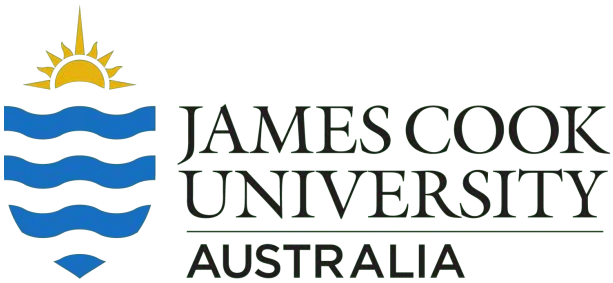


DOCTORAL THESIS

Explainable Deep Learning for Image Understanding

Lu YU

A thesis submitted in fulfillment of the requirements
for the degree of Doctor of Philosophy

in the

College of Science and Engineering

August 28, 2024

Acknowledgements

I would like to extend my heartfelt appreciation to my supervisors, Prof. Wei Xiang and Dr. Eric Wang, for their invaluable guidance and scholarly mentorship throughout my PhD journey. Their expertise, encouragement, and commitment to academic excellence have been instrumental in shaping the trajectory of my research.

My heartfelt thanks go to my primary supervisor, Prof. Wei Xiang, for his insightful feedback, constructive criticism, and the intellectual rigor he brought to our collaborative efforts. Prof. Wei Xiang not only guided my academic pursuits but also played an instrumental role in various aspects of my life, providing crucial financial support during challenging times. He provided me with the freedom to explore topics of interest, imparting visionary thoughts and broad knowledge that inspired me to delve into the fundamental problems within my research. His diverse perspectives significantly enriched the depth and breadth of my research. And his support created a nurturing academic environment that fostered collaboration and intellectual growth.

I would like to express my thanks to the member of my research group, Dr. Kang Han, for his valuable insights, constructive critiques, and the time he dedicated to evaluating and improving my research. I would also want to acknowledge the countless friends who provided encouragement, shared laughter during stressful times, and offered a sense of community throughout this journey. Your friendship has been a source of strength and joy.

I am indebted to my family for their unwavering encouragement, love, and understanding. The success of my work would not have been possible without the incredible support of my husband, who believed in me, encouraged me, and always stood by me at all times. Their support provided the emotional foundation necessary to navigate the demanding rigors of my PhD program.

Statement of the Contribution of Others

Nature of Assistance	Contribution	Names
Supervision	Primary supervision Secondary supervision	Prof. Wei Xiang Dr. Eric Wang
Intellectual support	Conceptualization Data analysis Paper/thesis revision Proofreading	Prof. Wei Xiang Dr. Eric Wang
Financial support	Tuition fee scholarship	James Cook University
Experiment	GPU resource Data storage	High Performance Computing at James Cook University
Infrastructure	Working office Computer	La Trobe University

Abstract

Explainable Artificial Intelligence (XAI) is a burgeoning research domain dedicated to aiding users in comprehending, trusting, and managing AI systems. Fundamentally, XAI strives to align with two core principles: enhancing human understanding and collaboration with AI, and empowering AI systems to augment human capabilities. XAI algorithms aim to provide insights into the underlying decision-making processes inherent in AI models. This not only enables the establishment of trust but also facilitates the identification of any unintended correlations that the network might have acquired to make its decisions. However, explanations produced by existing XAI algorithms are not always well-defined, some of these algorithms produce explanations that are difficult to comprehend, while others emphasize information from irrelevant or noisy regions. Novel algorithms dealing with these challenges will facilitate the applications of deep models for safety-critical applications.

This thesis proposes four novel contributions that address these challenges in improving the explainability of AI models across several domains. Our methods can handle various challenging real-world problems in computer vision tasks, such as handling inaccurate or noisy explanations, accommodating existing black-box models, and facilitating interactive explanations.

As the first contribution, we first analyze the reason for inaccurate/noisy explanations and describe a discovery that the utilization of intermediate features in a model with multi-scale fusion will improve the quality of explanations. We then build a novel explainable model with the proposed dual-attention module to learn and discover class discriminative and interpretable representations. We show that the proposed model is able to achieve state-of-the-art performance in terms of accurate explanations. The second contribution is an explainable vision transformer with pixel-wise attention. To enable richer representations of interpretable attention maps that align with input patterns, a set of attribute features for the target object is learned. In addition, a novel attribute-guided loss to facilitate the learning process in a self-supervised manner is introduced. This loss implicitly adds the regularization to force the representations to focus on various attributes of each target class through the attribute discriminability mechanism and attribute diversity mechanism. Simulation results are presented to illustrate that the proposed model achieves comparable performance to supervised

baselines, while surpassing the accuracy and interpretability of state-of-the-art black-box methods. Our third contribution is to propose an explainable model pruning technique to incorporate explainability into large well-trained models. We first introduce an explainability-aware mask for each prunable unit to quantify its contribution to predicting each class. Specifically, the proposed mask is fully differentiable and can be learned in an end-to-end manner. We demonstrate many benefits of the proposed mask, including more accurate pruning and fewer computational costs compared with existing black-box pruning methods. Then, this thesis describes how to learn the layer-wise pruning thresholds that differentiate the important and less-important units via a differentiable pruning operation. Experimental results on various models are provided to demonstrate the efficacy of the proposed method. The fourth contribution presents an interactive explanation method to edit generative models to obtain high-quality results. Given the overwhelming popularity of text prompting in numerous Generative AI scenarios, how to effectively support such inputs with explanation and guidance presents a significant challenge. End-users often lack knowledge about the quality of the text prompt they use to obtain the desired results from generative AI. We first propose the multi-view score consistency method to enable 3D editing by use of a diffusion prior, which is effective in providing additional supervision signals for learning 3D-consistent geometry. Then, we leverage the diffusion process to learn semantic representations and better edit a scene that faithfully aligns with the information of the text prompt. Lastly, experimental results on various real-world datasets are presented to illustrate the efficacy of the proposed framework across a range of text prompts.

In summary, these contributions push the boundaries of explanation approaches, paving the way for new avenues of progress in computer vision tasks. By addressing the limitations inherent in prior methods, this thesis lays the groundwork for enhancing the robustness and explainability of real-world applications.

List of Publications

The following publications were produced during the period of candidature:

[1] **L. Yu**, W. Xiang, J. Fang, Y. P. Chen, and R. Zhu, "A novel explainable neural network for Alzheimer's disease diagnosis," *Pattern Recognition*, vol. 131, pp. 1-12, Jun. 2022 (IF = 8.0). Related to Chapter 3.

[2] **L. Yu**, W. Xiang, J. Fang, Y. P. Chen, and L. Chi, "eX-ViT: A Novel eXplainable Vision Transformer for Weakly Supervised Semantic Segmentation," *Pattern Recognition*, vol. 142, pp. 1-13, Oct. 2023 (IF = 8.0). Related to Chapter 4.

[3] **L. Yu**, and W. Xiang, "X-Pruner: eXplainable Pruning for Vision Transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Vancouver, Canada, Jun. 2023, pp. 24355-24363. Related to Chapter 5.

[4] **L. Yu**, W. Xiang, and K. Han, "Edit-DiffNeRF: Editing 3D Neural Radiance Fields using 2D Diffusion Model," https://arxiv.org/abs/2306.09551. Related to Chapter 6.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Explainable Artificial Intelligence (XAI) is an emerging research area that aims to help users understand, trust, and manage AI systems [1]. At its core, XAI seeks to develop with two of the primary tenets: humans can better understand and collaborate with AI, while AI systems can empower humans and augment our capabilities [2]. From skilled Go players to autonomous vehicles, AI has realized capabilities reminiscent of science fiction from the past decade. These advancements showcase the remarkable progress in the AI field, pushing the boundaries beyond what was once deemed speculative. Despite substantial advancements, unlocking the full transformative potential promised by AI systems to our society remains elusive. The rising utilization of black-box Machine Learning (ML) models for crucial predictions in various contexts has led to an escalating demand for transparency from diverse stakeholders in AI. With the increasing prevalence of AI technologies, there is a growing necessity for users to understand AI, driven by the unpredictable nature of AI models and the potential consequences, particularly in critical domains such as healthcare, finance, self-driving cars, and law enforcement [3].

The primary objective within XAI domain is the development of reliable explainable algorithms capable of unraveling the complexities associated with black-box models. These algorithms aim to provide insights into the underlying decision-making processes inherent in deep neural network models. For instance, in scenarios where an AI system is utilized to detect malignant tumors from CT scans, it becomes crucial for medical professionals to understand the rationale guiding the decision-making process [5]. This involves a meticulous examination of the image regions detected by the AI system in the diagnostic process. Fig. 1.1 shows DARPA's conceptualization of explainable AI and the needs of a user that the model should address when explaining itself. This not only enables the establishment of trust but also facilitates the identification of any unintended correlations that the network might have acquired to make its

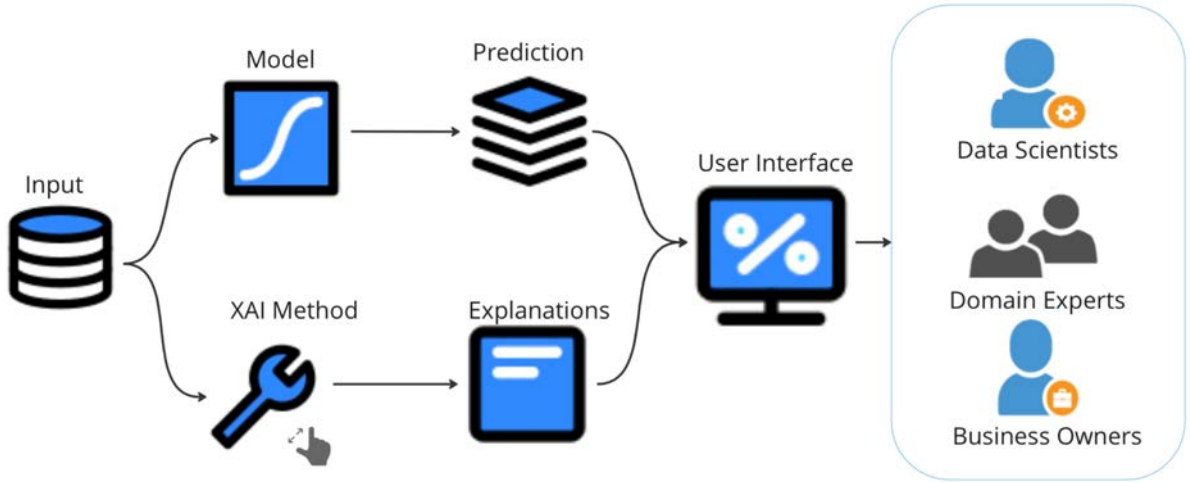


Figure 1.1: Explainable AI, as conceptualized in a DARPA program report [4].

decisions. Over the last few decades, the computer vision community has witnessed various endeavors in this direction, as elaborated in the review in [3].

Recent advances in XAI have led to its integration with various advanced AI techniques, enhancing the applicability and reliability of AI models across diverse domains. For instance, XAI has been combined with federated learning in 6G systems to develop secure and automated vehicular networks [6], [7]. The incorporation of XAI into FL models is anticipated to offer substantial benefits by facilitating decentralized, lightweight, and communication-efficient intelligence. In the cybersecurity domain, XAI outputs have proven useful for generating adversarial and poison samples designed to evade underlying classifiers [8]. Concurrently, defensive strategies such as focused data sampling [9], [10] and model regularization [11] have been proposed to counteract these attacks. Overall, these advancements underscore the critical role of XAI in enhancing both the functionality and security of AI systems, paving the way for more resilient and trustworthy applications across various domains.

The recent development of Generative AI, encompassing large language models (LLMs) [12] and visual generation techniques [13], [14], promises to revolutionize various human tasks. This interest is escalating with the availability of generative AI tools such as ChatGPT [15], DALL.E 2 [14], Github Copilot [16], and others. For example, OpenAI's Codex [17], an LLM capable of generating functional code snippets, adopting a scenario-based approach to understand developers' requirements for explanations when employing Generative AI in diverse programming situations, including natural language to code, code translation, and code auto-completion. However, there is a limited body of work addressing how to combine explainability with generative AI models in the field of computer vision. For example, an explanation could be necessary when specific prompts fail to yield desired results. A successful explanation should empower users to refine their prompts, leading to higher satisfaction with the newly

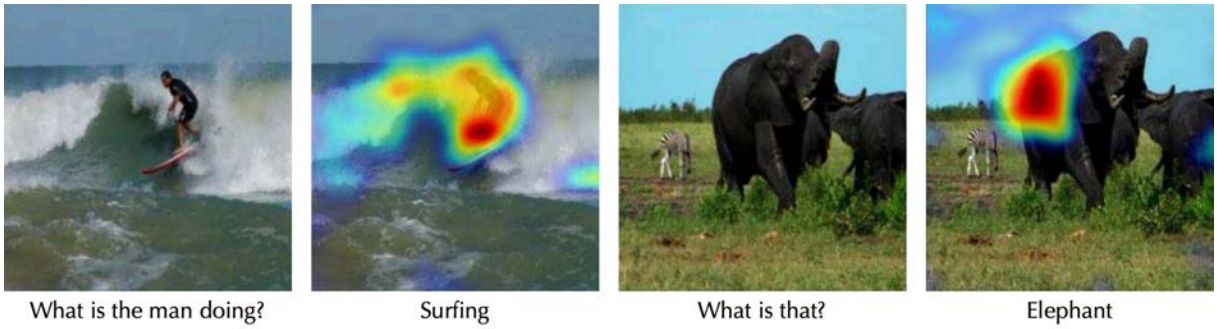


Figure 1.2: Explanation heatmaps obtained using Grad-CAM [19] corresponding to the predicted category for an image.

generated contents. Thus, effective approaches that can address these challenges are desired to facilitate wide-ranging generative applications.

