $s^* = \underset{s}{\text{argmin}} \left[N \log(s) + \frac{1}{2s^2} \sum_{i=1}^{N} \frac{|\hat{\mathbf{y}}_i - \mathbf{y}_i|^2}{\hat{\sigma}_i^2} \right]$. In addition, we present the (iv) *expected calibration error* (`ECE`) and (v) *sharpness* (`Sharpness`). While ECE is another metric to quantify the calibration of the uncertainty estimates, one must note that it may be possible to have an uninformative, yet average calibrated model [38, 336]. Therefore it is necessary to also present the Sharpness metric that encourages more-concentrated distributions. Finally, we present the (vi) predictive log-likelihood that assesses how well the predicted conditional distribution fits the data.

Implementation Details. Our LIKA method is generalizable across different types of architectures. Here we perform experiments with MLPs, 1D/2D CNNs, and GNNs. We take the well-established networks for the respective problems and modify them to produce the uncertainty estimates as described in [120, 246]. All the networks were trained using Adam optimizer [122]. The initial learning rate was set to $2e^4$ and cosine annealing was used to decay the learning rate over the course of the learning phase. The hyperparameters, (T_2, T_3) (Equation 7.4) were set to $(100, 100)$ and scheduled to exponentially decay over the course of the training. We provide the code in the supplementary.

7.5.2 Comparing to Uncertainty Estimation Methods

Compared methods. For each of the regression tasks, we compare our model (LIKA and its derivatives like Ens-LIKA, DO-LIKA, LIKA-Norm) to eight representative state-of-the-art methods for uncertainty estimation using DNNs for regression tasks, belonging to a diverse class of methods, i.e. Bayesian ensemble, test-time data augmentation, maximum

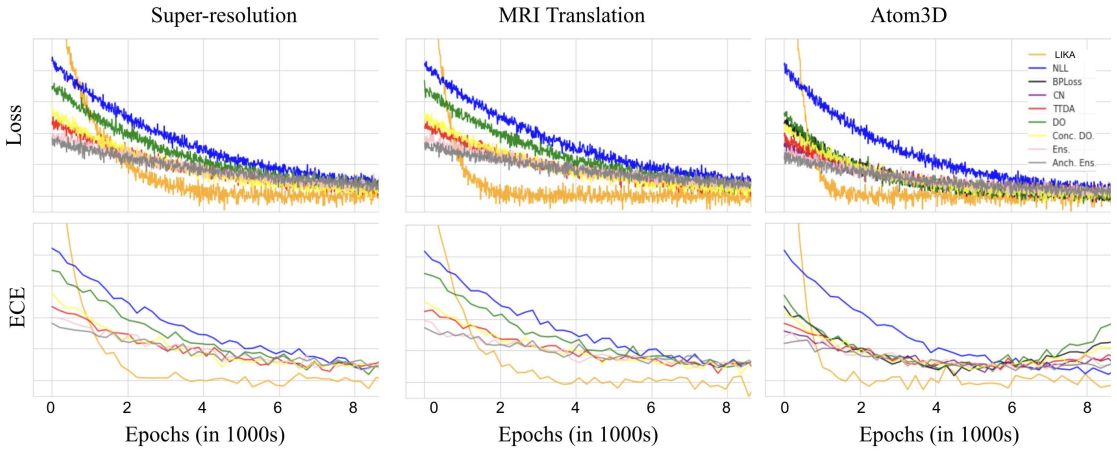

Figure 7.4: Plots comparing the required convergence time (number of epochs to converge) for different methods and corresponding ECE during the training on (i) Super-resolution, (ii) MRI translation, (iii) Atom3D.

likelihood and variants of the same, and finally quantile regression methods. In addition, we evaluate LIKA-Norm for some of our experiments. This method uses proper likelihood-based objective to train the network given by Equation 7.6.

Maximum likelihood methods: In this method (NLL) [120, 246] the network is modified to predict the mean and variance and then trained by optimizing negative log-likelihood. The variance head then provides uncertainty estimates for the prediction at the inference time. We also evaluate [245] (called NLL-FH) that uses a modified objective instead of the NLL of heteroscedastic Gaussian, using a backbone architecture similar to NLL (and other methods in this work), with the head split to predict both mean and variance as [120].

Test-time Data Augmentation Methods: In Test Time Data Augmentation (TTDA) [278, 9, 75] multiple perturbed copies of the input are passed through a deterministic network to estimate the predictive uncertainty at the inference stage.

Ensemble Methods: In Deep Ensemble (Ens) [138] multiple deterministic networks are trained to make the final prediction with uncertainty estimates. While the above estimates epistemic uncertainty, to capture the aleatoric uncertainty, We also evaluate Ens-NLL, which is an ensemble of 5 similar models except for the head of each model in the ensemble is split into two to predict both the mean and variances using the Gaussian-NLL loss. Each ensemble model is trained independently, with different weight initializations. The aleatoric uncertainty considered in evaluation of Ens-NLL is the mean of variance head for all the models in the ensemble. Similarly, we create an ensemble of LIKA (Ens-LIKA) for the evaluation.

Bayesian methods: In (DO) [70] the weights of the neural network are randomly dropped at training and inference time. Multiple forward passes for the same input at inference time allow us to estimate the uncertainty. While the above methods only consider the epistemic uncertainty, we also evaluate DO-NLL, which is similar to DO, except the head is split into two to predict both the mean and variances using the Gaussian-NLL loss

function, along with dropouts during training and evaluation. For DO-NLL, we consider the aleatoric uncertainty for evaluation obtained as the mean of variance head outputs at evaluation for a single sample with 100 forward passes and dropouts activated. Similarly, we create DO-LIKA for the evaluation.

Quantile Regression Methods: In Calibrated Quantile Regression Method (BPLoss) [38] proposes a model that specifies the full quantile function for the predictions and achieves a balance between calibration and sharpness. In Collaborating Networks for estimating uncertainty intervals (CN) [336] two networks are trained simultaneously, one to estimate the cumulative distribution function, and the other approximates its inverse. We note that some baseline methods (i.e., BPLoss and CN) have only been proposed for low-dimensional regression settings (where the output of a model is single scalar) and it is non-trivial and inefficient to scale it to high-dimensional regression settings (e.g., image translation, where the output for an input is a high-dimensional matrix/tensor). Therefore such models are compared only on low-dimensional regression tasks where they are applicable.

Quantitative results on convergence. In this experiment, we train different models to perform the different kinds of regression task and keep track of the training and validation loss to identify if the model has converged. For all the models we used the same optimizer (i.e., Adam [122]) with the same initial learning rate (i.e., `lr`$= 2 \times 10^{-4}$) and identical decaying schedule (i.e., cosine annealing for `lr`).

We observe in Figure 7.4 that the baseline methods consistently take longer time to converge while our proposed method (LIKA) consistently has faster convergence. For instance, on the super-resolution task, our method takes about 4,000 epochs to converge while the other baseline methods consistently take longer than 8000 epochs to converge. In particular, the NLL baseline takes the longest to converge. We also note that in the early phase of training, our LIKA has much higher loss, this is due to the additional temperature dependent loss terms (in Equation 7.4) that contribute to the overall loss. However, the higher values of the temperature T_2 and T_3 in the beginning of the training phase also allow faster convergence, as explained in Section 7.4. Moreover, towards the end of the training phase, the temperature parameters are annealed to a low value (close to zero) and the over all loss function reduces to a low value.

Figure 7.4 (second row) shows the evolution of ECE for the derived uncertainty using various methods during the training. Again we see that our LIKA achieves the lowest ECE much faster than the other methods. A similar trend is observed for the other datasets. For example, on Atom3D dataset, the proposed method converges at about 2000 epochs, much faster than other baselines, similarly, it achieves the lowest ECE much faster than other methods. These results show that our method converges much faster than the other methods, which is in line with our motivation to ensure a faster convergence for the regression uncertainty model along with better-calibrated uncertainty as described in Section 7.4.3.

Quantitative results on regression and uncertainty. Uncertainty-aware regression models must be evaluated on two fronts which are (i) the regression performance, i.e.,

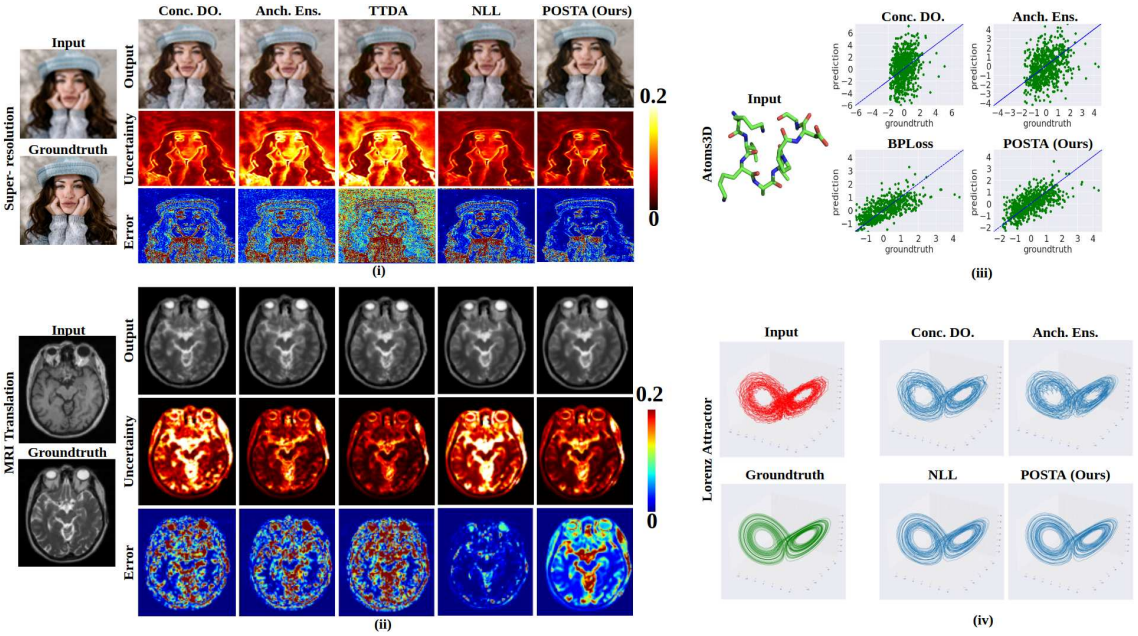

Figure 7.5: Qualitative results: input, predictions, groundtruth, and the error.

the quality of the target predictions and (ii) the quality of estimated uncertainty (the uncertainty should be sharp and well calibrated). We evaluate the model performance based on two set of metrics: (1) task-specific metrics that evaluate the regression results using `MAE`, `MSE`, `PSNR`, `SSIM`, and (2) calibration-specific metrics that evaluate the quality of the uncertainty estimates using `C.Coeff.`, `UCE`, `R.UCE`, `ECE`, `Sharp.`, and `Log-likeli.` Table 7.1 shows the quantitative results that evaluate regression and the quality of uncertainty estimates for different methods on multiple regression tasks. Our LIKA method also obtains high quality regression outputs. In two tasks (including super-resolution, and MRI translation), our LIKA achieves the best or competitive performance compared to the other methods. We note that while no single metric can indicate the "goodness" of uncertainty estimates (as there is no groundtruth for uncertainty values), the collective set of metrics such as `C.Coeff.`, `UCE`, `R.UCE`, `ECE`, `Log-likeli`, `Sharp.` provide a holisitic indication of "goodness" of uncertainty metric. The proposed method, LIKA, consistently performs well in terms of the above metrics. Overall, these quantitative results show that our method performs well in providing both satisfactory regression and uncertainty estimates.

Qualitative results on regression and uncertainty. Figure 7.5 shows the regression output on different datasets. Figure 7.5-(i) & (ii) visualizes the generated images for image super-resolution and MRI translation tasks. While the other methods often generate relatively blurry images with artifacts in colours, our model produces better output visually more similar to the ground-truth. Moreover, Figure 7.5-(i) & (ii) also shows the uncertainty maps, along with the prediction and error for super-resolution and MRI translation. We observe that for compared methods, uncertainty maps do not always