1.2 Challenges and Research Gaps

An intriguing avenue for advancing the understanding of deep neural networks involves developing models that autonomously generate explanations for their decisions. This approach addresses the challenge of reconciling high-performance yet opaque black-box models with lower-performing but interpretable models, a trade-off known as the accuracy-interpretability trade-off. This trade-off is pivotal as it ensures that AI systems not only excel in performance but also provide comprehensible and justifiable decisions [18]. In many practical scenarios, there exists a dilemma between achieving high precision, often associated with complex and opaque systems like deep neural networks, and ensuring that these models' decisions are understandable and explainable to humans. Models that are intrinsically explainable in visual analysis offer professionals and users a deeper understanding and confidence in utilizing deep learning-based systems. However, there is a notable scarcity of research exploring the effectiveness of explainable models in real-world applications. A common approach in current works is to design explainable models by leveraging feature selection techniques to concentrate on a subset of relevant features. Nevertheless, these manual selection methods introduce bias by favoring certain features over others, potentially overlooking crucial factors contributing to the model's predictions.

Meanwhile, while post-hoc explanation techniques (see Fig. 1.2), such as visualization methods, can offer insights into a model's outputs and discover important input features [20]–[22], it is crucial for these explanation algorithms to be robust if we intend to rely on them for holding deep neural networks accountable. Ghorbani *et al.* [23] conducted a study showcasing the feasibility of introducing imperceptible adversarial perturbations to input images. Surprisingly, this resulted in identical predicted

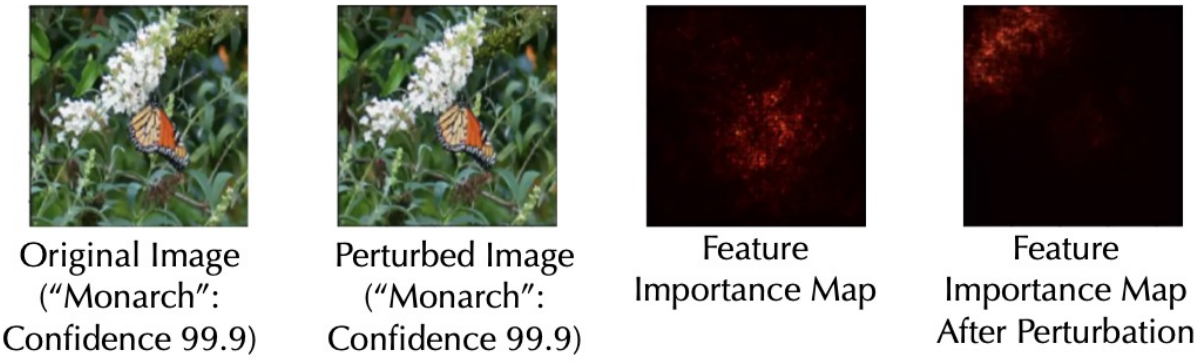


Figure 1.3: Post-hoc explanation algorithms are susceptible to adversarial attacks. Here, an image of a "Monarch" is perturbed with imperceptible noise such that the model still predicts the image as a "Monarch", however, the explanation (feature importance map) does not highlight the semantic pixels corresponding to "Monarch".

labels but vastly different explanations (see Fig. 1.3). Moreover, much of the existing research has predominantly focused on developing universal explanation techniques that are expected to perform well across all instances. However, as highlighted in this thesis, the pursuit of such universal solutions may resemble a quest for the mythical El Dorado. It's evident that no single technique can adapt effectively to all data and user contexts. According to Singh *et al.* [24], if we use post-hoc methods to generate explanations, there is a dependency on correlated contextual features and background information, potentially introducing biases into an explanation. For instance, images categorized as "microwave" may frequently appear alongside "refrigerator" or "sink" in the background, which inadvertently becomes a contextual cue used by the model to identify a "microwave". Explanation heatmaps obtained using Grad-CAM highlights this correlated contextual bias. Fig. 1.4 provides an example of such a heatmap. If we plan to deploy deep models for safety-critical applications, it is crucial for the model to have accurate explanations for its decisions.

Furthermore, large-scale models such as foundation models have become the cornerstone of many AI applications, powering various tasks such as text summarization, question answering, and visual analysis. They have also been fine-tuned for specific domains or applications, further enhancing their performance and adaptability. If the foundation model's outputs are used by humans to make decisions or take actions, explainability must be considered to enhance trust in the system and facilitate its acceptance. However, how to make these well-trained models explainable remains underexplored. These models are typically so large that scholars typically lack the necessary computational resources or cannot afford the associated costs to study or understand them. For instance, the Generative Pre-trained Transformer 3 (GPT-3) language model is estimated to require an expenditure of 4.6 million dollars and consume hundreds of MWh of energy for a single training session on a cloud GPU [15]. Although Google's

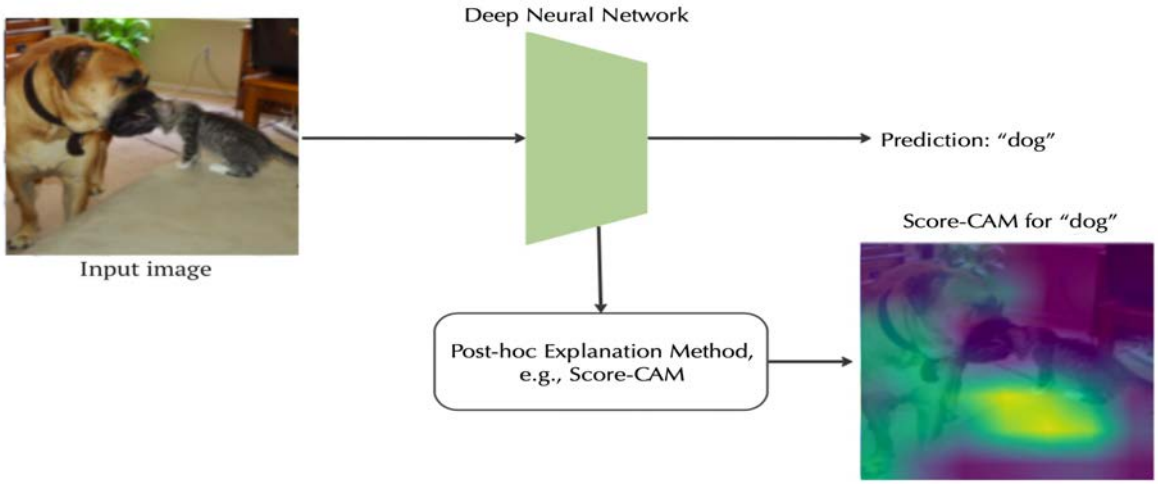


Figure 1.4: An explanation algorithm such as Score-CAM [26] demonstrates that the model's accurate classification of an image into the "dog" category depends on the existence of background objects such as a "table".

Bidirectional Encoder Representations from Transformers (BERT) [25] was made open source, its black-box nature makes it inaccessible for many researchers. Due to their size and lack of explainability, foundation models have limitations and are unsuitable for certain applications where trust is crucial. Consequently, the question of how to effectively explain these models and reduce their size remains open, as it contributes to the democratization of science and aligns with the principles of data processing and algorithmic transparency.

Given the overwhelming popularity of text prompting in numerous Generative AI scenarios [27], how to effectively support such inputs with explanation and guidance presents a significant challenge. End-users often lack knowledge about the quality of the text prompt they use to obtain the desired results from Generative AI. Similar to how adversarial examples lead to unexpected errors in Discriminative AI, Generative AI also faces a comparable issue when modifications made by users in the text prompt fail to produce the anticipated changes in the generated content or, in some cases, result in undesirable alterations [28]. The obscure and unexplainable relationship between the input text prompt and the generated output often compels end-users to engage in unguided trials, consuming their time and wasting computational resources. Consequently, users may find it difficult to establish trust and acceptance towards the generated content. Therefore, it is crucial to conduct further studies to comprehend how humans utilize prompts to interact with Generative AI.

Research gaps: XAI has gained significant attention due to the increasing complexity of AI models. Various methodologies have been developed to make AI systems interpretable, including model-specific approaches such as feature importance analysis [29], model-agnostic techniques like SHAP [30], and post-hoc explanation methods

such as LIME (Local Interpretable Model-agnostic Explanations [31]). Despite these advancements, several research gaps persist within the field of explainable AI:

- There is a critical trade-off between interpretability and performance, as some explainability techniques may increase model complexity or reduce accuracy.
- As neural networks grow in size and complexity, there is a notable gap in developing computationally efficient explainable techniques that can preserve the essential features and layers critical to the model's decision-making process, allowing for optimal model compression without sacrificing accuracy.
- The field lacks robust methodologies for assessing and improving the quality of samples generated by generative models. Tailored explainable methods are needed to provide insights into the generation process and sample quality, enabling systematic evaluation and enhancement of generative model outputs.

1.3 Research Questions

According to the research gaps outlined in the preceding section, this thesis aims to study the following research questions:

- How can we improve the transparency and explainability of deep neural networks without compromising their performance?
- How can XAI methods be leveraged to optimize the compression of large-scale models while maintaining or enhancing model performance?
- How can XAI techniques be developed and integrated into the training and evaluation of generative models to enhance the quality, reliability, and interpretability of generated samples?

1.4 Contributions

We have proposed novel approaches to address the aforementioned research questions. The advancement of knowledge and related publications are summarized as follows:

- We propose an explainable framework that identifies regions of interest influencing class categories, thereby enhancing the interpretability of neural networks. It consists of a predictor and an explainable tool, that is able to provide accurate visualization maps and prediction basis. Specifically, the predictor is designed

by applying attention mechanisms to multi-scale features to learn and discover class discriminative latent representations. Meanwhile, to explain our predictor, we propose a novel explainable tool that includes a high-resolution visualization method and a prediction-basis module. The former effectively integrates the feature maps of intermediate layers as well as the last convolutional layer, which surpasses state-of-the-art visualization approaches in producing high-resolution representations with more accurate localization of discriminative areas. The prediction-basis module provides prediction basis evidence via retrieved samples that are accessible to end-users.

Related publication: **L. Yu**, W. Xiang, J. Fang, Y. P. Chen, and R. Zhu, "A novel explainable neural network for Alzheimer's disease diagnosis," *Pattern Recognition*, vol. 131, pp. 1-12, Jun. 2022 (IF = 8.0).

- We propose a vision transformer dubbed the eXplainable Vision Transformer (eX-ViT), a transformer model with enhanced explainability by jointly discovering robust interpretable features and performing the prediction accurately. Specifically, eX-ViT is composed of the Explainable Multi-Head Attention (E-MHA) module, and the Attribute-guided Explainer (AttE) module. The E-MHA tailors explainable attention weights that are able to learn semantically interpretable representations from tokens in terms of model decisions with noise robustness. Meanwhile, AttE is able to encode discriminative attribute features for the target object through diverse attribute discovery, which constitutes faithful evidence for the model predictions.

  Related publication: **L. Yu**, W. Xiang, J. Fang, Y. P. Chen, and L. Chi, "eX-ViT: A Novel eXplainable Vision Transformer for Weakly Supervised Semantic Segmentation," *Pattern Recognition*, vol. 142, pp. 1-13, Oct. 2023 (IF = 8.0).

- We propose an explainable pruning framework dubbed X-Pruner, which is designed by integrating the explainability into the pruning process for a well-trained model. Specifically, to measure each prunable unit's contribution to predicting each target class, a novel explainability-aware mask is proposed and learned in an end-to-end manner. Then, to preserve the most informative units and learn the layer-wise pruning rate, we adaptively search the layer-wise threshold that differentiates between unpruned and pruned units based on their explainability-aware mask values.

  Related paper: **L. Yu**, and W. Xiang, "X-Pruner: eXplainable Pruning for Vision Transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Vancouver, Canada, Jun. 2023, pp. 24355-24363.

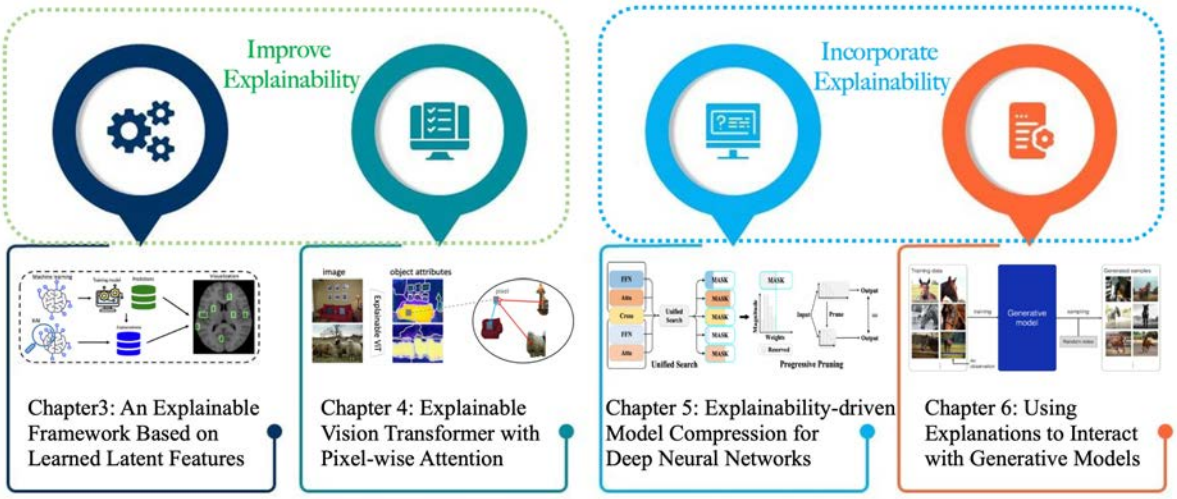


Figure 1.5: The overall structure of this thesis.

- We propose the Edit-DiffNeRF framework, that leverages explainability to allow users to interact with the generative process using text prompts. Instead of training the entire diffusion for each scene, our method focuses on editing the latent semantic space in frozen pretrained generative models by the delta module. This fundamental change to the standard framework enables us to make fine-grained modifications to the rendered views and effectively consolidate these instructions in a 3D scene via NeRF training. As a result, we are able to produce an edited 3D scene that faithfully aligns to input text instructions.

  Related paper: **L. Yu**, W. Xiang, and K. Han, "Edit-DiffNeRF: Editing 3D Neural Radiance Fields using 2D Diffusion Model," https://arxiv.org/abs/2306.09551.

1.5 Thesis Outline

This thesis will be divided into five parts. As depicted in Fig. 1.5, Chapters 3 and 4 will be focused on improving explainability, while Chapters 5 and 6 will concentrate on incorporating explainability. Specifically, Chapter 3 will cover the works focused on addressing model explainability, and Chapter 4 will focus on addressing model explainability for weakly supervised segmentation tasks. Chapters 5 and 6 will focus on enabling explainability. Chapter 7 will conclude this thesis.

Chapter 2 provides a comprehensive overview of explainable AI, including its definition, guiding principles, methodology, design considerations, and evaluation techniques.

Chapter 3 analyzes the issue of blurry visual explanations and presents a novel approach for enhancing the interpretability of neural networks through the aggregation of multi-scale intermediate features. A quantitative analysis is given to show that fused features with multi-scale information have a positive effect on generated explanation. We then formulate the proposed Dual Attention Module (DAM) to learn and discover class discriminative latent representations. This chapter also introduces a high-resolution visualization method that produces high-resolution visual explanations for the precise localization of target areas. Experimental results are presented in this chapter to show that the proposed model is able to achieve state-of-the-art performance in terms of accuracy.

Chapter 4 describes the novel explainable vision transformer (eX-ViT) with enhanced explainability and high performance. Starting with standard black-box models, this chapter discusses the problem of existing models, and how the proposed eX-ViT solves that problem. To enable richer representations of interpretable attention maps that align with informative input patterns, we present the Attribute-guided Explainer (AttE) to decompose the feature representation into a set of learnable attribute features for the target object, capable of capturing diverse and discriminative object features. In addition, a novel attribute-guided loss to promote the learning process inside AttE in a self-supervised manner is introduced. More precisely, this loss implicitly adds the regularization to force the representations to focus on various attributes of each target class through the attribute discriminability mechanism and attribute diversity mechanism. Simulation results are presented to illustrate that the proposed model achieves comparable performance to supervised baselines, while surpassing the accuracy and interpretability of state-of-the-art black-box methods using only image-level labels.

Chapter 5 presents a model pruning technique that incorporates explainability into the pruning process of a well-trained model. This chapter first introduces an explainability-aware mask for each prunable unit in a model, with the goal of quantifying its contribution to predicting each class. Specifically, the proposed mask is fully differentiable and can be learned in an end-to-end manner. We demonstrate many benefits of the proposed mask, including more accurate pruning and fewer computational costs compared with existing black-box pruning methods. Then, this chapter describes how to learn the layer-wise pruning thresholds that differentiate the important and less-important units via a differentiable pruning operation. Finally, experimental results are presented for various models to showcase the effectiveness of the proposed method.

Chapter 6 introduces a framework that leverages explainability to facilitate 3D editing through text prompts with high fidelity and multi-view consistency. This chapter centers on altering the latent semantic space within frozen pretrained diffusion models via the delta module. This fundamental modification to the standard diffusion

framework enables precise adjustments to rendered views and their consolidation in a 3D scene via NeRF training. Consequently, we achieve the production of an edited 3D scene that accurately aligns with input text instructions. Additionally, to ensure semantic consistency across different viewpoints, we propose a novel multi-view semantic consistency loss, which extracts a latent semantic embedding from the input view as a prior and endeavors to reconstruct it in various views. The effectiveness of our proposed framework in editing real-world 3D scenes across a variety of text prompts is demonstrated through experimental results on diverse real-world datasets.

Chapter 7 concludes this thesis and discusses possible future work in XAI field from different points of view.

Chapter 2

Background

2.1 Definitions of Explainability

While existing concepts of explainability primarily focus on human comprehension, they vary regarding which aspect of the model is to be explained: its internal workings, operations, data mapping, or representation [32]. Consequently, a formally agreed-upon definition remains elusive. An explanation comprises features from an interpretable domain that establish a connection between a data instance and the output of a model [33]. Explanations can vary in truthfulness, accuracy, and success, sometimes being deceptive or inaccurate. Therefore, multiple explanations are often utilized to achieve a comprehensive interpretation of a model. Miller suggests that explainability research should draw upon insights from philosophy, psychology, and cognitive science to understand how explanations are defined, generated, selected, evaluated, and presented [34]. Arrieta [35] proposed that the characterization of XAI should highlight how the clarity of a model's explanation relies explicitly on the audience: an AI system is considered explainable when it furnishes specific details or rationales tailored to render its operations clear or readily comprehensible to a given audience. In this thesis, we follow the existing literature on explainability by adopting a human-centered approach to elucidate why data scientists require explainability, how they utilize it, and how explainable methods can aid in designing interfaces to elucidate models.

2.2 Guidelines of XAI Systems

The recent inclusion of the "right to explanation" in the GDPR has sparked discussions regarding its practical implications and the potential impact on industry and research areas [36]. Although the updated GDPR mandates explanations only in specific contexts, AI and policy experts anticipate explanations to play a crucial role in future AI

system regulations [32]. To address the vagueness of the GDPR, researchers have proposed a framework to translate its language into actionable guidelines. These guidelines include (1) identifying the factors influencing a decision, (2) understanding how variations in these factors affect the decision, and (3) comparing similar inputs with different predictions [1]. However, according to this framework, an AI system only needs to fulfill one of these guidelines to be deemed explainable. Additionally, alternative post-hoc techniques for explaining decisions have been suggested, including the use of counterfactuals (i.e., "What if" questions), textual explanations, visualizations, local explanations, and representative data examples [37].

2.3 Method, Design, and Evaluation of XAI Systems

The ability to generate explanations relies on the model's capability to enable or integrate interpretations. Existing literature draws a distinct line between models that are inherently explainable and those that are elucidated externally through explanation techniques.

2.3.1 Post-hoc Techniques in XAI

When machine learning models fall short of transparency standards, an alternative approach must be developed and implemented to elucidate their decisions. This is the objective of post-hoc explainability techniques, specifically tailored for models that lack interpretability by default. In the literature, two categories of post-hoc explainability methods are identified: model-agnostic and model-specific approaches [35], [38]. Model-agnostic explanation methods enhance the versatility of the explanation technique by not being tied to a particular kind of model, thereby increasing its generalizability.

Model-agnostic explanations: These explanation methods serve to offer a general estimation of the behavior of a black-box model, indicating its typical behavior for a given dataset. Techniques like LIME [39], SHAP [40], Anchors [41], counterfactual explanations [42], and partial dependence plots [43] offer valuable insights into the decision-making processes of machine learning models without necessitating knowledge of their internal architectures. For instance, LIME approximates a model's local behavior using interpretable models, while SHAP values allocate feature contributions based on cooperative game theory principles. Anchors provide interpretable rules that reflect the model's predictions, and ICE plots visualize changes in instance predictions with varying feature values. Counterfactual explanations and partial dependence plots

offer supplementary perspectives on model behavior. These model-agnostic methods empower users to trust and comprehend predictions made by black-box models, thereby facilitating informed decision-making and model refinement endeavors [44], [45].

Model-specific explanations: There are also explanation methods specifically tailored for particular machine learning or deep learning models. Methods such as Grad-CAM [19] highlight crucial regions within input images corresponding to specific classes, offering visual insights into the decision-making process of the model. Layer-wise Relevance Propagation (LRP) [20] dissects predictions by assigning relevance scores to individual neurons or input features, illuminating the most influential aspects of the input. Integrated Gradients [46] assesses the contribution of each feature to the prediction by integrating gradients along a direct path from a baseline input to the actual input. DeepLIFT [47] attributes disparities in neuron activations to input features, elucidating the impact of input variations on predictions. SmoothGrad [48] enhances gradient-based attribution methods by mitigating noise in gradients, whereas saliency maps identify significant regions in input data that steer model predictions. These methodologies deepen comprehension of deep learning model behavior, facilitating model interpretation and debugging endeavors.

2.3.2 Explainable Models

These models provide users with the ability to analyze and grasp the mathematical transformation of inputs into outputs, allowing them to connect input attributes or features to their corresponding output. Users can gather and comprehend technical details regarding this mapping to a certain extent. For instance, Support Vector Machines (SVMs) and other linear classifiers offer explainability as they delineate data classes based on their positioning relative to decision boundaries [49]. However, as models grow in complexity, there arises a need for explanations alongside model outputs. Lipton [50] describes the transparency among these models across three levels: the ability to simulate the entire model, the decomposability of individual components, and algorithmic transparency.

Simulatability refers to the ability of a model to allow a user to fully understand its structure and operation. For a model to be considered entirely understandable, a human should be able to take the input data along with the model's parameters and analyze every computation necessary to generate a prediction within a reasonable timeframe. The extent of simulatability is influenced by the overall size of the model and the computational complexity required for inference. For example, in decision trees [51], the model's size may increase much faster than the time needed for inference.

Due to the limited capacity of human computation, this discrepancy may span several orders of magnitude. Consequently, high-dimensional models, extensive rule lists, and deep decision trees are not easily explainable and may exhibit lower explainability compared to more compact neural networks.

Decomposability refers to the extent to which a model can be broken down into its constituent parts (such as inputs and parameters), thereby facilitating a more intuitive approach to explainability [52]. For example, in a decision tree, each node could be associated with representations describing similar nodes sharing the same features. Similarly, the parameters of a linear model might reflect the relationship between features and the output. This form of transparency necessitates that inputs are individually explainable; features that are highly engineered or anonymous may not lead themselves well to decomposability.

Transparency refers to the level of confidence regarding an algorithm's ability to function sensibly in unforeseen scenarios [53]. For instance, linear models are deemed transparent since we can comprehend the shape of the error surface and make deductions based on it. This level of understanding provides a degree of confidence that the model will behave as anticipated in unfamiliar situations. Conversely, current deep learning techniques limit this aspect of algorithmic transparency as they cannot be completely observed. The main challenge for algorithmic transparency in such models is the necessity for mathematical analysis and features to be observed.

2.3.3 Evaluating XAI Methods

The introduction of diverse explanation methods has prompted researchers to devise diverse evaluation metrics for assessing a model's effectiveness in specific aspects of explainability. A comprehensive examination of these studies unveiled two primary approaches to evaluating XAI methods: functionally-grounded evaluations and human-centered evaluations [54], [55]. Furthermore, we correlate different attributes and characteristics of explanations to be evaluated within each group, as depicted in Fig. 2.1.