T	Methods	Metrics									
		MAE ↓	MSE ↓	SSIM ↑	PSNR ↑	C.Coeff. ↑	UCE↓	R.UCE.↓	Log-likeli. ↑	ECE ↓	Sharp. ↓
Boston-housing	NLL [120]	2.663	10.75	-	-	0.107	12.67	12.23	-2.42	11.5	8.21
	NLL-FH [245]	**2.551**	**10.14**	-	-	0.103	13.63	13.11	-2.62	11.7	8.33
	TTDA [75]	2.584	10.30	-	-	0.007	14.32	13.85	-2.24	11.8	9.28
	Ens-NLL [138]	2.913	12.82	-	-	0.114	10.38	9.98	-2.33	10.1	8.27
	DO-NLL [120]	2.661	12.83	-	-	0.116	8.783	8.226	-2.21	9.32	8.48
	BPLoss [38]	2.684	11.49	-	-	0.237	9.216	8.837	-2.11	9.72	9.01
	CN [336]	2.594	11.13	-	-	0.213	10.84	9.722	-2.23	9.65	9.67
	DO [70]	2.851	13.26	-	-	0.014	10.76	10.18	-2.46	10.2	8.66
	Ens [138]	2.971	13.76	-	-	0.011	11.26	10.78	-2.41	10.3	8.87
	LIKA (ours)	2.593	10.51	-	-	**0.348**	**0.756**	0.637	**-2.06**	**6.37**	**8.22**
	LIKA-Norm (ours)	2.633	10.94	-	-	0.311	0.818	0.682	-2.08	6.87	9.14
	Ens-LIKA (ours)	2.774	10.83	-	-	0.343	0.768	0.648	-2.11	6.55	8.51
	DO-LIKA (ours)	2.643	10.76	-	-	0.345	0.762	**0.631**	-2.14	6.82	8.85
Atom3D	NLL [120]	**0.498**	**0.463**	-	-	0.164	3.358	3.335	-0.22	1.38	3.32
	NLL-FH [245]	0.507	0.582	-	-	0.112	4.468	4.112	-0.32	2.24	3.82
	TTDA [75]	0.903	1.301	-	-	0.157	4.167	3.988	-0.38	1.94	4.78
	Ens-NLL [138]	0.922	0.983	-	-	0.166	4.115	3.977	-0.22	1.66	4.02
	DO-NLL [120]	0.950	1.224	-	-	0.135	4.177	3.956	-0.22	1.92	4.11
	BPLoss [38]	0.527	0.873	-	-	0.189	3.527	3.166	-0.21	1.55	3.12
	CN [336]	0.521	0.845	-	-	0.087	4.311	2.971	-0.16	1.77	3.18
	DO [70]	1.950	5.828	-	-	0.085	5.380	5.054	-0.24	2.12	4.32
	Ens [138]	1.215	2.388	-	-	0.138	4.623	4.376	-0.23	1.69	4.17
	LIKA (ours)	0.513	0.495	-	-	0.567	**0.296**	**0.277**	-0.18	**1.37**	3.17
	LIKA-Norm (ours)	0.554	0.585	-	-	0.511	0.377	0.315	-0.20	1.68	3.92
	Ens-LIKA (ours)	0.502	0.536	-	-	**0.591**	0.316	0.281	**-0.15**	1.52	**3.07**
	DO-LIKA (ours)	0.535	0.574	-	-	0.573	0.324	0.286	-0.17	1.58	3.11
Lorenz Attractor	NLL [120]	0.172	0.048	-	31.28	0.588	2.368	1.933	-0.13	**4.33**	**7.87**
	NLL-FH [245]	**0.168**	**0.043**	-	32.27	0.593	2.377	2.145	-0.12	4.56	8.13
	TTDA [75]	1.391	3.764	-	29.16	0.438	3.325	3.077	-0.17	5.96	8.91
	Ens-NLL [138]	0.175	0.051	-	31.11	0.567	2.491	2.213	-0.14	4.46	7.93
	DO-NLL [120]	0.187	0.058	-	30.87	0.536	2.564	2.277	-0.15	4.53	8.10
	DO [70]	1.373	3.463	-	29.85	0.281	2.864	2.134	-0.16	4.34	5.67
	Ens. [138]	2.544	11.65	-	24.32	0.778	6.726	6.294	-0.22	10.4	8.43
	LIKA (ours)	**0.153**	**0.029**	-	**32.33**	**0.821**	**0.779**	**0.356**	**-0.11**	4.36	9.12
	LIKA-Norm (ours)	0.286	0.039	-	30.14	0.561	0.922	0.414	-0.12	4.83	9.37
	Ens-Norm (ours)	0.164	0.032	-	31.66	0.801	0.833	0.372	-0.12	4.43	9.22
	DO-Norm (ours)	0.175	0.035	-	31.37	0.787	0.862	0.394	-0.12	4.69	9.31
Super-resolution	NLL [120]	0.693	0.414	0.955	37.15	0.189	0.581	0.512	-0.36	1.45	2.73
	NLL-FH [245]	0.671	0.394	0.958	37.33	0.195	0.531	0.491	-0.33	1.22	2.26
	TTDA [75]	0.883	0.691	0.939	34.94	0.047	1.175	0.994	-0.39	11.3	10.3
	Ens-NLL [138]	0.721	0.468	0.947	36.82	0.182	0.634	0.544	-0.38	1.95	3.36
	DO-NLL [120]	0.744	0.493	0.941	36.22	0.178	0.661	0.553	-0.39	2.11	3.64
	DO [70]	0.832	0.548	0.947	35.64	0.033	0.748	0.519	-0.38	4.67	6.32
	Ens. [138]	0.793	0.462	0.953	36.61	0.029	0.941	0.733	-0.36	8.76	10.2
	LIKA (ours)	**0.618**	**0.351**	**0.962**	**37.87**	**0.518**	**0.104**	**0.053**	**-0.16**	**0.74**	**0.83**
	LIKA-Norm (ours)	0.691	0.418	0.943	36.62	0.186	0.388	0.193	-0.19	1.15	1.53
	Ens-LIKA (ours)	0.663	0.392	0.955	36.88	0.447	0.192	0.081	-0.18	0.95	0.93
	DO-LIKA (ours)	0.672	0.399	0.951	36.52	0.458	0.185	0.076	-0.17	0.92	0.89
MRI Translation	NLL [120]	0.632	0.582	0.938	34.34	0.134	1.673	1.448	-0.28	4.03	5.12
	NLL-FH [245]	0.591	0.511	0.947	34.91	0.196	1.611	1.142	-0.29	4.15	5.33
	TTDA [75]	0.755	0.729	0.904	32.18	0.128	1.483	1.153	-0.37	7.21	9.74
	Ens-NLL [138]	0.644	0.611	0.931	33.92	0.142	1.688	1.466	-0.30	4.23	5.65
	DO-NLL [120]	0.648	0.627	0.924	33.24	0.139	1.714	1.535	-0.30	4.29	5.89
	DO [70]	0.732	0.683	0.912	32.45	0.159	0.864	0.771	-0.33	4.48	6.23
	Ens. [138]	0.681	0.611	0.927	33.76	0.110	1.143	0.974	-0.36	4.86	7.21
	LIKA (ours)	0.615	0.537	0.946	35.27	**0.432**	**0.098**	**0.062**	-0.30	**3.26**	5.78
	LIKA-Norm (ours)	0.687	0.613	0.935	34.33	0.266	0.158	0.088	-0.31	4.36	5.84
	Ens-LIKA (ours)	**0.598**	**0.502**	**0.949**	**35.31**	0.416	0.138	0.071	**-0.26**	4.38	**5.03**
	DO-LIKA (ours)	0.655	0.597	0.942	35.22	0.405	0.161	0.092	-0.30	4.39	5.89

Table 7.1: Evaluating different methods on five datasets using `MAE`, `MSE`, `PSNR`, `SSIM` (where applicable, to evaluate regression) and `C.Coeff.`, `UCE`, `R.UCE`, `Log-Likeli.`, `ECE`, `Sharp.` (to measure quality of uncertainty estimates). ↑/↓ indicates higher/lower is better. "T": tasks. Best results are in bold.

agree with error maps at pixel level (i.e., higher/lower uncertainty than the corresponding error), whereas our uncertainty maps are in agreement with the errors. This suggests

that our model provides better-calibrated uncertainty. Figure 7.5-(iii) shows the plots for predictions vs ground-truth on the Atom3D dataset. We can see that compared to other methods, our method yields predictions much closer to the ground-truth e.g., on the Atom3D dataset, our method produces regression output more highly correlated with the ground-truth. Figure 7.5-(iv) shows the input noisy trajectory, denoised output and the corresponding ground-truth for the Lorentz attractor dataset. We can see that compared to other methods, our method yields smoother trajectories.

7.5.3 Ablation Analysis of Annealing

Table 7.2 shows the ablation study of two temperature hyperparameters in our formulated temperature-dependent likelihood (Equation 7.2) along with different choices of priors for the super-resolution task.

We test the baseline that removes both temperature-dependent terms (i.e. $T_2 = T_3 = 0$) with a uniform prior, this is equivalent to the NLL method and is shown in the first row (`MAE` of 0.693). We then study the effect of fixing one of the temperatures at a non-zero value while setting the other temperature to 0. With $T_2 = 100, T_3 = 0$, we see a slight improvement in regression performance (`MAE` of 0.614 vs. 0.693) and much poorer performance with respect to uncertainty calibration (`UCE` of 1.169 vs. 0.581), this is due to more weighting of fidelity term between the prediction and the ground-truth along with suppression of the default calibration effect of NLL. On the other hand, $T_2 = 0, T_3 = 100$ suppresses the default fidelity term for NLL, therefore the output is of significantly worse quality (poor regression scores, `MAE` of 1.395 vs 0.693) this further degrades the quality of the uncertainty estimates (`UCE` 3.733 vs 0.581). We notice that if the model does not perform good regression, the quality of uncertainty estimate is also adversely effected.

We then study the effects of decaying one of the temperatures while setting other to 0. With T_2 decaying (i.e., $T_2 = \downarrow, T_3 = 0$) we see slightly better performance than $T_2 = 100, T_3 = 0$ (`MAE` of 0.612 vs. 0.614 and `UCE` of 0.983 vs. 1.169), whereas with T_2 decaying (i.e., $T_2 = 0, T_3 = \downarrow$) we see good regression performance but also an improved calibration performance (`UCE` of 0.152 vs. 0.581). With both the parameters decaying (i.e., $T_2 = \downarrow, T_3 = \downarrow$) we achieve improved regression and calibration results concluding that annealing works the best. In addition to uniform prior setup (i.e., $P(\theta) = \mathcal{U}(\theta)$), we evaluate two other priors (i) Gaussian prior on the parameters of the network, i.e., $P(\theta) = \mathcal{N}(\theta)$ that is equivalent to ℓ_2 regularization of weights and (ii) Laplace prior, i.e., $P(\theta) = \mathcal{E}(\theta)$ that is equivalent to ℓ_1 regularization of weights. With Gaussian/Laplace prior we achieve `MAE` of 0.625/0.612 showing that carefully crafted priors may further boost the performance, designing such priors will be explored in future works.

7.5.4 Evaluation on Out-of-Distribution Data

Previous works have studied the performance of various uncertainty-aware methods in the presence of out-of-distribution (OOD) samples at the inference time [193, 95, 183, 180].

Methods	Metrics									
	MAE ↓	MSE ↓	PSNR ↑	SSIM ↑	C.Coeff. ↑	UCE↓	R.UCE.↓	Log-likeli. ↑	ECE ↓	Sharp. ↓
$T_2 = 0, T_3 = 0$	0.693	0.414	37.15	0.955	0.189	0.581	0.512	-0.36	1.45	2.73
$T_2 = 10, T_3 = 10$	0.667	0.396	37.33	0.958	0.184	0.577	0.503	-0.31	1.42	2.24
$T_2 = 100, T_3 = 0$	0.614	0.384	37.72	0.961	0.062	1.169	0.833	-0.41	1.77	2.82
$T_2 = 0, T_3 = 100$	1.395	7.274	20.19	0.793	0.219	3.733	2.442	-0.44	2.12	3.11
$T_2 = 100\downarrow, T_3 = 0$	0.612	0.344	37.76	0.961	0.077	0.983	0.797	-0.27	1.03	1.35
$T_2 = 0, T_3 = 100\downarrow$	0.632	0.388	37.71	0.960	0.442	0.152	0.116	-0.20	0.85	0.98
$T_2 = 100\downarrow, T_3 = 100\downarrow$	0.618	**0.351**	37.87	0.962	**0.518**	**0.104**	**0.083**	-0.16	**0.74**	**0.83**
$T_2 = 100\downarrow, T_3 = 100\downarrow$ with $P(\theta) = \mathcal{N}(\theta)$	0.625	0.358	36.98	0.952	0.488	0.168	0.133	-0.24	1.12	1.47
$T_2 = 100\downarrow, T_3 = 100\downarrow$ with $P(\theta) = \mathcal{E}(\theta)$	**0.612**	0.353	**37.92**	**0.966**	0.503	0.118	0.102	**-0.15**	0.83	1.01

Table 7.2: Ablation study of temperature hyperparameters of the temperature-dependent likelihood used in the proposed *likelihood annealing* (LIKA) method on image super-resolution task.

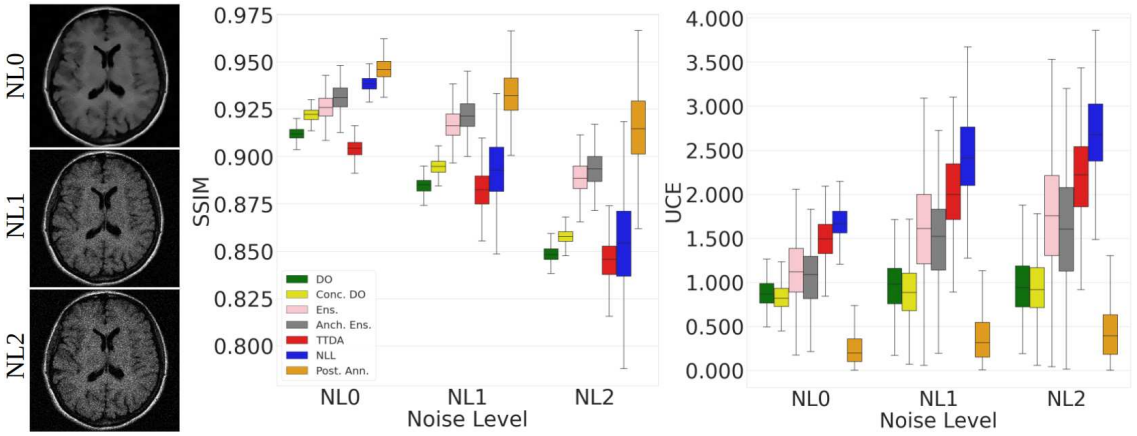

Figure 7.6: Evaluation of different methods using out-of-distribution input samples for MRI translation. While the models are trained on MRI samples at noise-level 0 (i.e., NL0), they are evaluated on increasingly noisy samples (i.e., at noise levels NL1 and NL2). We notice that the proposed method performs better than various baselines.

To evaluate if better quality of uncertainty estimates lead to better OOD performance, we evaluate all the uncertainty trained model for MRI Translation on OOD samples. MRI image acquisition is a noisy process that leads to noisy/corrupted images [163, 196, 294, 3]. Similar to [261, 264, 246], we study the performance of various uncertainty-aware models in the presence of noisy input samples (corrupted with varying degrees of noise) at test time. Figure 7.6-(left) shows the example of in-distribution (noise-level 0, NL0) and out-of-distribution samples (NL1 and NL2). The severity of corruption gradually increases from NL0 to NL2. From Figure 7.6-(middle and right), that shows the regression and quality of uncertainty estimates in the presence of OOD samples, we observe that the performance of various models degrades as severity of corruption increases from NL0 to NL2, however our LIKA method performs much better than the compared methods even at higher severity of corruption both in terms of regression and uncertainty calibration metric.