Functionally-grounded Evaluations: This form of evaluation utilizes specific characteristics of explainability as indicators of explanation quality. Objective experiments offer certain advantages, particularly as conducting human-subject studies often requires significant time, funding, and approvals, which may exceed the resources available to a machine learning researcher. Functionally-grounded evaluations are particularly useful when a set of baseline models has already been validated, potentially through human experiments. However, they can also be relevant during the early stages of method development or when human subject studies are impractical due

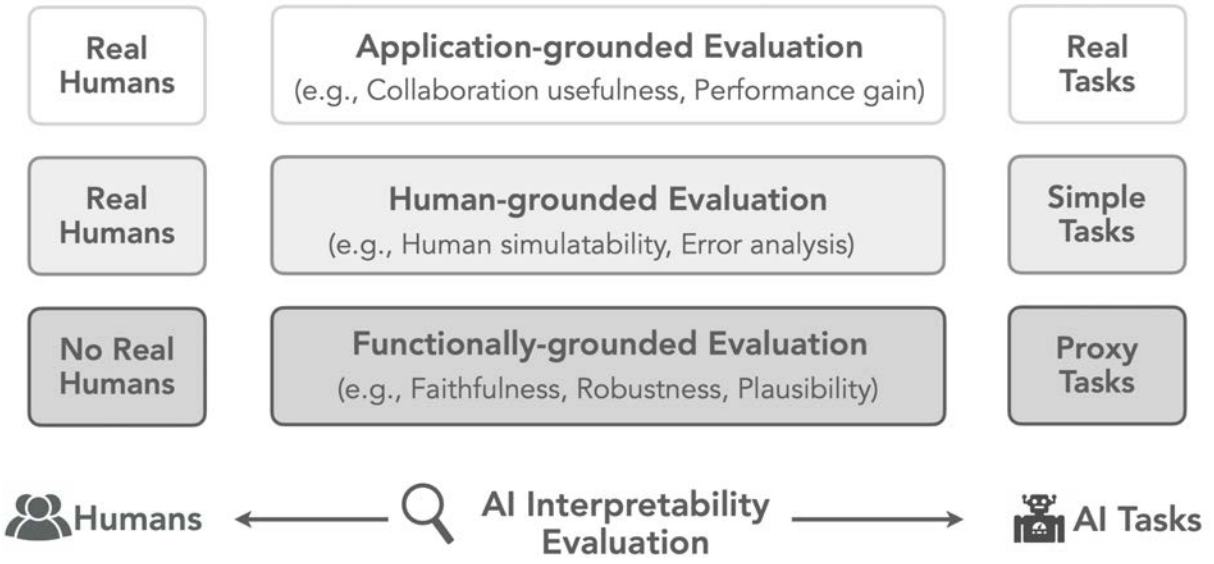


Figure 2.1: Useful XAI for humans in practice [56].

to ethical considerations [57]. In such instances, the properties of explanation methods outlined below can be employed to compare various approaches and assess their strengths and weaknesses.

The *faithfulness* metric assesses the precision with which the explanation mirrors the behavior of the model [58]. A prevalent approach to evaluating faithfulness, inspired by image interpretation practices, involves computing the smallest sufficient region (i.e., the smallest region containing an instance that supports a correct prediction) or the smallest destroying region (i.e., the smallest region whose removal leads to an incorrect prediction). High faithfulness is consistently desirable. When the model exhibits high accuracy and the explanation maintains high faithfulness, the explanation consequently achieves high accuracy as well. Conversely, low explanation accuracy is anticipated when the accuracy of the machine learning model is also low [59].

Robustness denotes the dependability and coherence of explanations offered by a method across diverse scenarios and circumstances [60]. A model is deemed robustly explainable if its explanations retain significance and consistency despite alterations in input data, model parameters, or environmental variables [61]. This robust interpretability ensures that the explanations provided by the model genuinely represent the underlying decision-making process and are not unduly affected by noise or disturbances.

Furthermore, there are additional metrics that may be pertinent to consider including consistency [62], compactness [63], and algorithmic complexity [61]. Algorithmic complexity is related to computational demands, particularly when computation duration presents a bottleneck in generating explanations. Consistency measures the degree of variation in explanations between two distinct models trained on the same task

and producing similar output predictions.

Human-Grounded Evaluations Human-grounded evaluations involve human input or judgment to assess machine learning models or algorithms [64]. Participants are often asked to provide feedback, ratings, or assessments on aspects like interpretability, usability, or task effectiveness. These evaluations may include user studies and surveys. They complement objective evaluations by offering insights into user perceptions and interactions with machine learning models in real-world scenarios. This understanding aids researchers in identifying areas for improvement from a user-centered perspective and in gauging the practical implications of model behavior [65]. Here we explore several properties that contribute to the quality of explanations.

Clarity refers to the extent to which the resulting explanation is explicit [66]. This attribute holds particular significance in safety-critical applications, where ambiguity must be minimized. *Justifiability* indicates the extent to which an expert can assess the explanations to verify whether the model aligns with domain knowledge [67]. *Informativeness* relates to the capacity of an explanation method to furnish relevant information to an end-user [68].

2.4 Boosting Model Performance through XAI

The integration of explainable AI (XAI) techniques has shown promise not only in enhancing transparency but also in improving the generalizability, efficiency, and fairness of AI models. For example, in domain generalization, end-to-end deep models often exploit biases unique to their training dataset, which leads to poor generalization. In fact, increasing the explainability of a deep classifier can improve its generalization, especially to novel domains. different works [69], [70] have proposed ways to learn more general representations by utilizing explainability as a means for bridging the visual-semantic gap between different domains, models can be made more robust across different datasets and environments. Additionally, XAI contributes to data-efficient training by identifying the most influential features or data points in a model's predictions [71]. This insight can guide the prioritization of high-impact data collection, reduce redundancy within training datasets, and concentrate computational resources on the most informative samples [72], [73]. As a result, models can achieve high performance with less data, thereby lowering the costs and time associated with extensive data collection and processing. Furthermore, biases in AI models, often originating from imbalances or undesirable patterns in training data, can be identified through explainability techniques. By elucidating how different features influence predictions across demographic groups, XAI facilitates the identification and mitigation of biases through strategies such as re-weighting, data augmentation, or adversarial training, leading to

fairer and more equitable model outcomes [74], [75]. In addition, explainability methods, such as saliency maps, can be used to identify and retain critical components of a model, thereby facilitating more effective and targeted pruning strategies [19], [26]. By highlighting redundancies and low-impact parameters, XAI enables adaptive pruning processes that maintain model accuracy while significantly reducing computational costs.

2.5 Applications of XAI Systems

Explanation algorithms could significantly advance research in computer vision and machine intelligence, powering many innovative applications. An explanation for the decision process is very helpful in facilitating numerous groundbreaking applications. The elucidation of decision processes proves particularly beneficial in the realm of computer vision, a field with diverse applications like object detection in autonomous vehicles, contributing to collision avoidance and traffic reduction. By integrating XAI techniques like SHAP [76], LIME [39], and gradient-weighted class activation mapping (Grad-CAM) [19], these systems can elucidate various traffic situations to drivers. This not only enhances user trust in the technology but also assists drivers in making critical decisions in complex traffic scenarios. Additionally, XAI algorithms are quite useful in monitoring suspected criminals, thereby mitigating criminal activities, as well as in structural monitoring and disaster management. In essence, the adoption of XAI algorithms addresses key challenges in computer vision-based applications by providing transparency, interpretability, and justification for classification results. This, in turn, opens up new possibilities for the responsible and effective deployment of AI technologies in diverse fields.

One of the important applications of the XAI is Google's What-If tool [77], which is an open-source visualization tool that allows users to analyze the behavior of various machine learning models. It provides an interactive interface to explore and understand model predictions, investigate the impact of input changes, and assess the impact of changes on outcomes. To further broaden engagement in the community, the What-If Toolkit offers insights into model behavior, data sources, and potential biases, enhancing transparency and accountability in AI systems, and facilitating communication about models' ethical and performance aspects.

Captum [78] is another interpretable product of the XAI. Captum is a powerful, flexible, and user-friendly model interpretability library for PyTorch. It makes state-of-the-art algorithms for explainability readily accessible to the entire PyTorch community, so researchers and developers can better understand which features, neurons,

and layers are contributing to a model's predictions. Captum supports model interpretability across modalities such as vision and text, and its extensible design allows researchers to add new algorithms. Captum also allows researchers to quickly benchmark their work against other existing algorithms available in the library.

As we have explored the foundational concepts and methodologies in the broader field of explainable AI, it is essential to examine how these principles are applied within specific application domains. One such domain where explainability plays a crucial role is medical imaging. In recent years, the integration of AI into medical imaging has revolutionized diagnostic processes, offering unprecedented accuracy and efficiency. However, the complexity and opacity of AI models present unique challenges in ensuring that these systems are both reliable and interpretable by clinicians. Chapter 3 will delve into the application of explainable AI techniques in medical imaging, discussing how XAI methods can enhance diagnostic accuracy and improve model transparency.

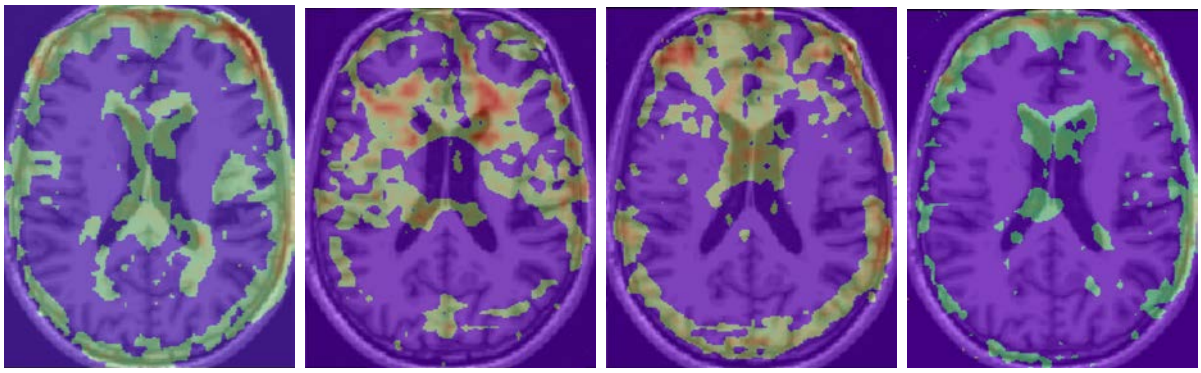
Chapter 3

An Explainable Framework Based on Learned Latent Features

In this chapter, we construct an explainable framework for medical image analysis, especially the Alzheimer's disease diganosis. We analyze the issue of blurry visual explanations and discover that the utilization of intermediate features in a model with multi-scale fusion will improve the quality of heatmaps. Based on this analysis, we propose the MAXNet that uses attention mechanisms to learn discriminative latent representations of brain volumes, and introduce the explainable tool to generate high-resolution visualization maps and prediction basis evidence to explain the predictor's decisions. We demonstrate that the proposed framework is effective in dealing with inaccurate visual explanations, and user-friendly due to its explainability.

3.1 Introduction

Alzheimer's disease (AD), the most common form of dementia, which could induce movement disorders and a series of subsequent syndromes, has affected over 50 million people universally and is growing rapidly [79]. Traditionally, the computer-aided detection of AD using machine learning methods develops feature descriptor and classification systems. However, the hand-crafted features suffer from subjectivity and cannot generalize well across instances. Thanks to extensive research on applications of deep learning (DL) such as CNNs, medical scientists have sought a new era of engagement with AI-based diagnosis of detecting AD at an early stage automatically [80]. With the advance of magnetic resonance technology, magnetic resonance imaging (MRI) data are often provided to observe the development of brain tissue morphology related to AD [81]. Plenty of DL architectures have been proposed to classify AD using brain MRIs and gained satisfactory performance [82]. However, despite their significant achievements, the predictions of existing models are not faithful with the expected reasoning. That is, they do not provide any explicitly visual or other forms of



(a) (b) (c) (d)

Figure 3.1: Visualization results of state-of-the-art methods for a AD patient: (a) CAMERAS [22]; (b) Grad-CAM [19]; (c) Grad-CAM++ [38]; (d) Score-CAM [26]. All of them provide blurry visual explanations or recognition of irrelevant noise.

explainable information associated with the final output. This becomes a major hurdle to apply these techniques on a mass scale due to the lack of humans' trust.

XAI is an emerging sub-field of AI pursuing to capture the properties that have influence over the decision of a model [35]. To fully uncover the CNNs, several works have proposed to build interpretable CNN models. Zhang et al. [26] proposed a general approach to train interpretable convolutional filters in CNN models, wherein each filter represents a certain part of the object. Lee et al. [83] designed to make final decisions based on the regional abnormality representation by use of complex nonlinear relationships among voxels. However, most existing methods only provide blurry heatmaps or recognition of atrophy with irrelevant noise (Fig. 3.1), this can be attributed to the fact that the leveraged last convolutional layer only extracts global features and misses the small attributes and discrepancies. Therefore, they are not able to provide enough details to precisely recognize crucial areas, and fail to localize small differences in medical imaging diagnosis.

Different from existing XAI works in the literature, we aim to develop an explainable framework for automated diagnosis of AD capable of providing accurate predictions with fine-grained heatmaps and prediction reasoning. We first build an explainable network dubbed MAXNet with two novel modules, Dual Attention Module (DAM) and Multi-resolution Fusion Module (MFM), to capture and fuse multi-resolution features. Intuitively, we hope the MAXNet can learn representations containing all the necessary voxel information for the correct predictions. Therefore, we design the cluster and contrastive loss functions to make the model learn and extract semantically informative latent features of the target label. Second, to provide high-resolution heatmaps and prediction reasoning, we propose an explainable tool that consists of a novel visualization method termed High-resolution Activation Mapping

(HAM), and a Prediction-basis Creation and Retrieval (PCR) module. The former is for yielding fine-grained heatmaps for disease areas, while the latter creates a prediction reference-set during training, in which subjects similar to a query volume are retrieved during testing, to enhance the explainability of predictions.

3.2 Related Work

Traditional methods for AD diagnosis There has been much interest in feature selection techniques to assist the diagnosis of AD using individual brain MRIs. A group of studies are proposed to utilize hand-crafted features extracted from MRI data in combination with different models. Zhang *et al.* [84] proposed a multi-task feature selection (MTFS) method that selects subsets of features from each modality. Based on this, Liu *et al.* [85] developed an inter-modality feature selection method (IMTFS) to process the complementary inter-modality features. Zhu *et al.* [86] adopted manifold regularized multi-task learning for AD diagnosis. Moreover, Shi *et al.* [79] first developed a nonlinear feature engineering module, then used the support vector machine (SVM) to identify AD patients. Cao *et al.* [87] explicitly extracted subset features and Region-of-Interests (ROIs), then combined these features in a multi-task learning framework for AD diagnosis. Gerardin *et al.* [88] modeled the shape of hippocampus regions via spherical harmonics and developed a classification procedure to automatically discriminate between patients. Stefan *et al.* [89] employed various measurements to obtain expressive MRI biomarkers and fed them into a linear discriminant analysis system. However, these traditional computer-aided methods learn hand-crafted representations can be prone to subjectivity, and are difficult to be optimized.

Deep learning methods for AD diagnosis Recently, deep learning techniques have made great progress on AD diagnosis with the benefit of automatic abstraction of multi-level latent features. Chen et al. [90] jointly used iterative spare and DL methods to learn representations of critical cortical regions that are used to diagnose AD. Su *et al.* [91] introduced domain adaptation to utilize feature distributions of brain images across multiple sites for binary classification. Pan *et al.* [92] devised a joint deep learning architecture to model the disease-image specificity as well as the disease diagnosis using incomplete MRI and fluorodeoxyglucose positron emission tomography (PET) images. Basaia *et al.* [82] built a 3D CNN for MRI data to distinguish among AD, c-MCI and s-MCI without any prior feature design. Lei *et al.* [93] introduced a convolutional network based on longitudinal multiple time points data for identifying AD subjects. Lian *et al.* [94] proposed a hierarchical fully convolutional network that automatically learns multi-scale feature representations in the whole brain structural magnetic resonance imaging (sMRI) data for AD diagnosis. Kröll *et al.* [95]employed

various residual structures to facilitate training and obtain information from previous layers. Gopinath *et al.* [96] proposed a new graph convolutional network for processing surface-valued data to output subject-based classification and regression. Despite the promising results of these models, almost all the prior approaches are designed with complex modules that are difficult to interpret.

Explainability A number of papers have been proposed to visualize a model's predictions by highlighting important regions that are believed to be intuitive to end-users. If we consider an image classification task as an example, a "good" visual explanation based on the model should be able to be (a) class-discriminative (i.e., localize the category in the image) and (b) high-resolution. Zhou *et al.* [21] introduced a technique called Class Activation Mapping (CAM) for identifying informative areas by a certain kind of classification CNNs that do not have fully-connected layers. Substantially, it utilized the last convolutional layer before the global pooling layer and combined weighted activation maps to produce explainable heatmaps, it turned out to be highly class-discriminative, but with quite blurry outputs as an undesirable attribute. Beyond that, Grad-CAM [19] generalized CAM to a relatively large set of CNN models without requiring a specific architecture, by backpropagating the gradient of a target class with respect to the pixel intensities. Jalwana *et al.* [22] proposed a mechanism to generate high-resolution heatmaps with improved activation map upsampling that corresponds to a model's logic. However, gradients for a deep learning model can be noisy and also easily to get vanished in sigmoid function or an activation function like ReLU. So Wang *et al.* [26] acquired each weight regarding individual activation map through feeding it into the network, and the heatmaps are yielded by the association between corresponding weights and maps. Although these algorithms achieved remarkable level of improvements, they either did not combine the advantage of both sides (class-discriminative and high-resolution), or get stuck into one of them.

In the domain of medical image analysis, Hannun *et al.* [97] utilized the electrocardiogram tool to interpretate the clinical ECG process in an end-to-end manner. Afshar *et al.* [98] took advantage of capsule networks to model nodule features and provide potential interpretability of the model. Malhotra *et al.* [49] proposed a multi-task model to predict COVID-19 in chest X-ray images and segmented the lung regions with COVID-19 symptoms. Xie *et al.* [99] conducted three iterations of design activities to formulate a system, which enables clinicians to explore and understand AI-based chest X-ray analysis. Chittajallu *et al.* [100] presented a human-in-the-loop XAI system for content-based image retrieval of video frames similar to a query image from invasive surgery videos for surgical education. Jin *et al.* [101] introduced an attention guided network to localize image biomarkers and provide intuitive explanations. Hu *et al.* [102] developed an interpretable multimodal fusion model by utilizing the Grad-CAM.

Nevertheless, existing DL methods do not provide high-resolution heatmaps and thus can not give reliable explanations. In our proposed method HAM, we aim to produce visual explanations with fine-scale information as well as being class-discriminative.

3.3 Proposed Methodology

In this section, we formulate the problem under consideration and introduce our proposed MAXNet, HAM and PCR.

3.3.1 Problem Formulation

Firstly, given labeled training data $\{\boldsymbol{m}_i, y_i\}_{i=1}^{M}$ containing M samples wherein $y_i \in \{0, 1\}$ is a binary class label referring to the presence/absence of the AD and $\boldsymbol{m}_i \in R^3$ is an MRI volume, the proposed MAXNet aims to predict the corresponding diagnosis label $\hat{y}_i$ given input $\boldsymbol{m}_i$.

On the other hand, the fine-grained visualization task is to provide heatmap $\boldsymbol{A}_{\text{HAM}}$ by integrating the activation maps in intermediate layers as well as the last convolutional layer

$$\boldsymbol{A}_{\text{HAM}} = \sum_n U\big(ReLU\big(N\big(\tfrac{1}{Z_n}\sum \tfrac{\partial s_i}{\partial \boldsymbol{F}_n}\big)\boldsymbol{F}_n\big)\big), \tag{3.1}$$

where Z_n is the number of filters in the n-th layer, $\boldsymbol{s}_i$ is the predicted score, and $\boldsymbol{F}_n$ is the n-th activation map. $N(\cdot)$ and $U(\cdot)$ represent the normalization and up-sampling functions, respectively. Moreover, the task of evidence presentation is to firstly create a reference set $\left\{\boldsymbol{R}^i_{\text{ref}}\right\}$ for each label y_i from the training dataset $\{\boldsymbol{m}, m\}$, where $\boldsymbol{R}^i_{\text{ref}} \subseteqq \boldsymbol{m}$. Afterwards, we can retrieve samples $\{\boldsymbol{R}_c, y_c\}_{c=1}^{3}$ that have the most similar latent features compared to the input volume during the test phase.

3.3.2 Framework Overview

We propose an explainable framework for automated diagnosis of the AD from MRI volumes, which is capable of providing accurate classification results with fine-grained visualization maps and a prediction basis. The schematic of our framework including MAXNet, HAM, and PCR is in Fig. 3.2. We first craft the so-called Multi-scale Attention eXplainable Network (MAXNet), to address the aforementioned challenging issues and power the visual interpretability elaborately. Then we present a new high-resolution visualization approach, referred to as High-resolution Activation Mapping (HAM), which extracts salient features related to the AD (e.g., the atrophy of cerebral cortex and hippocampus.) to interpret model decisions. Furthermore, a reference set $\boldsymbol{R}_{\text{ref}}$ is created by the Prediction-basis Creation and Retrieval module during training

Table 3.1: List of symbols and their descriptions.

$\boldsymbol{m}_i$	The i-th MRI volume in the training dataset
y_i	Label of $\boldsymbol{m}_i$
$\hat{y}_i$	Predicted label of $\boldsymbol{m}_i$
$\boldsymbol{R}^i_{\text{ref}}$	Reference set for label y_i
$\boldsymbol{R}_c$	Reference sample where $c = 1, 2, 3$
$\boldsymbol{m}^{\text{T}}_k$	The k-th MRI volume in the testing dataset
$\hat{y}^{\text{T}}_k$	Predicted label of $\boldsymbol{m}^{\text{T}}_k$
$\boldsymbol{p}^k$	Latent features for the k-th MRI volume $\boldsymbol{m}^{\text{T}}_k$
$\boldsymbol{p}^c$	Latent features for the c-th reference sample $\boldsymbol{R}_c$

to extract and save relevant samples for certain labels, and is then used during testing to provide evidence of samples $\boldsymbol{R}_c$ with labels y_c.

3.4 MAXNet Architecture

There are several essential modules that constitute the MAXNet: 1) the staged feature extraction flow; 2) the Dual Attention Module (DAM); 3) the Multi-resolution Fusion Module (MFM); 4) the cluster and contrastive loss functions. In comparison to mainstream CNN models, MAXNet has multi-resolution fields as highly complementary to capture accurate localization of homogeneous areas. It is also able to learn latent representations that are close to each volume's label with the proposed cluster and contrastive loss functions.

3.4.1 Staged Feature Extraction

We start our network with a high-to-low convolutional stream, which would be devised as five stages in the MAXNet as shown in Fig. 3.2. Each stage is a convolutional block, which is sequentially made of a convolutional layer, a batch normalization (BN) layer, a rectified linear unit (ReLU), and a max-pooling layer. We make several adaptations to create our high-to-low stream. First, as in stages 1 and 2 which produce larger spatial outputs compared to their higher counterparts, the kernel size is set to be $3 \times 3 \times 3$ and the number of filters is set to 15 and 25 respectively to save computational resources. Upon these, since the DAM would be applied to stages 3 and 4, we increase the express capacity across these two blocks with the number of convolutional kernels

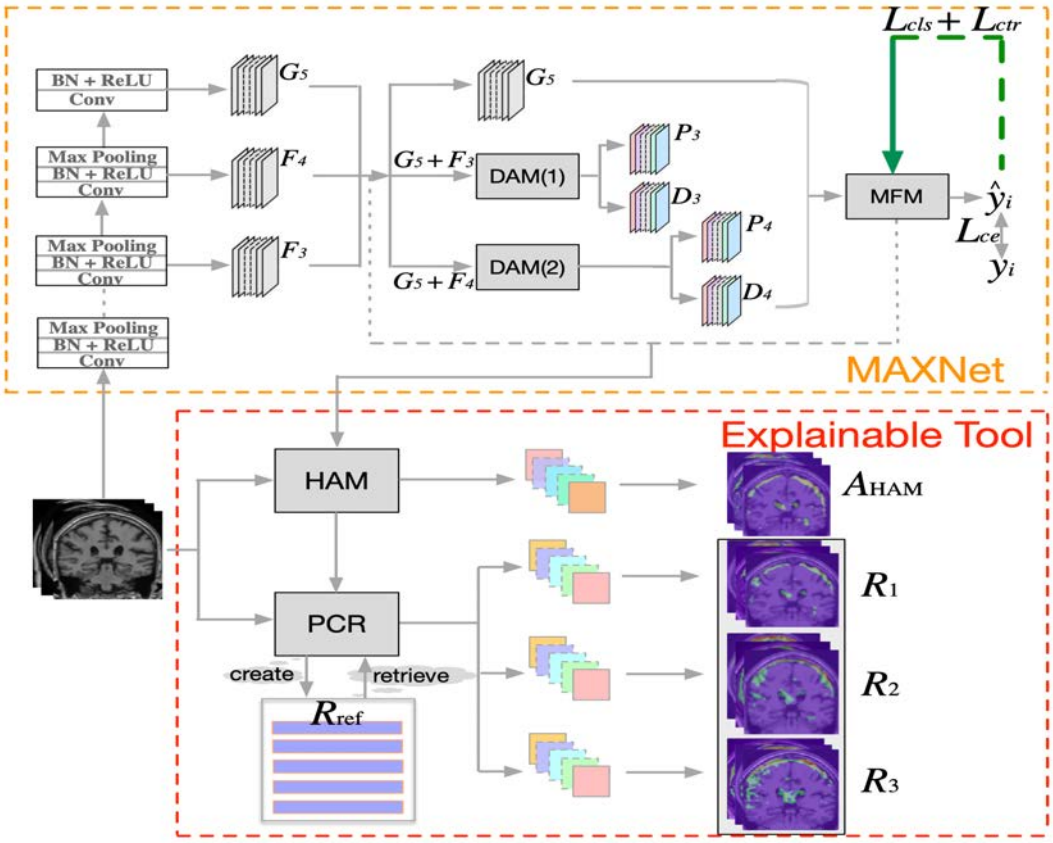


Figure 3.2: Schematic of the overall framework, which consists of the explainable model MAXNet and the explainable tool, i.e., HAM and the PCR module. In MAXNet, the high-to-low convolutional stream forms several stages (stages *1-5*). We define $\boldsymbol{F}_n$, $(n \in [1, 2, 3, 4, 5])$ as the intermediate activation response of the n-th stage before the max-pooling layer, and $\boldsymbol{G}_n$ as the final output of each stage n after max-pooling. $\boldsymbol{F}_3$ and $\boldsymbol{F}_4$ are leveraged to form the voxel-wise feature maps $\boldsymbol{P}_3$, $\boldsymbol{P}_4$, and the depth-wise feature maps $\boldsymbol{D}_3$ and $\boldsymbol{D}_4$ via the DAM respectively. Note that $\boldsymbol{G}_5$ from the last convolutional layer only extracts global features of the pathological abnormalities and misses the small subjects and discrepancies. Eventually, $\boldsymbol{P}_3$, $\boldsymbol{D}_3$, $\boldsymbol{P}_4$, $\boldsymbol{D}_4$, and $\boldsymbol{G}_5$ are fused via the MFM to produce the classification label $\hat{y}_i$. Subsequently, visual explanations $\boldsymbol{A}_{\text{HAM}}$ are obtained via HAM by multi-stage aggregation, and PCR is used to retrieve three reference samples $\boldsymbol{R}_1$, $\boldsymbol{R}_2$ and $\boldsymbol{R}_3$ most similar to the input volume, which are displayed as the evidence with ground-truth labels y_1, y_2, y_3.

to be the same as in stage 5, i.e., 50. We use convolutional layers with kernel size of $3 \times 3 \times 3$ for stages 3 and 4, and $1 \times 1 \times 1$ for stages 5.