7.6 Conclusion

This paper introduces a novel approach to improve the calibration of uncertainty estimates for regression tasks. We propose a temperature-dependent likelihood that allows for faster and more accurate learning, while avoiding the need for post-hoc calibration. Our method employs a temperature annealing technique during training, which has been shown to lead to 1.5 to 6 times faster convergence compared to existing approaches. Additionally, we demonstrate the effectiveness of our method in producing superior regression results with better calibrated uncertainty estimates, compared to five existing uncertainty estimation methods, across multiple datasets. We further investigate the potential of our approach in out-of-distribution scenarios, showing its ability to generalize well and highlighting its robustness. Our study also includes an ablation analysis, revealing key components of our method and providing valuable insights for future research in uncertainty estimation. Overall, our proposed temperature-dependent likelihood represents a promising direction for improving the efficiency and accuracy of uncertainty estimation in regression tasks.

8

Thesis Discussion and Conclusion

This thesis deals with the problem setting of uncertainty quantification in the computer vision domain. The purpose of our work is to quantify the uncertainty in the predictions made by machine learning-based computer vision models and to leverage the derived uncertainty to enhance the model capabilities.

In the previous chapters, a novel methods and applications of uncertainty quantifications for deep learning-based computer vision systems were established.

The following section reviews each contribution individually and collectively and discusses their strong sides, as well as take a look at their current limitations, proposing how the drawbacks could be overcome in the future.

8.1 Discussion of results

The investigation began with a novel approach to enhance the performance of GANs in medical imaging tasks through the integration of uncertainty in a progressive fashion to an existing GAN-based image translation method (Chapter 1). This exploration laid the groundwork for the subsequent development of a unique probabilistic method based on UGAC for unpaired image-to-image translation, offering a robust solution to address unseen perturbations in test data (Chapter 2).

We further expanded the scope of our research to the realm of learning to quantify the uncertainty in large-scale pretrained frozen computer vision models that often achieve state-of-the-art performances but are deterministic in nature. We proposed innovative techniques such as BayesCap (Chapter 3) to provide calibrated uncertainty estimates without the need for retraining. The utility of derived uncertainty estimates for active learning in dense regression models was also examined, culminating in the creation of the USIM-DAL framework (Chapter 4), which demonstrated significant potential in a variety of applications.

In an effort to enrich vision-language models (VLMs), we introduced ProbVLM (Chapter 5), a probabilistic adapter for pretrained VLMs. The success of this approach in

estimating multi-modal embedding uncertainties aiding active learning and model selection underlines the broad implications of this research.

Finally, our investigation into fast-calibrated uncertainty estimation for regression tasks led to the development of Likelihood Annealing (Chapter 6). This approach showcases how refined uncertainty estimation techniques can improve the convergence of deep regression models across a spectrum of regression problems, including high-dimensional tasks.

Collectively, the research undertaken in this thesis has advanced the field of machine learning, specifically in the areas of uncertainty estimation, image translation, and active learning. The results achieved not only improve upon current methodologies but also provide a foundation for further advancements and exploration in these vital areas.

As the demand for reliable, efficient, and robust artificial intelligence systems continues to grow, the ability to accurately quantify and interpret uncertainty will remain an essential component of these systems. The implications of these findings are far-reaching, particularly in sectors such as healthcare and autonomous driving, where they can make profound contributions. As we look ahead, it is anticipated that the findings and methodologies proposed in this thesis will serve as stepping stones for future research in these critical areas of artificial intelligence.

8.2 Conclusion and future works

The research conducted throughout this doctoral journey has opened many exciting avenues for future exploration. While substantial progress has been made in the realms of uncertainty estimation, image translation, active learning, and dense regression tasks, there are several aspects that warrant further investigation.

In the area of image-to-image translation, while our novel approach integrating aleatoric uncertainty into GANs has demonstrated promising results in medical imaging tasks, there is scope for broadening the application of this method to other domains.

The proposed probabilistic method based on UGAC has shown resilience to unseen perturbations in unpaired image-to-image translation tasks. Extending this resilience to a wider range of perturbations and scenarios could strengthen the robustness of such systems further.

The introduction of BayesCap represented a significant advancement in providing calibrated uncertainty estimates for non-Bayesian models. Future work could involve refining this method to reduce its computational requirements even further and to explore its applicability to different types of models beyond deep learning architectures.

Our exploration into the utility of uncertainty estimates for active learning in dense regression models led to the creation of the USIM-DAL framework. Going forward, it would be beneficial to investigate other statistical properties that could be leveraged to enhance the efficacy of active learning strategies.

The success of ProbVLM in enriching vision-language models suggests the potential for further exploration of probabilistic approaches in this area. In particular, work could be directed towards improving the model's performance in more complex multi-modal tasks.

Finally, the development of Likelihood Annealing, our method for fast calibrated uncertainty estimation for regression tasks, has demonstrated considerable promise. Future research could be directed towards developing techniques that can provide even faster and more accurate uncertainty quantification and applying these to more complex regression tasks.

In conclusion, while this research has made significant strides in uncertainty estimation, image translation, and active learning, there are still numerous avenues for exploration and development. We believe the techniques and methodologies proposed in this thesis will serve as a solid foundation for future work in these critical areas of machine learning and artificial intelligence.

Bibliography

[1] F. Aeffner et al. "Digital microscopy, image analysis, and virtual slide repository". In: *ILAR journal* (2018) (cit. on p. 45).

[2] A. S. Aghaei, K. N. Plataniotis, and S. Pasupathy. "Maximum likelihood binary detection in improper complex Gaussian noise". In: *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE. 2008, pp. 3209–3212 (cit. on p. 75).

[3] S. Aja-Fernández and G. Vegas-Sánchez-Ferrero. "Statistical analysis of noise in MRI". In: *Switzerland: Springer International Publishing* (2016) (cit. on p. 85).

[4] J.-B. Alayrac et al. "Flamingo: a visual language model for few-shot learning". In: *arXiv preprint arXiv:2204.14198* (2022) (cit. on pp. 57, 59).

[5] K. Armanious et al. "MedGAN: Medical image translation using GANs". In: *Computerized Medical Imaging and Graphics* (2020) (cit. on pp. 7, 8, 11, 12).

[6] K. Armanious et al. "MedGAN: Medical image translation using GANs". In: *Computerized Medical Imaging and Graphics* (2020) (cit. on pp. 17, 22, 29).

[7] K. Armanious et al. "Unsupervised medical image translation using Cycle-MedGAN". In: *European Signal Processing Conference (EUSIPCO)*. 2019 (cit. on pp. 7, 8).

[8] E. Artin. *The gamma function*. Courier Dover Publications, 2015 (cit. on p. 34).

[9] M. S. Ayhan and P. Berens. "Test-time data augmentation for estimation of heteroscedastic aleatoric uncertainty in deep neural networks". In: *MIDL* (2018) (cit. on pp. 30, 32, 36, 60, 65, 72, 80).

[10] G. Bae, I. Budvytis, and R. Cipolla. "Estimating and Exploiting the Aleatoric Uncertainty in Surface Normal Estimation". In: *IEEE ICCV*. 2021 (cit. on pp. 31, 47, 60).

[11] M. Bain et al. "A CLIP-Hitchhiker's Guide to Long Video Retrieval". In: *arXiv preprint arXiv:2205.08508* (2022) (cit. on p. 59).

[12] M. Baradad Jurjo et al. "Learning to see by looking at noise". In: *NeurIPS* (2021) (cit. on pp. 47–50).

[13] D. Bashkirova, B. Usman, and K. Saenko. "Adversarial self-defense for cycle-consistent GANs". In: *NeurIPS* (2019) (cit. on pp. 17, 22, 23).

[14] E. Begoli, T. Bhattacharya, and D. Kusnezov. "The need for uncertainty quantification in machine-assisted medical decision making". In: *Nature Machine Intelligence* (2019) (cit. on p. 17).

[15] D. A. Belsley, E. Kuh, and R. E. Welsch. *Regression diagnostics: Identifying influential data and sources of collinearity*. 2005 (cit. on p. 78).

[16] W. H. Beluch et al. "The Power of Ensembles for Active Learning in Image Classification". In: *CVPR*. 2018 (cit. on pp. 46, 48).

[17] S. Benaim and L. Wolf. "One-Sided Unsupervised Domain Mapping". In: *NIPS*. 2017 (cit. on p. 23).

[18] C. A. Bertram and R. Klopfleisch. "The pathologist 2.0: an update on digital pathology in veterinary medicine". In: *Veterinary pathology* (2017) (cit. on p. 45).

[19] V. Besnier, D. Picard, and A. Briot. "Learning uncertainty for safety-oriented semantic segmentation in autonomous driving". In: *ICIP*. 2021 (cit. on p. 60).

[20] V. Besnier et al. "Triggering Failures: Out-of-Distribution Detection by Learning From Local Adversarial Attacks in Semantic Segmentation". In: *ICCV*. 2021 (cit. on p. 60).

[21] V. Besnier et al. "Triggering Failures: Out-Of-Distribution detection by learning from local adversarial attacks in Semantic Segmentation". In: *IEEE ICCV*. 2021 (cit. on p. 71).

[22] M. Bevilacqua et al. "Low-complexity single-image super-resolution based on nonnegative neighbor embedding". In: *BMVC*. 2012 (cit. on p. 36).

[23] M. J. Black and A. Rangarajan. "On the unification of line processes, outlier rejection, and robust statistics with applications in early vision". In: *IJCV* (1996) (cit. on p. 16).

[24] C. Blundell et al. "Weight uncertainty in neural network". In: *ICML*. 2015 (cit. on pp. 31, 47, 60, 72).

[25] R. Bose et al. "Zero-Shot Remote Sensing Image Super-Resolution Based on Image Continuity and Self Tessellations". In: *GCPR*. 2022 (cit. on p. 47).

[26] C. Bouman and K. Sauer. "A generalized Gaussian image model for edge-preserving MAP estimation". In: *IEEE TIP* (1993) (cit. on p. 17).

[27] C. Bowles et al. "Pseudo-healthy image synthesis for white matter lesion segmentation". In: *International Workshop on Simulation and Synthesis in Medical Imaging*. 2016 (cit. on pp. 36, 40).

[28] K. Brinker. "Incorporating diversity in active learning with support vector machines". In: *ICML*. 2003 (cit. on p. 46).

[29] G. Caner, A. Tekalp, and W. Heinzelman. "Super resolution recovery for multi-camera surveillance imaging". In: *International Conference on Multimedia and Expo*. 2003 (cit. on p. 45).

[30] A. Chartsias et al. "Multimodal MR synthesis via modality-invariant latent representation". In: *IEEE TMI* (2017) (cit. on pp. 36, 40).

[31] H. Chen et al. "Learning prediction intervals for regression: Generalization and calibration". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2021 (cit. on pp. 71, 72).

[32] H. Chen et al. "Low-dose CT with a residual encoder-decoder convolutional neural network". In: *IEEE TMI* (2017) (cit. on p. 7).

[33] M. Chen et al. "Mandoline: Model evaluation under distribution shift". In: *ICML*. 2021 (cit. on p. 68).

[34] T. Chen, E. Fox, and C. Guestrin. "Stochastic gradient hamiltonian monte carlo". In: *ICML*. PMLR. 2014 (cit. on pp. 31, 72).

[35] Y.-C. Chen and R. Techawitthayachinda. "Developing deep learning in science classrooms: Tactics to manage epistemic uncertainty during whole-class discussion". In: *Journal of Research in Science Teaching* 58.8 (2021), pp. 1083–1116 (cit. on p. 73).

[36] C.-Y. Chuang, A. Torralba, and S. Jegelka. "Estimating generalization under distribution shifts via domain-invariant representations". In: *arXiv preprint arXiv:2007.03511* (2020) (cit. on p. 68).

[37] S. Chun et al. "Probabilistic embeddings for cross-modal retrieval". In: *CVPR*. 2021 (cit. on pp. 58–61, 64, 65).

[38] Y. Chung et al. "Beyond pinball loss: Quantile methods for calibrated uncertainty quantification". In: *NeurIPS* (2021) (cit. on pp. 71, 72, 79, 81, 83).

[39] C. Coglianese and D. Lehr. "Regulating by robot: Administrative decision making in the machine-learning era". In: *Geo. LJ* (2016) (cit. on p. 30).

[40] J. P. Cohen, M. Luck, and S. Honari. "Distribution matching losses can hallucinate features in medical image translation". In: *MICCAI*. 2018 (cit. on p. 29).

[41] C. Coleman et al. "Selection via Proxy: Efficient Data Selection for Deep Learning". In: *ICLR*. 2020 (cit. on p. 68).

[42] M. Cordts et al. "The cityscapes dataset for semantic urban scene understanding". In: *CVPR*. 2016 (cit. on pp. 17, 21).

[43] P. Coretto and C. Hennig. "Consistency, breakdown robustness, and algorithms for robust improper maximum likelihood clustering". In: *Journal of Machine Learning Research* 18.142 (2017), pp. 1–39 (cit. on p. 75).

[44] P. Coretto and C. Hennig. "Robust improper maximum likelihood: tuning, computation, and a comparison with other methods for robust Gaussian clustering". In: *Journal of the American Statistical Association* 111.516 (2016), pp. 1648–1659 (cit. on p. 75).

[45] J. Cornebise, I. Oršolić, and F. Kalaitzis. "Open High-Resolution Satellite Imagery: The WorldStrat Dataset–With Application to Super-Resolution". In: *arXiv preprint arXiv:2207.06418* (2022) (cit. on p. 45).

[46] J. Cornebise et al. "Witnessing atrocities: quantifying villages destruction in Darfur with crowdsourcing and transfer learning". In: *Proc. AI for Social Good NeurIPS2018 Workshop, NeurIPS'18*. 2018 (cit. on p. 45).

[47] P. Cui, W. Hu, and J. Zhu. "Calibrated reliable regression using maximum mean discrepancy". In: *NeurIPS* (2020) (cit. on p. 73).

[48] Y. CUI et al. "Bayes-MIL: A New Probabilistic Perspective on Attention-based Multiple Instance Learning for Whole Slide Images". In: *ICLR*. 2023 (cit. on p. 60).

[49] S. U. Dar et al. "Image synthesis in multi-contrast MRI with conditional generative adversarial networks". In: *IEEE TMI* (2019) (cit. on p. 17).

[50] E. Daxberger et al. "Laplace Redux-Effortless Bayesian Deep Learning". In: *NeurIPS* (2021) (cit. on pp. 30–32, 47, 72).