3.4.2 Dual Attention Module (DAM)

In a classical classification model, which usually extracts features by looking at each sub-area equivalently, much information about local context clues could be excluded via upper layers. Thus, our intention is to devise a Dual Attention Module (DAM), as demonstrated in Fig. 3.3, to capture both voxel-wise and depth-wise dependencies

from both high and low resolution feature maps. Consequently, the multi-resolution relationships can be well represented for the model decisions.

Voxel-wise attention

For voxel-wise dependencies and differences between different stages, a voxel-wise attention module is applied to both current stage and the final stage as depicted in Fig. 3.3(a). Specifically, the encoding process for stage n ($n \in [3,4]$) involves three steps: Firstly, we map $\boldsymbol{F}_n$ and $\boldsymbol{G}_5$ onto a mutual embedding space:

$$\hat{\boldsymbol{F}}_n = W_f(\boldsymbol{F}_n), \quad \boldsymbol{G}_A = W_a(\boldsymbol{G}_5), \tag{3.2}$$

where $W_f(\cdot)$ contains one convolution layer as Conv(filter=25, kernel-size=1, strides=1), and $W_a(\cdot)$ is composed of one learnable convolution layer Conv(filter=25, kernel-size=1, strides=1) followed by one up-sampling layer. After projection, we obtain $\hat{\boldsymbol{F}}_n \in R^{D_n^{'} \times C_n \times H_n^{'} \times W_n^{'}}$ and $\boldsymbol{G}_A \in R^{D_n^{'} \times C_n \times H_n^{'} \times W_n^{'}}$. Secondly, an element-wise product is applied for $\boldsymbol{G}_A$ and $\hat{\boldsymbol{F}}_n$ to get the following interaction-aware attention matrix:

$$\boldsymbol{c}_{i,j} = \hat{\boldsymbol{F}}_n(i) \times \boldsymbol{G}_A(j), \tag{3.3}$$

where $\boldsymbol{c}_{i,j}$ represents the correlations of voxels <i, j> for all elements in the activation feature maps. Note that feature $\boldsymbol{G}_5$ should have coarser but semantically stronger feature responses. Thus, $\hat{\boldsymbol{F}}_n$ and $\boldsymbol{G}_A$ have the same resolution but different temporal contextual coverage. Subsequently, we normalize $\boldsymbol{c}_{i,j}$ by

$$\boldsymbol{r}_{i,j} = \frac{\boldsymbol{c}_{i,j} - \min(\boldsymbol{c}_{i,j})}{\sum_{i,j}[\boldsymbol{c}_{i,j} - \min(\boldsymbol{c}_{i,j})]}. \tag{3.4}$$

The above normalization operation bears some resemblance to the soft-max function but does not generate a sparse output. Finally,we define a more discriminative representation $\boldsymbol{P}_n$ by

$$\boldsymbol{r}_{i,j} = \frac{\boldsymbol{c}_{i,j} - \min(\boldsymbol{c}_{i,j})}{\sum_{i,j}[\boldsymbol{c}_{i,j} - \min(\boldsymbol{c}_{i,j})]}. \tag{3.5}$$

The above normalization operation bears some resemblance to the soft-max function but does not generate a sparse output. Finally,we define a more discriminative representation $\boldsymbol{P}_n$ by

$$\boldsymbol{P}_n = \sum_{i=1}^{N} \boldsymbol{r} \times \boldsymbol{F}_n^i. \tag{3.6}$$

Through the use of this attention module, the proposed network is able to tell exactly where to look at the slice level, and further retrieve the visual explanation of a finer scale.

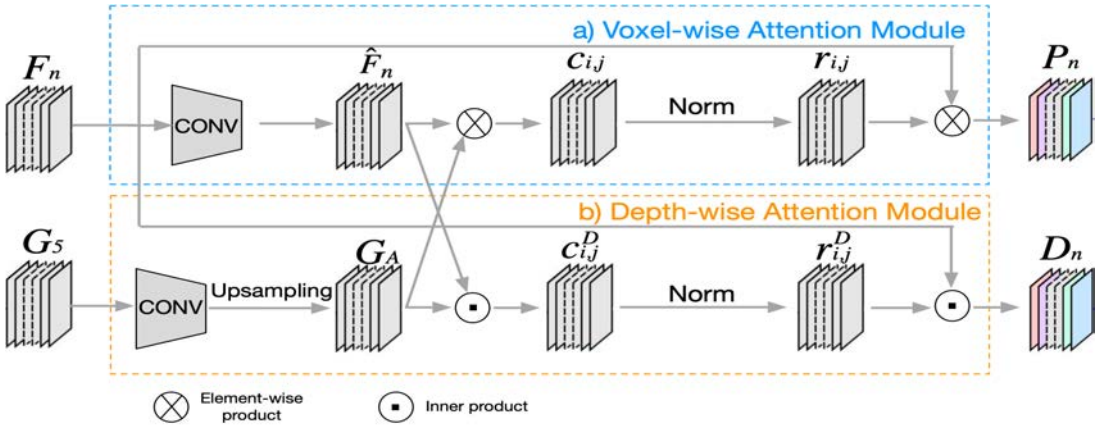


Figure 3.3: Block diagram of the Dual Attention Module (DAM), which is embedded into several stages of MAXNet, with the objective of capturing both voxel-wise and depth-wise dependencies and variations of feature maps $\boldsymbol{P}_n$ and $\boldsymbol{D}_n$ in hidden layers simultaneously.

Depth-wise attention

As depicted in Fig. 3.3(b), we propose a depth-wise attention module to perceive 3D context between slices. With $\hat{\boldsymbol{F}}_n$ and $\boldsymbol{G}_A$ acquired by Eq. (3.2), we do a transpose operation on them to get $\hat{\boldsymbol{F}}_n^{\mathrm{T}} \in R^{H_n^{'} \times W_n^{'} \times D_n^{'} \times C_n}$ and $\boldsymbol{G}_A^{\mathrm{T}} \in R^{H_n^{'} \times W_n^{'} \times D_n^{'} \times C_n}$. Then the inner product is taken

$$\boldsymbol{c}^D_{i,j} = \hat{\boldsymbol{F}}_n^{\mathrm{T}}(i)\boldsymbol{G}_A^{\mathrm{T}}(j), \tag{3.7}$$

and the obtained $\boldsymbol{c}^D_{i,j}$ is normalized by Eq. (3.4) to get $\boldsymbol{r}^D_{i,j}$. Subsequently, a depth-wise feature map $\boldsymbol{D}_n$ is computed by

$$\boldsymbol{D}_n = \textstyle\sum_{i=1}^{N} \boldsymbol{r}^D \boldsymbol{F}_n^i. \tag{3.8}$$

By predicting the result based on all $\boldsymbol{P}_n$ and $\boldsymbol{D}_n$ with finer and diverse receptive fields for views at both the voxel and channel levels, the network is enhanced to concentrate on the most considerable partial regions, boost the influence of subtle distinctions, and inhibit the background or trivial noise. Briefly speaking, the advantages of this proposed mechanism can be proclaimed on three fronts: 1) employing the voxel-wise attention allows low-scale stages to pay more attention on learning both local and global context attributes; 2) with the elaborate design of a depth-wise attention block, the model is extended to learn complex and flexible correlations between 3D features; 3) the DAM is significant since data of medical imaging are intrinsically noisy. In this case, a trainable block other than a linear parameter may be easier to achieve the global optimum.

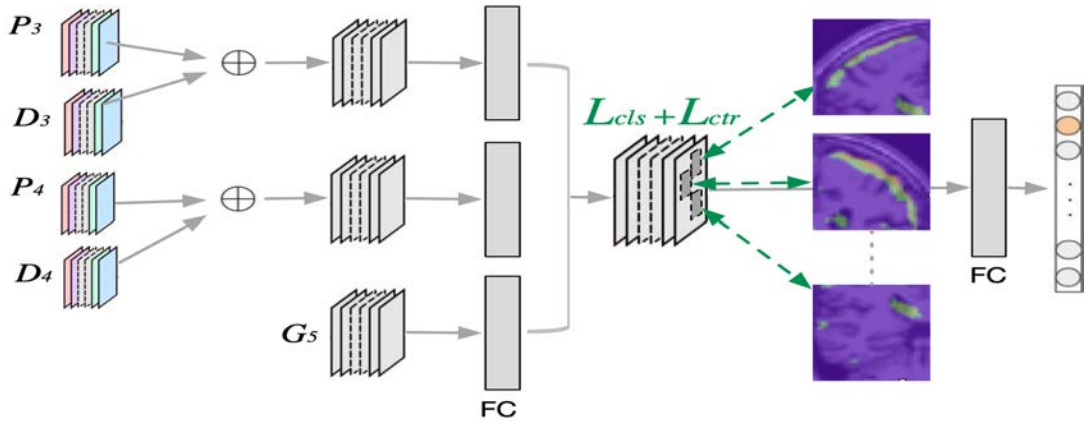


Figure 3.4: Block diagram of the Multi-resolution Fusion Module (MFM), which aggregates multi-resolution features $\boldsymbol{P}_n$, $\boldsymbol{D}_n$, and $\boldsymbol{G}_5$ by use of several fully-connected layers.

3.4.3 Multi-resolution Fusion Module (MFM)

In order to encourage the diversity of learned feature activations and enforce these features to be close to the label of its input, we construct the Multi-resolution Fusion Module (MFM) to combine multi-resolution features. The structure of the MFM is illustrated in Fig. 3.4.

We argue that a fusion module is supposed to be adaptive and can be fine-tuned in accordance with specific application scenarios. Firstly, we combine $\boldsymbol{P}_n$ and $\boldsymbol{D}_n$ as follows:

$$\hat{\boldsymbol{F}}_n = \beta_1 \boldsymbol{P}_n + \beta_2 \boldsymbol{D}_n, \; n \in [3,4], \tag{3.9}$$

where β_1 and β_2 are set to be 0.5 initially and learnable by the back propagation algorithm.

Secondly, we define a set of important class discriminative latent features $\boldsymbol{p}$ for the input $\boldsymbol{m}_i$ as follows:

$$\begin{aligned} \boldsymbol{p}(\boldsymbol{m}_i) = \max\Bigg(ReLU\Bigg(N\Bigg(\frac{1}{Z_n} \sum_{m} \sum_{p \in R^{i,j,k}} \hat{\boldsymbol{F}}_n(i,j,k) \Bigg) \Bigg), \\ U\Bigg(ReLU\Bigg(N\Bigg(\frac{1}{Z_5} \sum_{m'} \sum_{p \in R^{i,j,k}} \boldsymbol{G}_5(i,j,k) \Bigg) \Bigg) \Bigg) \Bigg), n \in [3,4], \end{aligned} \tag{3.10}$$

where Z_n / Z_5 is the number of convolution filters in $\hat{\boldsymbol{F}}_n / \boldsymbol{G}_5$. By using Eq. (3.10), $\boldsymbol{p}$ offers more accurate localization of important features by considering maximum values both from the intermediate features and the last convolutional features. We then project these latent features as

$$\boldsymbol{z}_j = -\log(||\hat{\boldsymbol{z}}_j - \boldsymbol{p}_j||_2^2) + \eta, \quad j \in \{1, ..., N\}, \tag{3.11}$$

where $\hat{z}_j$ is extracted from $\hat{\boldsymbol{F}}_3$, $\hat{\boldsymbol{F}}_4$ and $\boldsymbol{G}_5$ after FC layers. η is empirically set to $1e-4$. The final output is produced by

$$\hat{y}_i = \text{argmax}(\text{softmax}(FC(\boldsymbol{z}))). \tag{3.12}$$

The intuition behind this is that the predicted score $FC(\boldsymbol{z})$ w.r.t. $\hat{y}_i$ is high when latent features $\boldsymbol{p}$ preserved by $\boldsymbol{z}$ are important. In that case, the model is able to learn good representations by merely asking the latent features $\boldsymbol{p}$ to be close to its predicted label.

3.4.4 Loss Function

Although both DAM and MFM provide a strong capacity for feature learning, it is non-trivial to obtain interpretable representations without additional regularization. Therefore, we propose to learn a meaningful latent space via additional objective constraints i.e., cluster loss $\mathcal{L}_{\text{cls}}$ and contrastive loss $\mathcal{L}_{\text{ctr}}$. With which the most important features are clustered around the ground-truth label, and are well separated from features related to other labels. We achieve this goal by jointly optimizing the following loss function

$$\mathcal{L} = \mathcal{L}_{ce} + \alpha_1 \mathcal{L}_{\text{cls}} + \alpha_2 \mathcal{L}_{\text{ctr}}, \tag{3.13}$$

$$\mathcal{L}_{\text{cls}} = \frac{1}{n} \sum_{i=1}^{n} \min_{\hat{\boldsymbol{z}}_i} ||\hat{\boldsymbol{z}} - \boldsymbol{p}_i||_2^2, \quad \boldsymbol{p}_i \in \boldsymbol{p}^{y_i}, \tag{3.14}$$

$$\mathcal{L}_{\text{ctr}} = -\frac{1}{m} \sum_{i=1}^{m} \min_{\hat{\boldsymbol{z}}_i} ||\hat{\boldsymbol{z}}_i - \boldsymbol{p}_i||_2^2, \quad \boldsymbol{p}_i \notin \boldsymbol{p}^{y_i}, \tag{3.15}$$

$$\begin{aligned} \boldsymbol{p}^{y_i} = \max\Bigg(& ReLU\Bigg(N\Bigg(\frac{1}{Z_n} \sum_{m} \sum_{p \in R^{i,j,k}} \boldsymbol{s}_n^{y_i} \hat{\boldsymbol{F}}_n(i,j,k) \Bigg) \Bigg), \\ & U\Bigg(ReLU\Bigg(N\Bigg(\frac{1}{Z_5} \sum_{m'} \sum_{p \in R^{i,j,k}} \boldsymbol{s}_5^{y_i} \boldsymbol{G}_5(i,j,k) \Bigg) \Bigg) \Bigg) \Bigg), n \in [3,4], \end{aligned} \tag{3.16}$$

where $\boldsymbol{s}_n^{y_i}$ is the predicted score, $\mathcal{L}_{ce}$ is the cross-entropy loss and α_1, α_2 are hyper-parameters. Intuitively, minimizing the $\mathcal{L}_{\text{cls}}$ encourages the model to have at least one representation similar to its true label's latent features, while the contrastive loss $\mathcal{L}_{\text{ctr}}$ penalizes the similarity between its representations and other labels' features.

Consequently, with the $\mathcal{L}_{\text{cls}}$ and $\mathcal{L}_{\text{ctr}}$ terms, the loss function in Eq. (3.13) encourages our model to learn and cluster the latent features into a semantically meaningful space, which facilitates the prediction of MAXNet and the generation of fine-grained interpretable heatmaps.

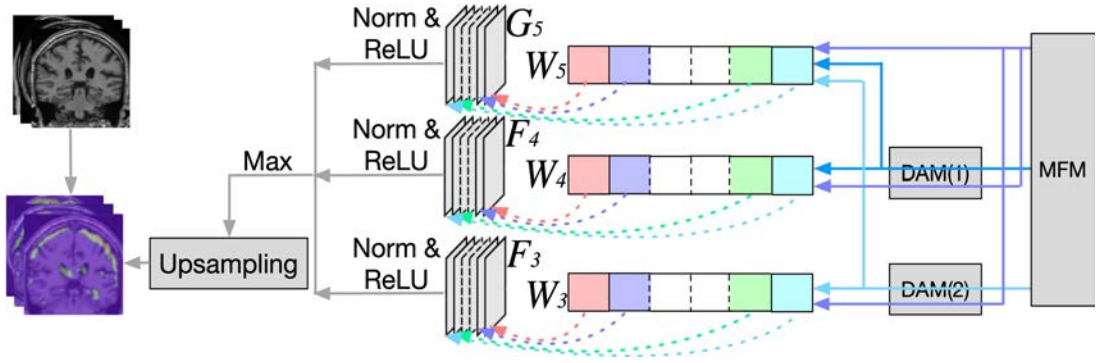


Figure 3.5: Block diagram of the High-resolution Activation Mapping (HAM). Each arrow shows the gradient of the classification logit. Our method takes intermediate activations as inputs, and considers the maximum values from the intermediate features $\boldsymbol{F}_3$ and $\boldsymbol{F}_4$ as well as the final activation $\boldsymbol{G}_5$, which offers more accurate localization.

3.5 Explaining the MAXNet Predictions

In this section, we propose HAM to capture fine-grained heatmaps $\boldsymbol{A}_{\text{HAM}}$ from a different perspective. Also, this section will elaborate on the PCR module, to provide supplemental evidence of reference samples with ground-truth labels.

3.5.1 Proposed HAM for High-resolution Heatmaps

Most existing visualization methods only consider the last convolutional layer, which extracts global features of the pathological abnormalities and misses the small subjects and discrepancies. Instead, we propose High-resolution Activation Mapping (HAM), which consider values from both the intermediate features and the last convolutional features, to offer more accurate localization. The detailed structure is depicted in Fig. 3.5. Recall $\hat{y}_i$ is the model's predicted label for the input $\boldsymbol{m}_i$, and $\boldsymbol{s}_i$ is its corresponding predicted score before the soft-max function. Based on the proposed DAM operated onto $\boldsymbol{F}_n$ in stages 3 and 4, it follows

$$\frac{\partial \boldsymbol{s}_i}{\partial \boldsymbol{F}_n} = \boldsymbol{W}_n^1 + \boldsymbol{W}_n^2, \tag{3.17}$$

where $\frac{\partial \boldsymbol{s}_i}{\partial \boldsymbol{F}_n}$ is the gradient of $\boldsymbol{s}_i$ w.r.t. $\boldsymbol{F}_n$, and it is decomposed into two terms. One is $\boldsymbol{W}_n^1$ derived from stages, and $\boldsymbol{W}_n^2$ is the gradient flowing back from the DAM. Then the visualization map $\boldsymbol{A}_{\text{HAM}}$ is given by

$$\begin{aligned} \boldsymbol{A}_{\text{HAM}} = U\Bigg(\max\Bigg(ReLU\Bigg(N\Bigg(\frac{1}{m}\sum_{m}\frac{\partial \boldsymbol{s}_i}{\partial \boldsymbol{F}_3}\boldsymbol{F}_3(i,j,k)\Bigg)\Bigg)\Bigg), \\ U\Bigg(ReLU\Bigg(N\Bigg(\frac{1}{m'}\sum_{m'}\frac{\partial \boldsymbol{s}_i}{\partial \boldsymbol{F}_n}\boldsymbol{F}_n(i,j,k)\Bigg)\Bigg)\Bigg)\Bigg), n \in [4,5], \end{aligned} \tag{3.18}$$

where $F_5 = G_5$, m and m' are the number of convolution filters for F_3 and F_n.

Overall, this method takes intermediate activations as well as the features from the last convolutional layer as input, which is certainly different from state-of-the-art methods which only employ the last convolutional features. Therefore, compared to other techniques that produce blurry maps and lose too much discriminative information, our HAM approach is able to learn and identify high-resolution features of brain areas through capturing diverse cues successfully.

3.5.2 Prediction-basis Creation and Retrieval (PCR)

Fig. 3.2 illustrates the proposed PCR module for our explainable model MAXNet. Our intuition is that we want to identify samples that have morphologically similar features compared to the input volume. First, let us define the so-called reference sample and formulate the problem that the PCR aims to tackle.

Definition 1 (Reference Sample) *Given an MRI volume $\boldsymbol{m}_k^{\mathrm{T}}, k \in [1, 2, ..., K]$ in the test dataset and the proposed model MAXNet $\psi(\cdot)$, $\boldsymbol{R}_c$ is called a reference sample of $\boldsymbol{m}_k^{\mathrm{T}}$ when*

$$\boldsymbol{R}_c = \mathrm{argmin}_{\boldsymbol{R}_c} D(\boldsymbol{R}_c, \boldsymbol{m}_k^{\mathrm{T}}) \quad s.t.\ \psi(\boldsymbol{m}_k^{\mathrm{T}}) = \psi(\boldsymbol{R}_c), \tag{3.19}$$

where $D(\cdot)$ is a function of evaluating the similarity between $\boldsymbol{R}_c$ and $\boldsymbol{m}_k^{\mathrm{T}}$.

Problem 1 *Given the $\boldsymbol{m}_k^{\mathrm{T}}$ and our model $\psi(\cdot)$, let $\hat{y}_k^{\mathrm{T}} = \psi(\boldsymbol{m}_k^{\mathrm{T}})$, and $\boldsymbol{p}^k$ are the latent features of $\boldsymbol{m}_k^{\mathrm{T}}$. The goal is to retrieve reference samples $\{\boldsymbol{R}_c, y_c\}_{c=1}^{3}$ and corresponding latent features $\boldsymbol{p}^c$ similar to $\boldsymbol{p}^k$, with the objective of providing instance-level justifications for the model output $\hat{y}_k^{\mathrm{T}}$.*

We found that the well-trained model MAXNet is able to learn pivotal and various features in brain images, e.g., the atrophy of cerebral cortex and hippocampus, the enlargement of frontal and temporal horns of the lateral ventricles, and the enlarged sulcal spaces with atrophy of gyri. These pathological changes are believed to be important for AD diagnosis by expert clinicians [80]. As a result, we consider the similarity of two generated latent representations since they consist of a set of representative features for predictions. That is, the retrieved $\boldsymbol{p}^c$ is minimally different from $\boldsymbol{p}^k$.

Firstly, given each of training data $\{\boldsymbol{m}_i, y_i\}$, MAXNet predicts its label $\hat{y}_i$ and calculates its latent representation $\boldsymbol{p}^{\hat{y}_i}$ by Eq. (3.16). In what follows, we construct an auxiliary diagnosis reference set based on the training dataset, which contains both volumes $\boldsymbol{R}_{\mathrm{ref}}$ as potential reference samples and corresponding latent features $\boldsymbol{p}$. Specifically, $\boldsymbol{R}_{\mathrm{ref}}$ consists of subsets $\boldsymbol{R}_{\mathrm{ref}}^i$, and $\boldsymbol{p}$ contains subset $\boldsymbol{p}^{y_i}$ for $y_i, i \in [0, 1]$.

Secondly, given an input $\boldsymbol{m}_k^{\mathrm{T}}$ during testing, MAXNet yields the prediction $\hat{y}_k^{\mathrm{T}}$ and PCR retrieves reference samples $\boldsymbol{R}_c$ with label y_c by Eq. (3.19). Where define the similarity evaluation function $D(\cdot)$ as follows

$$D(\boldsymbol{R}_c, \boldsymbol{m}_k^{\mathrm{T}}) = \frac{1}{n}\sum_{j=1}^{n} ||\boldsymbol{p}_j^c - \boldsymbol{p}_j^k||_2^2, \quad \boldsymbol{p}_j^k \in \boldsymbol{p}^{y_k^{\mathrm{T}}}, \boldsymbol{p}_j^c \in \boldsymbol{p}^{y_c}, \tag{3.20}$$

where $||\cdot||_2$ is the L2 norm, $\boldsymbol{p}^{y_k^{\mathrm{T}}}$ and $\boldsymbol{p}^{y_c}$ are calculated by Eq. (3.16). This applies to our intuition that a reference sample should contain the discriminative features that are highly aligned with the input sample's.

With Eq. (3.19) and Eq. (3.20), PCR is able to retrieve samples $\boldsymbol{R}_c$ with latent representations $\boldsymbol{p}^c$ that are morphologically similar to $\boldsymbol{m}_k^{\mathrm{T}}$'s features $\boldsymbol{p}^k$. $\boldsymbol{p}^c$ can be particularly beneficial for internists since the most important features for classifying are retained. By utilizing the PCR module, it will not only help us gain trust and acknowledgment of human users in evidence-centered fields such as medical imaging, but also provide scientific confidence in real-world applications. Moreover, we believe this module can be extended to benefit other interpretable processes for multi-classification problems.

3.5.3 Reasoning Process of our PCR Module

In this section, we present a probabilistic explanation for the proposed PCR's reasoning process.