[51] W. Deng, S. Gould, and L. Zheng. "What does rotation prediction tell us about classifier accuracy under varying testing environments?" In: *ICML*. 2021 (cit. on p. 68).

[52] W. Deng and L. Zheng. "Are labels always necessary for classifier accuracy evaluation?" In: *CVPR*. 2021 (cit. on p. 68).

[53] M. Dohopolski et al. "Predicting lymph node metastasis in patients with oropharyngeal cancer by using a convolutional neural network with associated epistemic and aleatoric uncertainty". In: *Physics in Medicine & Biology* 65.22 (2020), p. 225002 (cit. on p. 73).

[54] C. Dong et al. "Image super-resolution using deep convolutional networks". In: *IEEE TPAMI* (2015) (cit. on p. 31).

[55] A. Dosovitskiy et al. "Flownet: Learning optical flow with convolutional networks". In: *IEEE ICCV*. 2015 (cit. on p. 30).

[56] Q. Dou et al. "Multilevel contextual 3-D CNNs for false positive reduction in pulmonary nodule detection". In: *IEEE Transactions on Biomedical Engineering* (2016) (cit. on p. 7).

[57] S. Ebrahimi et al. "Uncertainty-guided continual learning with bayesian neural networks". In: *arXiv preprint arXiv:1906.02425* (2019) (cit. on p. 46).

[58] A. Emami-Naeini, M. M. Akhter, and S. M. Rock. "Effect of model uncertainty on failure detection: the threshold selector". In: *IEEE Transactions on Automatic Control* (1988) (cit. on p. 60).

[59] R. Eschenhagen et al. "Mixtures of Laplace Approximations for Improved *Post-Hoc* Uncertainty in Deep Learning". In: *NeurIPS Workshop on Bayesian Deep Learning* (2021) (cit. on p. 32).

[60] F. Faghri et al. "Vse++: Improving visual-semantic embeddings with hard negatives". In: *BMVC*. 2018 (cit. on p. 64).

[61] B. Fang et al. "UATVR: Uncertainty-Adaptive Text-Video Retrieval". In: *arXiv preprint arXiv:2301.06309* (2023) (cit. on pp. 60, 61).

[62] S. Feldman, S. Bates, and Y. Romano. "Improving conditional coverage via orthogonal quantile regression". In: *Advances in Neural Information Processing Systems* (2021) (cit. on p. 71).

[63] D. J. Field. "Relations between the statistics of natural images and the response properties of cortical cells". In: *Josa a* (1987) (cit. on p. 46).

[64] G. Franchi et al. "One Versus all for deep Neural Network for uncertaInty (OVNNI) quantification". In: *IEEE Access* 10 (2021), pp. 7300–7312 (cit. on p. 60).

[65] G. Franchi et al. "TRADI: Tracking deep neural network weight distributions". In: *ECCV*. 2020 (cit. on pp. 60, 71).

[66] C. Frogner and T. Poggio. "Fast and Flexible Inference of Joint Distributions from their Marginals". In: *ICML*. 2019 (cit. on p. 15).

[67] H. Fu et al. "Deep ordinal regression network for monocular depth estimation". In: *IEEE CVPR*. 2018 (cit. on pp. 29, 31, 32).

[68] H. Fu et al. "Geometry-consistent generative adversarial networks for one-sided unsupervised domain mapping". In: *CVPR*. 2019 (cit. on pp. 17, 23).

[69] Y. Gal. "Uncertainty in deep learning". In: (2016) (cit. on p. 72).

[70] Y. Gal and Z. Ghahramani. "Dropout as a bayesian approximation: Representing model uncertainty in deep learning". In: *ICML*. 2016 (cit. on pp. 17, 31, 36, 60, 61, 71, 80, 83).

[71] P. Gao et al. "Clip-adapter: Better vision-language models with feature adapters". In: *arXiv preprint arXiv:2110.04544* (2021) (cit. on p. 59).

[72] V. Garcia Satorras, Z. Akata, and M. Welling. "Combining Generative and Discriminative Models for Hybrid Inference". In: *NeurIPS*. 2019 (cit. on p. 78).

[73] S. Garg et al. "Leveraging unlabeled data to predict out-of-distribution performance". In: *ICLR*. 2022 (cit. on p. 68).

[74] J. Gast and S. Roth. "Lightweight Probabilistic Deep Networks". In: *CVPR*. 2018 (cit. on p. 60).

[75] J. Gawlikowski et al. "A survey of uncertainty in deep neural networks". In: *arXiv preprint arXiv:2107.03342* (2021) (cit. on pp. 32, 80, 83).

[76] A. Geiger et al. "Vision meets robotics: The kitti dataset". In: *The International Journal of Robotics Research* (2013) (cit. on p. 42).

[77] C. Gentile. "The robustness of the p-norm algorithms". In: *Machine Learning* (2003) (cit. on p. 16).

[78] G. Ghiasi et al. "Open-vocabulary image segmentation". In: *arXiv preprint arXiv:2112.12143* (2021) (cit. on p. 59).

[79] C. Gillmann et al. "Uncertainty-aware Visualization in Medical Imaging-A Survey". In: *Computer Graphics Forum*. Wiley Online Library. 2021 (cit. on p. 71).

[80] C. Godard, O. Mac Aodha, and G. J. Brostow. "Unsupervised monocular depth estimation with left-right consistency". In: *IEEE CVPR*. 2017 (cit. on pp. 30, 31).

[81] C. Godard et al. "Digging into self-supervised monocular depth estimation". In: *IEEE ICCV*. 2019 (cit. on pp. 29–31, 42, 43).

[82] S. Gohshi. "Real-time super resolution algorithm for security cameras". In: *International Joint Conference on e-Business and Telecommunications (ICETE)*. 2015 (cit. on p. 45).

[83] I. J. Goodfellow et al. "Generative adversarial networks". In: *NIPS* (2014) (cit. on p. 21).

[84] M. Gorriz et al. "Cost-effective active learning for melanoma segmentation". In: *arXiv preprint arXiv:1711.09168* (2017) (cit. on pp. 46, 48).

[85] A. Graves. "Practical variational inference for neural networks". In: *NIPS* (2011) (cit. on pp. 31, 47, 60, 72).

[86] D. Guillory et al. "Predicting with confidence on unseen distributions". In: *ICCV*. 2021 (cit. on p. 68).

[87] C. Guo et al. "On calibration of modern neural networks". In: *ICML*. 2017 (cit. on pp. 17, 32, 37, 72, 73, 79).

[88] H. Guo, H. Wang, and Q. Ji. "Uncertainty-guided probabilistic transformer for complex action recognition". In: *CVPR*. 2022 (cit. on p. 60).

[89] M. Gutmann and A. Hyvärinen. "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models". In: *AISTATS*. 2010 (cit. on p. 59).

[90] P. W. Hamilton et al. "Digital pathology and image analysis in tissue biomarker research". In: *Methods* (2014) (cit. on p. 45).

[91] D. Harrison Jr and D. L. Rubinfeld. "Hedonic housing prices and the demand for clean air". In: *Journal of environmental economics and management* (1978) (cit. on p. 78).

[92] T. Hastie, R. Tibshirani, and M. Wainwright. *Statistical Learning with Sparsity: The Lasso and Generalizations*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. CRC Press, 2015. ISBN: 9781498712170. URL: https://books.google.de/books?id=f-A_CQAAQBAJ (cit. on p. 16).

[93] M. Havaei et al. "Brain tumor segmentation with deep neural networks". In: *Medical Image Analysis* (2017) (cit. on p. 7).

[94] D. J. Heeger and J. R. Bergen. "Pyramid-based texture analysis/synthesis". In: *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*. 1995 (cit. on p. 46).

[95] D. Hendrycks, K. Lee, and M. Mazeika. "Using pre-training can improve model robustness and uncertainty". In: *International Conference on Machine Learning*. 2019 (cit. on p. 84).

[96] J. Hornauer and V. Belagiannis. "Gradient-Based Uncertainty for Monocular Depth Estimation". In: *ECCV*. 2022 (cit. on p. 60).

[97] S. Hu et al. "Supervised uncertainty quantification for segmentation with multiple annotations". In: *MICCAI*. 2019 (cit. on p. 17).

[98] P. J. Huber et al. "The 1972 wald lecture robust statistics: A review". In: *Annals of Mathematical Statistics* (1972) (cit. on p. 16).

[99] E. Hüllermeier and W. Waegeman. "Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods". In: *Machine Learning* (2021) (cit. on p. 70).

[100] T. Huster et al. "Pareto GAN: Extending the Representational Power of GANs to Heavy-Tailed Distributions". In: *ICML*. 2021 (cit. on p. 62).

[101] T. Huynh et al. "Estimating CT image from MRI data using structured random forest and auto-context model". In: *IEEE TMI* (2015) (cit. on p. 7).

[102] J. E. Iglesias et al. "Is synthesizing MRI contrast useful for inter-modality analysis?" In: *MICCAI*. 2013 (cit. on pp. 36, 40).

[103] A. Ignatov, R. Timofte, et al. "PIRM challenge on perceptual image enhancement on smartphones: report". In: *ECCV Workshop*. 2019 (cit. on p. 78).

[104] S. Iizuka, E. Simo-Serra, and H. Ishikawa. "Let there be color! Joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification". In: *ACM Transactions on Graphics* (2016) (cit. on pp. 17, 29).

[105] G. Ilharco et al. "Editing Models with Task Arithmetic". In: *arXiv preprint arXiv:2212.04089* (2022) (cit. on p. 68).

[106] G. Ilharco et al. "Patching open-vocabulary models by interpolating weights". In: *arXiv preprint arXiv:2208.05592* (2022) (cit. on p. 68).

[107] P. Isola et al. "Image-to-image translation with conditional adversarial networks". In: *IEEE CVPR*. 2017 (cit. on pp. 7, 9, 11, 12, 15, 17, 20–22).

[108] P. Izmailov et al. "Averaging weights leads to wider optima and better generalization". In: *arXiv preprint arXiv:1803.05407* (2018) (cit. on p. 60).

[109] S. D. Jain and K. Grauman. "Active image segmentation propagation". In: *CVPR*. 2016 (cit. on p. 46).

[110] S. Jeong et al. "Memory-guided Unsupervised Image-to-image Translation". In: *CVPR*. 2021 (cit. on p. 16).

[111] Y. Ji et al. "MAP: Modality-Agnostic Uncertainty-Aware Vision-Language Pre-training Model". In: *arXiv preprint arXiv:2210.05335* (2022) (cit. on p. 60).

[112] C. Jia et al. "Scaling up visual and vision-language representation learning with noisy text supervision". In: *ICML*. 2021 (cit. on pp. 57, 58).

[113] A. Jungo and M. Reyes. "Assessing reliability and challenges of uncertainty estimations for medical image segmentation". In: *MICCAI*. 2019 (cit. on p. 17).

[114] H. D. Kabir et al. "Neural network-based uncertainty quantification: A survey of methodologies and applications". In: *IEEE access* (2018) (cit. on p. 17).

[115] K. Kamnitsas et al. "Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation". In: *Medical Image Analysis* (2017) (cit. on p. 7).

[116] T. Karras et al. "Progressive Growing of GANs for Improved Quality, Stability, and Variation". In: *ICLR*. 2018 (cit. on pp. 7, 8).

[117] H. Kataoka et al. "Pre-training without natural images". In: *Proceedings of the Asian Conference on Computer Vision*. 2020 (cit. on p. 46).

[118] A. Kendall, V. Badrinarayanan, and R. Cipolla. "Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding". In: *arXiv preprint arXiv:1511.02680* (2015) (cit. on p. 36).

[119] A. Kendall and Y. Gal. "What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?" In: *NIPS*. 2017 (cit. on pp. 7, 17).

[120] A. Kendall and Y. Gal. "What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?" In: *NIPS* (2017) (cit. on pp. 31–33, 37, 42, 60, 61, 71–74, 79, 80, 83).

[121] A. Kendall and Y. Gal. "What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?" In: *NeurIPS*. 2017 (cit. on pp. 46, 47, 49, 51).

[122] D. P. Kingma and J. Ba. "Adam: A method for stochastic optimization". In: *preprint arXiv:1412.6980* (2014) (cit. on pp. 10, 37, 79, 81).

[123] D. P. Kingma and J. Ba. "Adam: A method for stochastic optimization". In: *ICLR* (2015) (cit. on p. 22).

[124] M. Kirchhof, E. Kasneci, and S. J. Oh. "Probabilistic Contrastive Learning Recovers the Correct Aleatoric Uncertainty of Ambiguous Inputs". In: *arXiv preprint arXiv:2302.02865* (2023) (cit. on p. 59).

[125] A. Kirsch, J. Van Amersfoort, and Y. Gal. "Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning". In: *NeurIPS* (2019) (cit. on p. 60).

[126] M. Kläs and A. M. Vollmer. "Uncertainty in machine learning applications: A practice-driven classification of uncertainty". In: *International conference on computer safety, reliability, and security*. 2018 (cit. on p. 70).

[127] S. Kohl et al. "A Probabilistic U-Net for Segmentation of Ambiguous Images". In: *NeurIPS*. 2018 (cit. on p. 17).

[128] B. Kompa, J. Snoek, and A. L. Beam. "Second opinion needed: communicating uncertainty in medical machine learning". In: *NPJ Digital Medicine* 4.1 (2021), p. 4 (cit. on p. 73).

[129] E. R. Kretzmer. "Statistics of television signals". In: *The bell system technical journal* (1952) (cit. on pp. 46, 48).

[130] R. Krishna et al. "Visual genome: Connecting language and vision using crowdsourced dense image annotations". In: *IJCV* (2017) (cit. on p. 51).

[131] V. Kuleshov, N. Fenner, and S. Ermon. "Accurate uncertainties for deep learning using calibrated regression". In: *ICML*. 2018 (cit. on pp. 32, 73).

[132] N. Kumar et al. "Kernel generalized Gaussian and robust statistical learning for abnormality detection in medical images". In: *ICIP*. 2017 (cit. on p. 62).

[133] N. Kumar et al. "Kernel generalized-Gaussian mixture model for robust abnormality detection". In: *MICCAI*. 2017 (cit. on p. 62).

[134] O. Kupyn et al. "Deblurgan: Blind motion deblurring using conditional adversarial networks". In: *IEEE CVPR*. 2018 (cit. on pp. 29, 31, 32, 36).

[135] O. Kupyn et al. "Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better". In: *IEEE ICCV*. 2019 (cit. on pp. 29, 31, 36).

[136] T. Küstner et al. "MR-based respiratory and cardiac motion correction for PET imaging". In: *Medical Image Analysis* (2017) (cit. on p. 7).

[137] S. Laine et al. "High-Quality Self-Supervised Deep Image Denoising". In: *NeurIPS*. 2019 (cit. on p. 21).

[138] B. Lakshminarayanan, A. Pritzel, and C. Blundell. "Simple and scalable predictive uncertainty estimation using deep ensembles". In: *arXiv preprint arXiv:1612.01474* (2016) (cit. on pp. 31, 49, 71, 72, 80, 83).

[139] B. Lakshminarayanan, A. Pritzel, and C. Blundell. "Simple and scalable predictive uncertainty estimation using deep ensembles". In: *NIPS*. 2017 (cit. on pp. 17, 60, 61).

[140] M.-H. Laves et al. "Calibration of Model Uncertainty for Dropout Variational Inference". In: *arXiv preprint arXiv:2006.11584* (2020) (cit. on pp. 32, 37).

[141] M.-H. Laves et al. "Well-calibrated model uncertainty with temperature scaling for dropout variational inference". In: *arXiv preprint arXiv:1909.13550* (2019) (cit. on p. 36).

[142] M.-H. Laves et al. "Well-calibrated regression uncertainty in medical imaging with deep learning". In: *MIDL*. 2020 (cit. on pp. 31, 32, 47, 60, 73, 75, 79).

[143] C. Ledig et al. "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network". In: *IEEE CVPR*. 2017 (cit. on pp. 29, 31, 32, 36).

[144] C. Ledig et al. "Photo-realistic single image super-resolution using a generative adversarial network". In: *IEEE CVPR*. 2017 (cit. on pp. 47, 51).

[145] A. B. Lee, D. Mumford, and J. Huang. "Occlusion models for natural images: A statistical study of a scale-invariant dead leaves model". In: *IJCV* (2001) (cit. on p. 46).

[146] C. Leibig et al. "Leveraging uncertainty information from deep neural networks for disease detection". In: *Scientific reports* (2017) (cit. on p. 71).

[147] D. Levi et al. "Evaluating and calibrating uncertainty prediction in regression tasks". In: *arXiv preprint arXiv:1905.11659* (2019) (cit. on pp. 32, 73, 75, 79).

[148] H. Li et al. "A Differentiable Semantic Metric Approximation in Probabilistic Embedding for Cross-Modal Retrieval". In: *NeurIPS*. 2022 (cit. on pp. 59, 60).

[149] J. Li et al. "BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models". In: *arXiv preprint arXiv:2301.12597* (2023) (cit. on p. 59).

[150] J. Li et al. "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation". In: *ICML*. 2022 (cit. on pp. 57–59, 61, 65, 66).

[151] Y Li, B. Sixou, and F Peyrin. "A review of the deep learning methods for medical images super resolution problems". In: *Irbm* (2021) (cit. on p. 45).

[152] J. Liang et al. "SwinIR: Image Restoration Using Swin Transformer". In: *ICCVw*. 2021 (cit. on p. 47).

[153] B. Lim et al. "Enhanced Deep Residual Networks for Single Image Super-Resolution". In: *CVPRw*. 2017 (cit. on p. 45).

[154] T.-Y. Lin et al. "Microsoft coco: Common objects in context". In: *ECCV*. 2014 (cit. on pp. 58, 59, 64).

[155] Z. Lin, S. Trivedi, and J. Sun. "Locally Valid and Discriminative Prediction Intervals for Deep Learning Models". In: *Advances in Neural Information Processing Systems* (2021) (cit. on p. 72).

[156] T. Lindvall. *Lectures on the coupling method*. Courier Corporation, 2002 (cit. on pp. 15, 18).

[157] G. Litjens et al. "1399 H&E-stained sentinel lymph node sections of breast cancer patients: the CAMELYON dataset". In: *GigaScience* (2018) (cit. on p. 51).

[158] G. Litjens et al. "A survey on deep learning in Medical Image Analysis". In: *Medical Image Analysis* (2017) (cit. on pp. 7, 31).

[159] M.-Y. Liu, T. Breuel, and J. Kautz. "Unsupervised image-to-image translation networks". In: *NIPS*. 2017 (cit. on pp. 15–17, 23).

[160] M.-Y. Liu, T. Breuel, and J. Kautz. "Unsupervised image-to-image translation networks". In: *NIPS* (2017) (cit. on p. 36).

[161] J. Long, E. Shelhamer, and T. Darrell. "Fully convolutional networks for semantic segmentation". In: *CVPR*. 2015 (cit. on p. 17).

[162] J. Lu et al. "Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks". In: *NeurIPS* (2019) (cit. on p. 59).

[163] A. Macovski. "Noise in MRI". In: *Magnetic resonance in medicine* (1996) (cit. on p. 85).

[164] W. J. Maddox et al. "A simple baseline for bayesian uncertainty in deep learning". In: *NeurIPS* (2019) (cit. on pp. 36, 71, 72).

[165] F. Mainardi, G. Pagnini, and R. Saxena. "Fox H functions in fractional diffusion". In: *Journal of Computational and Applied Mathematics* (2005). Proceedings of the Seventh International Symposium on Orthogonal Polynomials,Special Functions and Applications (cit. on p. 63).

[166] D. Martin et al. "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics". In: *ICCV*. 2001 (cit. on p. 51).

[167] D. Martin et al. "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics". In: *IEEE ICCV*. 2001 (cit. on p. 36).

[168] A. M. Mathai, R. K. Saxena, and H. J. Haubold. *The H-function: theory and applications*. Springer Science & Business Media, 2009 (cit. on p. 63).

[169] S. Mathew et al. "Augmenting Colonoscopy Using Extended and Directional CycleGAN for Lossy Image Translation". In: *CVPR*. 2020 (cit. on p. 15).

[170] R. McAllister et al. "Concrete problems for autonomous vehicle safety: Advantages of Bayesian deep learning". In: *IJCAI*. 2017 (cit. on p. 31).

[171] A. Mehrtash et al. "Pep: Parameter ensembling by perturbation". In: *NeurIPS* (2020) (cit. on p. 71).

[172] R. Mehta et al. "Uncertainty Evaluation Metric for Brain Tumour Segmentation". In: *MIDL*. 2020 (cit. on p. 17).

[173] S. Messoudi, S. Rousseau, and S. Destercke. "Deep conformal prediction for robust models". In: *Information Processing and Management of Uncertainty in Knowledge-Based Systems: 18th International Conference, IPMU 2020, Lisbon, Portugal, June 15–19, 2020, Proceedings, Part I 18*. Springer. 2020, pp. 528–540 (cit. on p. 72).

[174] R. Michelmore, M. Kwiatkowska, and Y. Gal. "Evaluating uncertainty quantification in end-to-end autonomous driving control". In: *arXiv preprint arXiv:1811.06817* (2018) (cit. on p. 42).

[175] Y. Ming et al. "Delving into Out-of-Distribution Detection with Vision-Language Representations". In: *NeurIPS*. 2022 (cit. on p. 59).

[176] M. Monteiro et al. "Stochastic segmentation networks: Modelling spatially correlated aleatoric uncertainty". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 12756–12767 (cit. on p. 72).

[177] N. Mu et al. "Slip: Self-supervision meets language-image pre-training". In: *ECCV*. 2022 (cit. on pp. 57–59, 61).

[178] J. Mukhoti et al. "Deterministic neural networks with appropriate inductive biases capture epistemic and aleatoric uncertainty". In: *arXiv preprint arXiv:2102.11582* (2021) (cit. on p. 72).

[179] S. A. Munagala et al. "CLActive: Episodic Memories for Rapid Active Learning". In: *CoLLAS*. 2022 (cit. on p. 60).

[180] M. Mundt et al. "Open set recognition through deep neural network uncertainty: Does out-of-distribution detection require generative classifiers?" In: *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*. 2019 (cit. on p. 84).

[181] S. Nah, T. Hyun Kim, and K. Mu Lee. "Deep multi-scale convolutional neural network for dynamic scene deblurring". In: *IEEE CVPR*. 2017 (cit. on pp. 31, 36).

[182] T. Nair et al. "Exploring uncertainty measures in deep networks for multiple sclerosis lesion detection and segmentation". In: *Medical Image Analysis* (2020) (cit. on pp. 7, 17, 47).

[183] J. Nandy, W. Hsu, and M. L. Lee. "Towards maximizing the representation gap between in-domain & out-of-distribution examples". In: *Advances in Neural Information Processing Systems* (2020) (cit. on p. 84).

[184] J. Nazarovs et al. "Understanding Uncertainty Maps in Vision With Statistical Testing". In: *CVPR*. 2022 (cit. on p. 60).

[185] A. Neculai, Y. Chen, and Z. Akata. "Probabilistic Compositional Embeddings for Multimodal Image Retrieval". In: *CVPR-W*. 2022 (cit. on p. 60).

[186] D. Nie et al. "Medical image synthesis with deep convolutional adversarial networks". In: *IEEE Transactions on Biomedical Engineering* (2018) (cit. on p. 7).

[187] M.-E. Nilsback and A. Zisserman. "Automated flower classification over a large number of classes". In: *ICVGIP*. 2008 (cit. on pp. 58, 65).

[188] D. A. Nix and A. S. Weigend. "Estimating the mean and variance of the target probability distribution". In: *ICNN*. 1994 (cit. on p. 60).

[189] J. Oh and N. Kwak. "Generalized mean for robust principal component analysis". In: *Pattern Recognition* (2016) (cit. on p. 17).

[190] S. J. Oh et al. "Modeling uncertainty with hedged instance embedding". In: *ICLR* (2019) (cit. on pp. 58, 59, 61).

[191] A. v. d. Oord, Y. Li, and O. Vinyals. "Representation learning with contrastive predictive coding". In: *arXiv preprint arXiv:1807.03748* (2018) (cit. on p. 59).

[192] K. Osawa et al. "Practical deep learning with Bayesian principles". In: *NeurIPS* (2019) (cit. on p. 32).

[193] Y. Ovadia et al. "Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift". In: *Advances in neural information processing systems* (2019) (cit. on p. 84).

[194] J. Park et al. "Probabilistic representations for video contrastive learning". In: *CVPR*. 2022 (cit. on p. 60).

[195] T. Park et al. "Contrastive learning for unpaired image-to-image translation". In: *ECCV*. 2020 (cit. on pp. 16, 17, 23).

[196] T. B. Parrish et al. "Impact of signal-to-noise on functional MRI". In: *Magnetic Resonance in Medicine* (2000) (cit. on p. 85).

[197] D. Pathak et al. "Context encoders: Feature learning by inpainting". In: *IEEE CVPR*. 2016 (cit. on pp. 31, 32).

[198] T. Pearce et al. "High-quality prediction intervals for deep learning: A distribution-free, ensembled approach". In: *ICML*. 2018 (cit. on p. 73).

[199] B. Phan et al. "Calibrating uncertainties in object localization task". In: *arXiv preprint arXiv:1811.11210* (2018) (cit. on pp. 32, 75).

[200] T. Plotz and S. Roth. "Benchmarking denoising algorithms with real photographs". In: *IEEE CVPR*. 2017 (cit. on pp. 29, 31, 32).

[201] B. A. Plummer et al. "Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models". In: *CVPR*. 2015 (cit. on pp. 58, 59, 64).

[202] J. Portilla and E. P. Simoncelli. "A parametric texture model based on joint statistics of complex wavelet coefficients". In: *IJCV* (2000) (cit. on p. 46).

[203] A. Prabhu, C. Dognin, and M. Singh. "Sampling bias in deep active classification: An empirical study". In: (2019) (cit. on p. 60).

[204] A. Radford et al. "Language Models are Unsupervised Multitask Learners". In: (2019) (cit. on p. 67).

[205] A. Radford et al. "Learning transferable visual models from natural language supervision". In: *ICML*. 2021 (cit. on pp. 57–59, 61, 65, 67, 68).

[206] A. Raj and F. Bach. "Convergence of uncertainty sampling for active learning". In: *ICML*. 2022 (cit. on p. 60).

[207] V. Rangnekar et al. "USIM-DAL: Uncertainty-aware Statistical Image Modeling-based Dense Active Learning for Super-resolution". In: (2023) (cit. on p. 60).

[208] C. Redies, J. Hasenstein, and J. Denzler. "Fractal-like image statistics in visual art: similarity to natural scenes". In: *Spatial vision* (2008) (cit. on p. 46).

[209] S. Reed et al. "Learning Deep Representations of Fine-Grained Visual Descriptions". In: *CVPR*. 2016 (cit. on p. 58).

[210] P. Refregier and F. Goudail. *Statistical Image Processing Techniques for Noisy Images: An Application-Oriented Approach*. Springer US, 2013 (cit. on p. 21).

[211] T. Reiss et al. "Panda: Adapting pretrained features for anomaly detection and segmentation". In: *IEEE CVPR*. 2021 (cit. on p. 42).

[212] C. Riquelme, G. Tucker, and J. Snoek. "Deep bayesian bandits showdown: An empirical comparison of bayesian deep networks for thompson sampling". In: *arXiv preprint arXiv:1802.09127* (2018) (cit. on p. 32).

[213] E. C. Robinson et al. "Identifying population differences in whole-brain structural networks: A machine learning approach". In: *NeuroImage* (2010) (cit. on pp. 17, 22).

[214] E. C. Robinson et al. "Identifying population differences in whole-brain structural networks: a machine learning approach". In: *NeuroImage* (2010) (cit. on pp. 36, 79).

[215] Y. Romano, E. Patterson, and E. Candes. "Conformalized quantile regression". In: *Advances in neural information processing systems* (2019) (cit. on p. 71).

[216] R. Rombach et al. *High-Resolution Image Synthesis with Latent Diffusion Models*. 2022 (cit. on p. 59).

[217] O. Ronneberger, P. Fischer, and T. Brox. "U-net: Convolutional networks for biomedical image segmentation". In: *MICCAI*. 2015 (cit. on p. 9).

[218] O. Ronneberger, P. Fischer, and T. Brox. "U-net: Convolutional networks for biomedical image segmentation". In: *MICCAI*. 2015 (cit. on pp. 36, 79).

[219] T. Rott Shaham et al. "Spatially-Adaptive Pixelwise Networks for Fast Image Translation". In: *CVPR*. 2021 (cit. on p. 15).

[220] N. Roy and A. McCallum. "Toward optimal active learning through monte carlo estimation of error reduction". In: *ICML, Williamstown* (2001) (cit. on p. 46).

[221] S. Roy et al. "Uncertainty-guided source-free domain adaptation". In: *ECCV*. 2022 (cit. on p. 60).

[222] A. Rueda, N. Malpica, and E. Romero. "Single-image super-resolution of brain MR images using overcomplete dictionaries". In: *Medical Image Analysis* (2013) (cit. on p. 7).

[223] C. Saharia et al. "Image super-resolution via iterative refinement". In: *IEEE TPAMI* (2022) (cit. on p. 47).

[224] M. van der Schaar et al. "How artificial intelligence and machine learning can help healthcare systems respond to COVID-19". In: *Machine Learning* (2021) (cit. on p. 30).

[225] C. Schuhmann et al. "Laion-400m: Open dataset of clip-filtered 400 million image-text pairs". In: *arXiv preprint arXiv:2111.02114* (2021) (cit. on pp. 57, 59).

[226] C. Schuhmann et al. "Laion-5b: An open large-scale dataset for training next generation image-text models". In: *arXiv preprint arXiv:2210.08402* (2022) (cit. on p. 59).

[227] W. Schwarting, J. Alonso-Mora, and D. Rus. "Planning and decision-making for autonomous vehicles". In: *Annual Review of Control, Robotics, and Autonomous Systems* (2018) (cit. on p. 30).

[228] P. Seeböck et al. "Exploiting epistemic uncertainty of anatomy segmentation for anomaly detection in retinal OCT". In: *IEEE TMI* (2019) (cit. on p. 17).

[229] O. Sener and S. Savarese. "Active learning for convolutional neural networks: A core-set approach". In: *arXiv preprint arXiv:1708.00489* (2017) (cit. on p. 46).

[230] B. Settles. "Active learning literature survey". In: (2009) (cit. on p. 60).

[231] S. Shafaei et al. "Uncertainty in machine learning: A safety perspective on autonomous driving". In: *International Conference on Computer Safety, Reliability, and Security*. 2018 (cit. on pp. 70, 71).

[232] A. Shapeev et al. "Active learning and uncertainty estimation". In: *Machine Learning Meets Quantum Physics* (2020) (cit. on p. 60).

[233] R. Shaw et al. "MRI k-space motion artefact augmentation: model robustness and task-specific uncertainty". In: *MIDL*. 2019 (cit. on p. 10).

[234] Y. Shi and A. K. Jain. "Probabilistic face embeddings". In: *ICCV*. 2019 (cit. on pp. 60, 62, 65, 113).

[235] H.-C. Shin et al. "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning". In: *IEEE TMI* (2016) (cit. on p. 7).

[236] A. Shocher, N. Cohen, and M. Irani. ""zero-shot" super-resolution using deep internal learning". In: *CVPR*. 2018 (cit. on pp. 45, 47).

[237] E. P. Simoncelli. "4.7 statistical modeling of photographic images". In: *Handbook of Video and Image Processing* (2005) (cit. on pp. 46, 48).

[238] K. Simonyan and A. Zisserman. "Very deep convolutional networks for large-scale image recognition". In: *ICLR* (2015) (cit. on p. 31).

[239] A. Singh et al. "Flava: A foundational language and vision alignment model". In: *CVPR*. 2022 (cit. on pp. 57–59, 61).

[240] J. W. Soh, S. Cho, and N. I. Cho. "Meta-transfer learning for zero-shot super-resolution". In: *CVPR*. 2020 (cit. on p. 45).

[241] H. Song et al. "Distribution calibration for regression". In: *ICML*. 2019 (cit. on p. 73).

[242] Y. Song and M. Soleymani. "Polysemous visual-semantic embedding for cross-modal retrieval". In: *CVPR*. 2019 (cit. on p. 64).

[243] H. Soury and M.-S. Alouini. "New results on the sum of two generalized Gaussian random variables". In: *GlobalSIP*. 2015 (cit. on pp. 63, 113).

[244] N. Srivastava et al. "Dropout: a simple way to prevent neural networks from overfitting". In: *JMLR* (2014) (cit. on p. 36).

[245] A. Stirn et al. "Faithful Heteroscedastic Regression with Neural Networks". In: *arXiv preprint arXiv:2212.09184* (2022) (cit. on pp. 80, 83).

[246] V. P. Sudarshan et al. "Towards lower-dose PET using physics-based uncertainty-aware multimodal learning with robustness to out-of-distribution data". In: *Medical Image Analysis* (2021) (cit. on pp. 17, 29, 47, 60, 79, 80, 85).

[247] J. J. Sun et al. "View-invariant probabilistic embedding for human pose". In: *ECCV*. 2020 (cit. on p. 60).

[248] L. P. Swiler, T. L. Paez, and R. L. Mayes. "Epistemic uncertainty quantification tutorial". In: *Proceedings of the 27th International Modal Analysis Conference*. 2009 (cit. on p. 73).

[249] N. Tagasovska and D. Lopez-Paz. "Single-Model Uncertainties for Deep Learning". In: *NeurIPS* (2019) (cit. on p. 73).

[250] R. Tanno et al. "Bayesian image quality transfer with CNNs: exploring uncertainty in dMRI super-resolution". In: *MICCAI*. 2017 (cit. on pp. 7, 8, 17).

[251] R. Tanno et al. "Uncertainty modelling in deep learning for safer neuroimage enhancement: Demonstration in diffusion MRI". In: *NeuroImage* (2021) (cit. on p. 17).

[252] B. Thomee et al. "YFCC100M: The new data in multimedia research". In: *Communications of the ACM* (2016) (cit. on p. 57).

[253] C. Tian et al. "Deep learning on image denoising: An overview". In: *Neural Networks* (2020) (cit. on pp. 29, 31).

[254] R. Timofte et al. "NTIRE 2018 Challenge on Single Image Super-Resolution: Methods and Results". In: *IEEE CVPR Workshop*. 2018 (cit. on p. 78).

[255] R. J. Townshend et al. "Atom3d: Tasks on molecules in three dimensions". In: *NeurIPS 2021* (2020) (cit. on p. 78).

[256] D. Tran et al. "Plex: Towards reliability using pretrained large model extensions". In: *arXiv preprint arXiv:2207.07411* (2022) (cit. on p. 60).

[257] N.-T. Tran, T.-A. Bui, and N.-M. Cheung. "Dist-gan: An improved gan using distance constraints". In: *ECCV*. 2018 (cit. on pp. 16, 17).

[258] R. Tyleček and R. Šára. "Spatial pattern templates for recognition of objects with regular structure". In: *GCPR*. 2013 (cit. on pp. 17, 21).

[259] U. Upadhyay and S. P. Awate. "A mixed-supervision multilevel GAN framework for image quality enhancement". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2019, pp. 556–564 (cit. on pp. 7, 17, 29, 47).

[260] U. Upadhyay and S. P. Awate. "Robust super-resolution GAN, with manifold-based and perception loss". In: *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*. IEEE. 2019, pp. 1372–1376 (cit. on pp. 7, 17, 29, 47).

[261] U. Upadhyay, Y. Chen, and Z. Akata. "Robustness via Uncertainty-aware Cycle Consistency". In: *NeurIPS* (2021) (cit. on pp. 31, 33, 36, 40, 47, 60, 73, 79, 85).

[262] U. Upadhyay, Y. Chen, and Z. Akata. "Uncertainty-aware Generalized Adaptive CycleGAN". In: *preprint arXiv:2102.11747* (2021) (cit. on pp. 7–9).

[263] U. Upadhyay, V. P. Sudarshan, and S. P. Awate. "Uncertainty-aware GAN with Adaptive Loss for Robust MRI Image Enhancement". In: *ICCV workshop on Computer Vision for Automated Medical Diagnosis*. 2021 (cit. on pp. 17, 31, 47, 60, 71).

[264] U. Upadhyay, V. P. Sudarshan, and S. P. Awate. "Uncertainty-aware gan with adaptive loss for robust mri image enhancement". In: *IEEE ICCV Workshop*. 2021 (cit. on pp. 29, 31, 33, 36, 40, 73, 85).

[265] U. Upadhyay, V. P. Sudarshan, and S. P. Awate. "Uncertainty-aware GAN with adaptive loss for robust MRI image enhancement". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 3255–3264 (cit. on p. 47).

[266] U. Upadhyay, V. P. Sudarshan, and S. P. Awate. "Uncertainty-aware GAN with adaptive loss for robust MRI image enhancement". In: *ICCV-W*. 2021 (cit. on p. 62).

[267] U. Upadhyay et al. "BayesCap: Bayesian Identity Cap for Calibrated Uncertainty in Frozen Neural Networks". In: *European Conference on Computer Vision*. Springer. 2022, pp. 299–317 (cit. on pp. 46, 49, 51, 73).

[268] U. Upadhyay et al. "BayesCap: Bayesian Identity Cap for Calibrated Uncertainty in Frozen Neural Networks". In: *ECCV*. 2022 (cit. on pp. 60, 62).

[269] M. A. Valiuddin et al. "Improving Aleatoric Uncertainty Quantification in Multi-annotated Medical Image Segmentation with Normalizing Flows". In: *Uncertainty for Safe Utilization of Machine Learning in Medical Imaging, and Perinatal Imaging, Placental and Preterm Image Analysis: 3rd International Workshop, UNSURE 2021, and 6th International Workshop, PIPPI 2021, Held in Conjunction with MICCAI 2021, Strasbourg, France, October 1, 2021, Proceedings 3*. Springer. 2021, pp. 75–88 (cit. on p. 73).

[270] H. Van Nguyen, K. Zhou, and R. Vemulapalli. "Cross-Domain Synthesis of Medical Images Using Efficient Location-Sensitive Deep Network". In: *MICCAI*. Ed. by N. Navab et al. 2015 (cit. on p. 17).

[271] G. Varma et al. "IDD: A dataset for exploring problems of autonomous navigation in unconstrained environments". In: *IEEE WACV*. 2019 (cit. on p. 42).

[272] K. R. Varshney and H. Alemzadeh. "On the safety of machine learning: Cyber-physical systems, decision sciences, and data products". In: *Big data* (2017) (cit. on p. 70).

[273] C. Verpoorter et al. "A global inventory of lakes based on high-resolution satellite imagery". In: *Geophysical Research Letters* (2014) (cit. on p. 45).

[274] K. Vougioukas, S. Petridis, and M. Pantic. "DINO: A Conditional Energy-Based GAN for Domain Translation". In: *ICLR*. 2021 (cit. on p. 15).

[275] C. Wah et al. "The caltech-ucsd birds-200-2011 dataset". In: (2011) (cit. on pp. 58, 64).

[276] B. Wang et al. "Adversarial cross-modal retrieval". In: *ACM-MM*. 2017 (cit. on p. 57).

[277] C. Wang et al. "Perceptual adversarial networks for image-to-image transformation". In: *IEEE TIP* (2018) (cit. on pp. 11, 12).

[278] G. Wang et al. "Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks". In: *Neurocomputing* (2019) (cit. on pp. 7, 17, 30–33, 36, 47, 60, 65, 73, 80).

[279] G. Wang et al. "Automatic brain tumor segmentation based on cascaded convolutional neural networks with uncertainty estimation". In: *Frontiers in Computational Neuroscience* (2019) (cit. on p. 7).

[280] G. Wang et al. "Automatic brain tumor segmentation using convolutional neural networks with test-time augmentation". In: *International MICCAI Brainlesion Workshop*. 2018 (cit. on p. 7).

[281] G. Wang et al. "Test-time augmentation with uncertainty estimation for deep learning-based medical image segmentation". In: *MIDL* (2018) (cit. on p. 36).

[282] K. Wang et al. "A comprehensive survey on cross-modal retrieval". In: *arXiv preprint arXiv:1607.06215* (2016) (cit. on p. 57).

[283] K. Wang et al. "Cost-effective active learning for deep image classification". In: *IEEE Transactions on Circuits and Systems for Video Technology* (2016) (cit. on pp. 46, 48).

[284] X. Wang and L. Aitchison. "Bayesian OOD detection with aleatoric uncertainty and outlier exposure". In: *Fourth Symposium on Advances in Approximate Bayesian Inference*. 2021 (cit. on pp. 31, 60).

[285] X. Wang et al. "Esrgan: Enhanced super-resolution generative adversarial networks". In: *ECCVw*. 2018 (cit. on p. 47).

[286] Y. Wang et al. "Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving". In: *IEEE CVPR*. 2019 (cit. on p. 42).

[287] y. wang, L. Yu, and J. van de Weijer. "DeepI2I: Enabling Deep Hierarchical Image-to-Image Translation by Transferring from GANs". In: *NeurIPS*. 2020 (cit. on p. 15).

[288] Z. Wang and J. Ye. "Querying discriminative and representative samples for batch mode active learning". In: *ACM TKDD* (2015) (cit. on p. 46).

[289] Z. Wang et al. "Image quality assessment: from error visibility to structural similarity". In: *IEEE TIP* (2004) (cit. on pp. 10, 22, 37, 52, 79).

[290] Z. Wang et al. "Simvlm: Simple visual language model pretraining with weak supervision". In: *arXiv preprint arXiv:2108.10904* (2021) (cit. on p. 59).

[291] M. Welling and Y. W. Teh. "Bayesian learning via stochastic gradient Langevin dynamics". In: *ICML*. 2011 (cit. on pp. 31, 60, 72).

[292] S. Whitehead et al. "Reliable Visual Question Answering: Abstain Rather Than Answer Incorrectly". In: *ECCV*. 2022 (cit. on p. 60).

[293] H. Wieslander et al. "Deep learning with conformal prediction for hierarchical analysis of large-scale whole-slide tissue images". In: *IEEE journal of biomedical and health informatics* 25.2 (2020), pp. 371–380 (cit. on p. 72).

[294] N. Wiest-Daesslé et al. "Rician noise removal by non-local means filtering for low signal-to-noise ratio MRI: applications to DT-MRI". In: *International Conference on Medical Image Computing and Computer-assisted Intervention*. 2008 (cit. on p. 85).

[295] J. M. Wolterink et al. "Deep MR to CT synthesis using unpaired data". In: *International workshop on Simulation and Synthesis in Medical Imaging*. 2017 (cit. on p. 7).

[296] S. Woo et al. "Cbam: Convolutional block attention module". In: *ECCV*. 2018 (cit. on p. 47).

[297] M. Wortsman et al. "Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time". In: *ICML*. 2022 (cit. on p. 68).

[298] M. Wortsman et al. "Robust fine-tuning of zero-shot models". In: *CVPR*. 2022 (cit. on p. 68).

[299] B. Xie et al. "Towards fewer annotations: Active learning via region impurity and prediction uncertainty for domain adaptive semantic segmentation". In: *CVPR*. 2022 (cit. on p. 60).

[300] S. Xie and Z. Tu. "Holistically-nested edge detection". In: *ICCV*. 2015 (cit. on p. 17).

[301] M. Xu et al. "A simple baseline for zero-shot semantic segmentation with pre-trained vision-language model". In: *arXiv preprint arXiv:2112.14757* (2021) (cit. on p. 59).

[302] W. Xu et al. "Motion planning under uncertainty for on-road autonomous driving". In: *IEEE ICRA*. 2014 (cit. on pp. 42, 71).

[303] X. Yan et al. "Parsimonious quantile regression of financial asset tail dynamics via sequential learning". In: *Advances in neural information processing systems* (2018) (cit. on p. 71).

[304] D. Yang et al. "Remote sensing image super-resolution: Challenges and approaches". In: *IEEE DSP*. 2015 (cit. on p. 45).

[305] G. Yang et al. "Probabilistic modeling of semantic ambiguity for scene graph generation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 12527–12536 (cit. on p. 58).

[306] Q. Yang et al. "MRI cross-modality image-to-image translation". In: *Scientific reports* (2020) (cit. on p. 17).

[307] Q. Yang et al. "Low-dose CT image denoising using a generative adversarial network with Wasserstein distance and perceptual loss". In: *IEEE TMI* (2018) (cit. on p. 7).

[308] Y. Yang and M. Loog. "Active learning using uncertainty information". In: *ICPR*. 2016 (cit. on p. 60).

[309] Y. Yang et al. "Multi-class active learning by uncertainty sampling with diversity maximization". In: *IJCV* (2015) (cit. on pp. 46, 60).

[310] L. Yao et al. "FILIP: fine-grained interactive language-image pre-training". In: *arXiv preprint arXiv:2111.07783* (2021) (cit. on p. 59).

[311] C. Ye, Y. Li, and X. Zeng. "An improved deep network for tissue microstructure estimation with uncertainty quantification". In: *Medical image analysis* (2020) (cit. on p. 17).

[312] D. H. Ye et al. "Modality propagation: coherent synthesis of subject-specific scans with data-driven regularization". In: *MICCAI*. 2013 (cit. on pp. 36, 40).

[313] X. Yi, E. Walia, and P. Babyn. "Generative adversarial network in medical imaging: A review". In: *Medical Image Analysis* (2019) (cit. on p. 7).

[314] B. Yu et al. "Ea-GANs: edge-aware generative adversarial networks for cross-modality MR image synthesis". In: *IEEE TMI* (2019) (cit. on pp. 36, 40).

[315] J. Yu et al. "Coca: Contrastive captioners are image-text foundation models". In: *arXiv preprint arXiv:2205.01917* (2022) (cit. on p. 59).

[316] J. Yu et al. "Free-form image inpainting with gated convolution". In: *IEEE CVPR*. 2019 (cit. on pp. 36, 37, 39, 40).

[317] J. Yu et al. "Generative image inpainting with contextual attention". In: *IEEE CVPR*. 2018 (cit. on pp. 29, 31, 36, 37).

[318] X. Yu, G. Franchi, and E. Aldea. "SLURP: Side learning uncertainty for regression problems". In: *BMVC*. 2021 (cit. on p. 60).

[319] Y. Yu, H. Sajjad, and J. Xu. "Learning Uncertainty for Unknown Domains with Zero-Target-Assumption". In: *ICLR*. 2023 (cit. on p. 60).

[320] R. Zeyde, M. Elad, and M. Protter. "On single image scale-up using sparse-representations". In: *International conference on curves and surfaces*. 2010 (cit. on p. 36).

[321] H. Zhang, V. Sindagi, and V. M. Patel. "Image de-raining using a conditional generative adversarial network". In: *IEEE Transactions on Circuits and Systems for Video Technology* (2019) (cit. on pp. 7, 11).

[322] J. Zhang, U. Norinder, and F. Svensson. "Deep learning-based conformal prediction of toxicity". In: *Journal of chemical information and modeling* 61.6 (2021), pp. 2648–2657 (cit. on p. 72).

[323] J. Zhang, B. Kailkhura, and T. Y.-J. Han. "Mix-n-match: Ensemble and compositional methods for uncertainty calibration in deep learning". In: *ICML*. 2020 (cit. on p. 32).

[324] K. Zhang, L. V. Gool, and R. Timofte. "Deep Unfolding Network for Image Super-Resolution". In: *CVPR*. 2020 (cit. on p. 15).

[325] P. Zhang et al. “Vinvl: Revisiting visual representations in vision-language models”. In: *CVPR*. 2021 (cit. on p. 59).

[326] R. Zhang, P. Isola, and A. A. Efros. “Colorful image colorization”. In: *ECCV*. 2016 (cit. on pp. 29, 32).

[327] Z. Zhang et al. “Reducing uncertainty in undersampled mri reconstruction with active acquisition”. In: *IEEE CVPR*. 2019 (cit. on pp. 8, 51).

[328] H. Zhao, H. Li, and L. Cheng. “Synthesizing filamentary structured images with gans”. In: *preprint arXiv:1706.02185* (2017) (cit. on p. 11).

[329] C. Zheng, T.-J. Cham, and J. Cai. “The Spatially-Correlative Loss for Various Image Translation Tasks”. In: *CVPR*. 2021 (cit. on p. 16).

[330] L. Zhong et al. “Predict CT image from MRI data using KNN-regression with learned local descriptors”. In: *IEEE ISBI*. 2016 (cit. on p. 7).

[331] B. Zhou et al. “Places: A 10 million Image Database for Scene Recognition”. In: *IEEE TPAMI* (2017) (cit. on pp. 36, 42).

[332] C. Zhou and R. C. Paffenroth. “Anomaly detection with robust deep autoencoders”. In: *ACM KDD*. 2017 (cit. on p. 42).

[333] K. Zhou et al. “Conditional prompt learning for vision-language models”. In: *CVPR*. 2022 (cit. on p. 59).

[334] K. Zhou et al. “Learning to Prompt for Vision-Language Models”. In: *IJCV* (2022) (cit. on p. 59).

[335] Q. Zhou et al. “Bayesian inference and uncertainty quantification for medical image reconstruction with Poisson data”. In: *SIAM Journal on Imaging Sciences* (2020) (cit. on p. 17).

[336] T. Zhou et al. “Estimating uncertainty intervals from collaborating networks”. In: *Journal of Machine Learning Research* (2021) (cit. on pp. 71, 72, 79, 81, 83).

[337] W. Zhou et al. “PatternNet: A benchmark dataset for performance evaluation of remote sensing image retrieval”. In: *ISPRS journal of photogrammetry and remote sensing* (2018) (cit. on p. 51).

[338] J.-Y. Zhu et al. “Unpaired image-to-image translation using cycle-consistent adversarial networks”. In: *IEEE ICCV*. 2017 (cit. on pp. 9, 16–18, 20, 21, 23).

[339] Y. Zhu et al. “Cross-domain medical image translation by shared latent Gaussian mixture model”. In: *MICCAI*. 2020 (cit. on p. 29).

A

Publications

This thesis is based on the following publications. An overview of contributions can be found in Section 1.2. Asterisks (*) indicate shared first-author publications.

- "Uncertainty-Guided Progressive GANs for Medical Image Translation" Published in International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI) - 2021. *Uddeshya Upadhyay, Yanbei Chen, Tobias Hepp, Sergios Gatidis, Zeynep Akata*

- "Robustness via Uncertainty-aware Cycle Consistency" Published in Conference on Neural Information Processing Systems (NeurIPS) - 2021. *Uddeshya Upadhyay, Yanbei Chen, Zeynep Akata*

- "BayesCap: Bayesian Identity Cap for Calibrated Uncertainty in Frozen Neural Networks" Published in European Conference on Computer Vision (ECCV) - 2022. *Uddeshya Upadhyay*, Shyamgopal Karthik*, Yanbei Chen, Massimiliano Mancini, Zeynep Akata*

- "USIM-DAL: Uncertainty-aware Statistical Image Modeling-based Dense Active Learning for Super-resolution" Published in Conference on Uncertainty in Artificial Intelligence (UAI) - 2023. *Vikrant Rangnekar*, Uddeshya Upadhyay*, Zeynep Akata, Biplab Banerjee*

- "ProbVLM: Probabilistic Adapter for Frozen Vision-Language Models" Published in International Conference on Computer Vision (ICCV) - 2023. *Uddeshya Upadhyay*, Shyamgopal Karthik*, Massimiliano Mancini, Zeynep Akata*

- "Likelihood Annealing: Fast Calibrated Uncertainty for Regression". *Uddeshya Upadhyay, Jae Myung Kim, Cordelia Schmid, Bernhard Schölkopf, Zeynep Akata*

B

Appendix - ProbVLM: Probabilistic Adapter for Frozen Vision-Language Models

B.1 Additional Theoretical Support

We discuss Equation 4 from the main paper and how we simplify the same to obtain a loss function suitable for training deep learning models. Given an image and text embedding pair $(\mathbf{z}_\mathcal{V}, \mathbf{z}_\mathcal{T})$ (from frozen model) representing similar concepts, the output distributions from $\mathbf{\Psi}(\cdot;\zeta)$, $\mathcal{G}(\mathbf{z};\hat{\mathbf{z}}_\mathcal{V}, \hat{\alpha}_\mathcal{V}, \hat{\beta}_\mathcal{V})$ and $\mathcal{G}(\mathbf{z};\hat{\mathbf{z}}_\mathcal{T}, \hat{\alpha}_\mathcal{T}, \hat{\beta}_\mathcal{T})$ (later referred to as $\mathcal{G}_\mathcal{V}(\mathbf{z})$) and $\mathcal{G}_\mathcal{T}(\mathbf{z})$) should match. This can be measured directly from the likelihood as, $p(\mathbf{z}_v = \mathbf{z}_u)$, where $\mathbf{z}_v \sim \mathcal{G}_\mathcal{V}(\mathbf{z})$ and $\mathbf{z}_u \sim \mathcal{G}_\mathcal{T}(\mathbf{z})$ as in [234] , i.e.,

$$p(\mathbf{z}_v = \mathbf{z}_u) := \iint \mathcal{G}_\mathcal{V}(\mathbf{z}_v)\mathcal{G}_\mathcal{T}(\mathbf{z}_u)\delta(\mathbf{z}_v - \mathbf{z}_u)d\mathbf{z}_v d\mathbf{z}_u \tag{B.1}$$

where $\delta(\cdot)$ refers to the *Dirac-delta distribution*. The above integral can be simplified further by defining $\Delta\mathbf{z} = \mathbf{z}_v - \mathbf{z}_u$ and seeking $p(\Delta\mathbf{z}) = 0$. As both $\mathbf{z}_v$ and $\mathbf{z}_u$ are GGD random variables, $\Delta\mathbf{z}$ follows the distribution based on the *Bivariate Fox H-function* [243] given by,

$$\begin{gathered}\Delta z \sim \tfrac{1}{2\Gamma(1/\hat{\beta}_\mathcal{V}),\Gamma(1/\hat{\beta}_\mathcal{T})}\times \\ \int \mathcal{H}_{1,2}^{1,1}\left[At^2\middle|\begin{matrix}(1-\frac{1}{\hat{\mathbf{z}}_\mathcal{V}},\frac{1}{\hat{\mathbf{z}}_\mathcal{T}})\\(0,1)(\frac{1}{2},1)\end{matrix}\right]\mathcal{H}_{1,2}^{1,1}\left[Bt^2\middle|\begin{matrix}(1-\frac{1}{\hat{\mathbf{z}}_\mathcal{T}},\frac{1}{\hat{\mathbf{z}}_\mathcal{T}})\\(0,1)(\frac{1}{2},1)\end{matrix}\right]\cos t(\mu - z)dt\end{gathered} \tag{B.2}$$

Where $A = \frac{\hat{\alpha}_\mathcal{V}^2\Gamma(1/\hat{\beta}_\mathcal{V})}{4\Gamma(3/\hat{\beta}_\mathcal{V})}$, $B = \frac{\hat{\alpha}_\mathcal{T}^2\Gamma(1/\hat{\beta}_\mathcal{T})}{4\Gamma(3/\hat{\beta}_\mathcal{T})}$, $\mu = \hat{\mathbf{z}}_v - \hat{\mathbf{z}}_u$, and $\mathcal{H}$ is the *Fox H function* [243] given by,

$$\begin{aligned}&H_{p,q}^{m,n}\left[z \mid \begin{matrix}(a_1,A_1) & (a_2,A_2) & \dots & (a_p,A_p)\\(b_1,B_1) & (b_2,B_2) & \dots & (b_q,B_q)\end{matrix}\right]\\ &= \frac{1}{2\pi i}\int_L \frac{\prod_{j=1}^{m}\Gamma\left(b_j+B_j s\right)\prod_{j=1}^{n}\Gamma\left(1-a_j-A_j s\right)}{\prod_{j=m+1}^{q}\Gamma\left(1-b_j-B_j s\right)\prod_{j=n+1}^{p}\Gamma\left(a_j+A_j s\right)}z^{-s}ds\end{aligned} \tag{B.3}$$

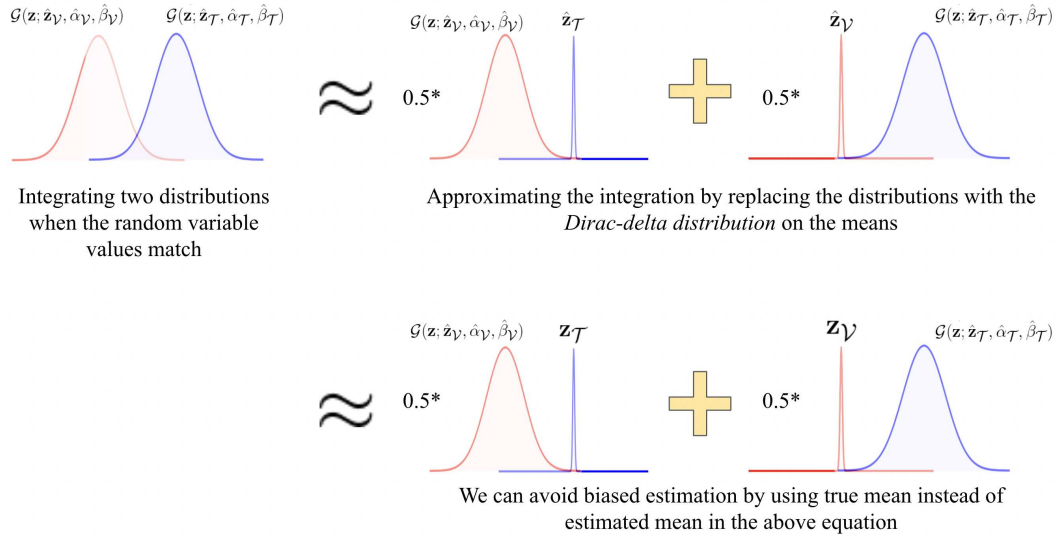

Figure B.1: Visualizing the approximation in Equation B.4.

Equation B.2 does not provide a scalable objective function suitable for training deep neural networks. Hence, we propose an approximation that is easily scalable for deep-learning models given by,

$$
\begin{aligned}
p(\mathbf{z}_v = \mathbf{z}_u) &= \iint \mathcal{G}_\mathcal{V}(\mathbf{z}_v)\mathcal{G}_\mathcal{T}(\mathbf{z}_u)\delta(\mathbf{z}_v - \mathbf{z}_u)d\mathbf{z}_v d\mathbf{z}_u \\
&\approx \int \frac{1}{2}\left(\mathcal{G}_\mathcal{V}(\mathbf{z})\delta(\mathbf{z} - \mathbf{z}_\mathcal{T}) + \mathcal{G}_\mathcal{T}(\mathbf{z})\delta(\mathbf{z} - \mathbf{z}_\mathcal{V})\right) d\mathbf{z}
\end{aligned} \tag{B.4}
$$

To understand the above approximation, we refer to Figure B.1. We notice that the integral in Equation B.1 tries to convolve the two distribution, with an additional constraint of those distributions being equal in value. While convolving the two generalized gaussian distributions is hard, Figure B.1 shows that a rough approximation for the same is to convolve a generalized gaussian distribution with the Dirac-delta distribution. Further, instead of using the estimated means from `ProbVLM` in the Dirac-delta distribution (that are to be near-perfect reconstructions of the embeddings obtained from the frozen network), we use the embeddings from the frozen encoders as shown in Figure B.1. This finally leads to Equation B.4. The first term in the integral, $\int \mathcal{G}_\mathcal{V}(\mathbf{z})\delta(\mathbf{z} - \mathbf{z}_\mathcal{T})d\mathbf{z}$, is the likelihood of the text embedding $\mathbf{z}_\mathcal{T}$ under the predicted distribution, $\mathcal{G}_\mathcal{V}(\mathbf{z})$, for the visual embedding. Similarly, the second term is the likelihood of the visual embedding $\mathbf{z}_\mathcal{V}$ under the predicted distribution, $\mathcal{G}_\mathcal{T}(\mathbf{z})$, for the text embedding. Negative log of Equation B.4 leads to a scalable objective function that can be used to learn the optimal parameters for vision and text

components of `ProbVLM` ($\mathbf{\Psi}_{\mathcal{V}}(\cdot;\zeta_{\mathcal{V}})$ and $\mathbf{\Psi}_{\mathcal{T}}(\cdot;\zeta_{\mathcal{T}})$),

$$L_{\text{cross}}(\zeta_{\mathcal{V}},\zeta_{\mathcal{T}}) := \underbrace{\left(\frac{|\hat{\mathbf{z}}_{\mathcal{V}}-\mathbf{z}_{\mathcal{T}}|}{\hat{\alpha}_{\mathcal{V}}}\right)^{\hat{\beta}_{\mathcal{V}}} - \log\frac{\hat{\beta}_{\mathcal{V}}}{\hat{\alpha}_{\mathcal{V}}} + \log\Gamma(\frac{1}{\hat{\beta}_{\mathcal{V}}})+}_{\text{Cross-modal: vision}\rightarrow\text{text}}$$
$$\underbrace{\left(\frac{|\hat{\mathbf{z}}_{\mathcal{T}}-\mathbf{z}_{\mathcal{V}}|}{\hat{\alpha}_{\mathcal{T}}}\right)^{\hat{\beta}_{\mathcal{T}}} - \log\frac{\hat{\beta}_{\mathcal{T}}}{\hat{\alpha}_{\mathcal{T}}} + \log\Gamma(\frac{1}{\hat{\beta}_{\mathcal{T}}})}_{\text{Cross-modal: text}\rightarrow\text{vision}} \tag{B.5}$$

In practice, the exponential of β in the above equation often makes training unstable. To make it more stable, we make use of the Taylor-series expansion and note that

$$\left(\frac{|\hat{\mathbf{z}}-\mathbf{z}|}{\hat{\alpha}}\right)^{\hat{\beta}} = \left(1+(\frac{|\hat{\mathbf{z}}-\mathbf{z}|}{\hat{\alpha}}-1)\right)^{\hat{\beta}}$$
$$\approx 1-\hat{\beta}+\hat{\beta}(\frac{|\hat{\mathbf{z}}-\mathbf{z}|}{\hat{\alpha}}) \tag{B.6}$$

This way, the variable $\hat{\beta}$ no longer in exponent and as a result loss becomes more stable during optimization.

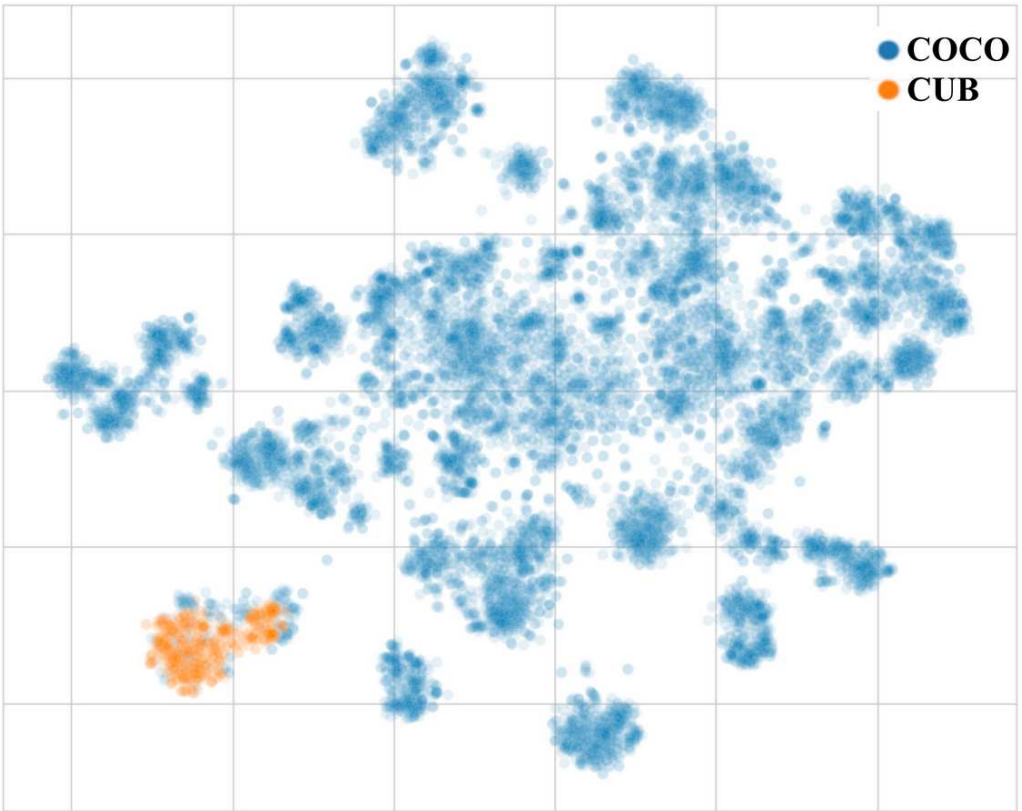

Figure B.2: tSNE plot for MS-COCO and CUB image embeddings illustrating the diversity of MS-COCO.

		Datasets											
		CUB			Flowers			Flickr			COCO		
M		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
V-B32		35.3	64.9	79.3	54.5	84.7	94.0	79.0	94.7	97.1	50.6	75.0	83.6
	i2t	**85.1**	89.4	81.9	53.3	55.2	37.2	64.2	61.0	55.1	61.0	62.3	57.2
		92.1	95.0	90.1	69.6	70.6	52.3	77.0	73.6	68.8	75.8	76.5	73.3
		16.0	34.4	44.6	25.5	47.8	61.8	56.5	82.2	88.3	30.1	55.7	66.8
	t2i	**63.9**	63.0	60.5	37.3	33.5	31.7	36.2	35.5	35.1	35.9	36.9	35.4
		72.8	71.8	70.7	47.4	43.3	43.7	47.5	46.9	46.7	47.2	49.3	47.8
V-B16		34.2	66.2	80.4	52.1	82.8	91.6	82.7	96.2	98.9	53.0	77.1	85.1
	i2t	**85.1**	89.4	81.9	53.3	55.2	37.2	64.2	61.0	55.1	61.0	62.3	57.2
		92.1	95.0	90.1	69.6	70.6	52.3	77.0	73.6	68.8	75.8	76.5	73.3
		15.0	33.3	44.1	25.4	46.4	57.9	61.0	84.2	89.6	33.3	58.6	68.9
	t2i	**63.9**	63.0	60.5	37.3	33.5	31.7	36.2	35.5	35.1	35.9	36.9	35.4
		72.8	71.8	70.7	47.4	43.3	43.7	47.5	46.9	46.7	47.2	49.3	47.8
RN-50		31.1	61.7	75.9	53.0	87.1	95.0	77.7	95.2	97.3	49.1	72.5	81.8
	i2t	**85.1**	89.4	81.9	53.3	55.2	37.2	64.2	61.0	55.1	61.0	62.3	57.2
		92.1	95.0	90.1	69.6	70.6	52.3	77.0	73.6	68.8	75.8	76.5	73.3
		15.3	35.0	46.5	31.5	54.3	66.7	55.1	81.2	87.9	28.3	53.1	64.3
	t2i	**63.9**	63.0	60.5	37.3	33.5	31.7	36.2	35.5	35.1	35.9	36.9	35.4
		72.8	71.8	70.7	47.4	43.3	43.7	47.5	46.9	46.7	47.2	49.3	47.8

Table B.1: Zero-shot performance on COCO, Flickr, CUB and FLO with for both Image-to-Text (i2t) and Text-to-Image (t2i) Retrieval for CLIP Models (M) with Vision Transformer (V-B32, V-B16) and ResNet (RN-50) backbones.

B.2 Additional Quantitative Experiments

We provide the zero-shot results for the CLIP model trained with different visual backbones in Table B.1, while the results after fine-tuning are presented in Table B.2. While Zero-Shot CLIP achieves promising results on all four datasets, these are much worse when compared to the results obtained when fine-tuning on the desired target dataset (42.3 vs. 15.0 for a ViT B/16 on CUB t2i R@1). However, this comes at the cost of worse performance on the remaining datasets due to catastrophic forgetting and has to be mitigated via several strategies.

Figure B.2 shows the tSNE plots for the CLIP embeddings obtained from a relatively diverse dataset (e.g., COCO) compared to a niche dataset (e.g., CUB consisting of only birds). As indicated in the plot, a niche dataset will likely not be able to capture all the representations spread in the embedding space, leading to poor generalization, as shown

		CLIP backbones fine-tuned on											
		CUB			**Flowers**			**Flickr**			**COCO**		
D		V-B32	V-B16	RN-50	V-B32	V-B16	RN-50	V-B32	V-B16	RN-50	V-B32	V-B16	RN-50
CUB	**i2t**	**58.8**	**66.1**	**53.9**	25.2	23.8	13.4	32.4	31.1	26.2	31.5	32.5	26.8
	t2i	**41.3**	**42.3**	**37.4**	18.4	16.8	13.1	16.6	17.1	16.1	16.6	16.9	14.3
Flowers	**i2t**	54.5	51.1	44.3	**80.7**	**82.0**	**73.8**	49.5	55.2	49.7	47.9	47.2	43.6
	t2i	25.5	31.2	29.6	**57.8**	**59.0**	**53.3**	31.3	29.3	30.8	28.7	29.2	31.7
Flickr	**i2t**	68.9	73.5	48.2	51.4	62.4	24.4	**90.0**	**92.7**	**87.1**	86.7	90.2	87.7
	t2i	48.6	54.7	31.4	32.3	40.5	17.0	**73.4**	**77.5**	**68.3**	69.9	74.5	68.7
COCO	**i2t**	32.6	42.6	22.0	24.8	31.8	8.9	56.9	61.5	52.0	**73.4**	**69.5**	**64.3**
	t2i	19.5	27.1	12.5	32.3	19.7	6.8	38.7	43.9	33.0	**49.8**	**52.3**	**45.3**

Table B.2: Result for fine-tuning CLIP on different Datasets (D) for Image-to-Text (i2t) and Text-to-Image (t2i) retrieval.

in Table B.2. This is because CUB has images that only contain birds, whereas COCO is a much larger dataset containing 80 different object categories (including birds). Therefore, fine-tuning either the VLM or `ProbVLM` on a larger, more diverse dataset such as COCO would lead to better generalization and transferability across datasets.