Firstly, we consider the classification task as a problem of estimating conditional probabilities, in which our goal is to obtain the conditional distribution $P(Y = y_k, \boldsymbol{X} = \boldsymbol{m}_k^{\mathrm{T}})$. Inspired by Bayes' Theorem, the problem can be further expressed as:

$$P(Y = y_k | \boldsymbol{X} = \boldsymbol{m}_k^{\mathrm{T}}) = \frac{P(\boldsymbol{X} = \boldsymbol{m}_k^{\mathrm{T}} | Y = y_k) P(Y = y_k)}{\sum_i P(\boldsymbol{X} = \boldsymbol{m}_k^{\mathrm{T}} | Y = y_i) P(Y = y_i)}. \tag{3.21}$$

Then we define a group of latent features $\boldsymbol{z}_i^k(\boldsymbol{m}_k^{\mathrm{T}}) = \mathrm{argmin}_{\boldsymbol{z}_i} ||\boldsymbol{z}_i - \boldsymbol{p}_i||_2^2, \quad \boldsymbol{p}_i \in \boldsymbol{p}^{y_k}$, where $\boldsymbol{z}_i$ is sampled from $\hat{\boldsymbol{z}}_i$, $\boldsymbol{p}^{y_k}$ is computed by Eq. (3.16). Here we assume that for any input $\boldsymbol{m}_k^{\mathrm{T}}$, there exists only one latent representation $\boldsymbol{z}_i$ that is most similar to $\boldsymbol{p}_i$. As a result, we can make a reasonable assumption that $\boldsymbol{z}^k(\boldsymbol{m}_k^{\mathrm{T}})$ contains sufficient information about y_k. Then we prove the label-conditional probability $P(\boldsymbol{X} = \boldsymbol{m}_k^{\mathrm{T}} | Y = y_k)$ can be written as follows:

$$\begin{aligned} P(\boldsymbol{X} = \boldsymbol{m}_k^{\mathrm{T}} | Y = y_k) = P(\boldsymbol{X} = \boldsymbol{m}_k^{\mathrm{T}} | \boldsymbol{z}_1^k(\boldsymbol{m}_k^{\mathrm{T}}) = \boldsymbol{z}_1, ..., \boldsymbol{z}_n^k(\boldsymbol{m}_k^{\mathrm{T}}) = \boldsymbol{z}_n, Y = y_k) \\ \cdot P(\boldsymbol{z}_1^k(\boldsymbol{m}_k^{\mathrm{T}}) = \boldsymbol{z}_1, ..., \boldsymbol{z}_n^k(\boldsymbol{m}_k^{\mathrm{T}}) = \boldsymbol{z}_n | Y = y_k). \end{aligned} \tag{3.22}$$

Based on Eq. (3.22) it can be concluded that if $\boldsymbol{X} = \boldsymbol{m}_k^{\mathrm{T}}$, then the probability of $z_1^k(\boldsymbol{m}_k^{\mathrm{T}}) = z_1, ..., z_n^k(\boldsymbol{m}_k^{\mathrm{T}}) = z_n$ should be 1. Subsequently, we make another assumption

$$\begin{aligned} &P(\boldsymbol{X} = \boldsymbol{m} | z_1^k(\boldsymbol{m}) = z_1, ..., z_n^k(\boldsymbol{m}) = z_n, Y = y_k) \\ &= P(\boldsymbol{X} = \boldsymbol{m} | z_1^j(\boldsymbol{m}) = z_1, ..., z_n^j(\boldsymbol{m}) = z_n, Y = y_j), \quad \forall \boldsymbol{m} \in \boldsymbol{m}^{\mathrm{T}}, \quad \forall y_k, y_j \in \{0, 1\}, \end{aligned} \tag{3.23}$$

which means that for a given label y_k or y_j, the probability that $\boldsymbol{m}$'s latent features $z(\boldsymbol{m})$ are most similar to y_k or y_j is essentially the same. Plugging Eq. (3.22) and Eq. (3.23) into Eq. (3.21) gives rise to

$$P(Y = y_k | \boldsymbol{X} = \boldsymbol{m}_k^{\mathrm{T}}) = \frac{P(z_1^k(\boldsymbol{m}) = z_1, ..., z_n^k(\boldsymbol{m}) = z_n | Y = y_k) P(Y = y_k)}{\sum_j P(z_i^j(\boldsymbol{m}) = z_i, ..., z_n^j(\boldsymbol{m}) = z_n | Y = y_j) P(Y = y_j)}, \tag{3.24}$$

where $P(z_1^k(\boldsymbol{m}) = z_1, ..., z_n^k(\boldsymbol{m}) = z_n | Y = y_k) = \mu(||\boldsymbol{z} - \boldsymbol{p}^{y_k}||_2^2)$ is the optimal distribution based on our loss function in Eq. (3.14).

Based on the above equations, it can be concluded that a reference sample $\boldsymbol{R}_c$ which has latent representations $\boldsymbol{p}^c$ theoretically guarantees the accurate information of instance-level explanations provided by $\boldsymbol{R}_c$ if it satisfies $\boldsymbol{R}_c = \mathrm{argmin}_{\boldsymbol{R}_c} \frac{1}{n} \sum_{j=1}^{n} ||\boldsymbol{p}_j^c - \boldsymbol{p}_j^k||_2^2$.

We further analyze the impact of a reference sample on the original prediction accuracy.

Theorem 3.5.1 *Given an MRI volume $\boldsymbol{m}_k^{\mathrm{T}}$ and model $\psi(\cdot)$, $\boldsymbol{R}_c$ is a reference sample of $\boldsymbol{m}_k^{\mathrm{T}}$ with label y_c. $\hat{y}_k^{\mathrm{T}} = \psi(\boldsymbol{m}_k^{\mathrm{T}})$, the latent representations for $\hat{y}_k^{\mathrm{T}}$ and y_c are $\boldsymbol{p}^k$ and $\boldsymbol{p}^c$, respectively, and $\boldsymbol{z}_j^{y_k}$ is extracted using Eq. (3.11). Assume that:*

- *$\exists$ 0< ξ <1, $||\boldsymbol{p}_j^c - \boldsymbol{p}_j^k||_2 \leq (\sqrt{1+\xi} - 1)||\boldsymbol{z}_j^{y_k} - \boldsymbol{p}_j^k||_2$ and $||\boldsymbol{z}_j^{y_k} - \boldsymbol{p}_j^k||_2 \leq \sqrt{1-\xi}$ $j \in \{1, ..., n\}$;*
- *the weight in the last FC layer is 1 for $\boldsymbol{z}_j^{y_k} = -\mathrm{argmax}_{\hat{\boldsymbol{z}}_j} \log(||\hat{\boldsymbol{z}}_j - \boldsymbol{p}_j^{y_i}||_2^2) + \eta$, and 0 otherwise.*

Then using $\boldsymbol{p}^c$ in lieu of $\boldsymbol{p}^k$ can modify $\psi(\cdot)$'s predicted logit for at most $\Delta_k = \sum_j \log(1 + \xi), j \in \{1, ..., N\}$. If the logit score between correct and incorrect labels are at least $2\Delta_k$, then the latent features $\boldsymbol{p}^c$ of the reference sample $\boldsymbol{R}_c$ can be used to correctly explain $\psi(\cdot)$'s decision about $\boldsymbol{m}_k^{\mathrm{T}}$.

Proof 1 *Denote by s_k $\psi(\cdot)$'s logit score with the correctly predicted label $\hat{y}_k^{\mathrm{T}}$. Then it follows from Eq. (3.11) that*

$$s_k = -\sum_j \log(||\boldsymbol{z}_j^{y_k} - \boldsymbol{p}_j^k||_2^2) + \eta, j \in \{1, ..., N\}. \tag{3.25}$$

Let Δ_k be the logit change by choosing reference sample $\boldsymbol{R}_c$ to explain $\psi(\cdot)$'s decision about $\boldsymbol{m}_k^{\mathrm{T}}$. Then we have

$$\Delta_k = s_k - s_c = \sum_j \log\left(\frac{||\boldsymbol{z}_j^{y_k} - \boldsymbol{p}_j^c||_2^2}{||\boldsymbol{z}_j^{y_k} - \boldsymbol{p}_j^k||_2^2}\right) \tag{3.26}$$

According to the assumptions in Theorem 1, we have $||\boldsymbol{p}_j^c - \boldsymbol{p}_j^k||_2 \geq (\sqrt{1+\xi} - 1)||\boldsymbol{z}_j^{y_k} - \boldsymbol{p}_j^k||_2$, and $||\boldsymbol{z}_j^{y_k} - \boldsymbol{p}_j^c||_2 \leq ||\boldsymbol{z}_j^{y_k} - \boldsymbol{p}_j^k||_2 + ||\boldsymbol{p}_j^c - \boldsymbol{p}_j^k||_2$, which in turn gives us

$$\Delta_k = \sum_j \log\left(\frac{||\boldsymbol{z}_j^{y_k} - \boldsymbol{p}_j^c||_2^2}{||\boldsymbol{z}_j^{y_k} - \boldsymbol{p}_j^k||_2^2}\right) \leq \sum_j \log(1+\xi). \tag{3.27}$$

Subsequently, we suppose that the corrected logic score s_k is $2\Delta_k$ larger than any other incorrect score s_i, i.e., $s_k \geq s_i + 2\Delta_k$. Therefore, when using the reference sample $\boldsymbol{R}_c$'s latent features $\boldsymbol{p}^c$ to explain $\psi(\cdot)$'s decision about $\boldsymbol{m}_k^{\mathrm{T}}$, we have

$$s_c \geq s_k - \Delta_k \geq s_i. \tag{3.28}$$

Given Eq. (3.28), we can claim that model $\psi(\cdot)$ still can correctly classify the volume with the provided latent representations $\boldsymbol{p}^c$ from the reference sample $\boldsymbol{R}_c$.

It is noted in the experiments that in our well-trained MAXNet, the assumption always holds that $s_k \geq s_i + 2\Delta_k$. Moreover, the distance $||\boldsymbol{p}_j^k - \boldsymbol{p}_j^c||_2$ is generally smaller than $||\boldsymbol{z}_j^{y_k} - \boldsymbol{p}_j^k||_2$, which verifies our assumptions and in turn confirms the effectiveness of our PCR module. Empirically, the value of ξ is set to 0.24.

3.5.4 Metrics for Evaluation of PCR

In order to evaluate the accuracy of the reference set $\boldsymbol{R}_c$, we design two evaluation metrics and conduct a series of experiments to quantify the effectiveness of reference samples.

Definition 2 (Swap Deletion Confidence)

$$\rho^{\mathrm{D}}_{m_k,r_c} = \frac{(s(\boldsymbol{m}_k^{\mathrm{T}}) - s(\boldsymbol{m}_k^{\mathrm{T}} \odot \boldsymbol{K})) \odot (s(\boldsymbol{m}_k^{\mathrm{T}}) - s(\boldsymbol{m}_k^{\mathrm{T}} \odot \boldsymbol{C}))}{||s(\boldsymbol{m}_k^{\mathrm{T}}) - s(\boldsymbol{m}_k^{\mathrm{T}} \odot \boldsymbol{K})||_2 - ||s(\boldsymbol{m}_k^{\mathrm{T}}) - s(\boldsymbol{m}_k^{\mathrm{T}} \odot \boldsymbol{C})||_2}, \tag{3.29}$$

where $s(\cdot)$ is the predicted score, $\odot$ is the hadamard product. $\boldsymbol{K}$, $\boldsymbol{C}$ are with the same dimension as $\boldsymbol{m}_k^{\mathrm{T}}$. For each $k_i \in \boldsymbol{K}$, $k_i = 0$ if the position i is located in $\boldsymbol{p}^k$, otherwise $k_i = 1$. For $c_j \in \boldsymbol{C}$, $c_j = 0$ if the position j is located in $\boldsymbol{p}^c$, otherwise $c_j = 1$. Consequently, $\rho^{\mathrm{D}}_{m_k,r_c}$ measures the similarity between γ_o^i and γ_c^j.

As is detailed in Theorem 1, $\boldsymbol{p}^c$ and $\boldsymbol{p}^k$ have been proved to be expressive features for $\boldsymbol{m}_k^{\mathrm{T}}$'s prediction. Therefore, our intuition here is to evaluate if there are similar

changes of the predictions by removing features $\boldsymbol{p}^c / \boldsymbol{p}^k$ from $\boldsymbol{m}_k^{\mathrm{T}}$. Arguably, a larger $\rho^{\mathrm{D}}_{m_k, r_c}$ indicates $\boldsymbol{p}^c$ causes a similar decrease compared to $\boldsymbol{p}^k$ in prediction accuracy when removed.

Following Definition 2, we define the so-called Swap Insertion Confidence from a different yet complementary angle.

Definition 3 (Swap Insertion Confidence)

$$\rho^{\mathrm{I}}_{m_k, r_c} = \frac{s(\boldsymbol{m}_k^{\mathrm{T}} \odot \boldsymbol{K}') \odot s(\boldsymbol{m}_k^{\mathrm{T}} \odot \boldsymbol{C})}{||s(\boldsymbol{m}_k^{\mathrm{T}} \odot \boldsymbol{K}')||_2 - ||s(\boldsymbol{m}_k^{\mathrm{T}} \odot \boldsymbol{C})||_2}, \tag{3.30}$$

where for $k_i \in \boldsymbol{K}'$, $k_i = 1$ if the position i is located in $\boldsymbol{p}^k$, otherwise $k_i = 0$. For $c_j \in \boldsymbol{C}$, $c_j = 1$ if the position j is located in $\boldsymbol{p}^c$, otherwise $c_j = 0$. Likewise, the intuition behind this is to evaluate the similarity of prediction changes by adding features from $\boldsymbol{p}^c / \boldsymbol{p}^k$ into $\boldsymbol{m}_k^{\mathrm{T}}$. A larger $\rho^{\mathrm{I}}_{m_k, r_c}$ means $\boldsymbol{p}^c$ causes a similar increase compared to $\boldsymbol{p}^c$ in prediction accuracy when added. In the following sections, we will provide experimental results concerning these two evaluation metrics.

3.6 Experimental Results

3.6.1 Dataset

The datasets used in our experimental studies is the the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset [103], [104]. We select the T1 weighted, pre-processed, baseline MRI data in the ADNI dataset, and a single scan per subject visit was selected. To address the issue of explainability and keep the classification task simple, we only select two diagnosis groups in ADNI, which contain 826 cognitively normal individuals and 422 Alzheimer's patients with at least one session's MRI volumes available. With the consideration of data heterogeneity, we carefully extract data samples from the ADNI dataset to form three non-overlapping subsets. Each subset is further split into 1779 images for training, 427 for validation, and 575 for testing. In order to avoid biased generalization estimates due to same subject image similarities, each subject is only selected into just one of the sets (i.e., the training, validation, and test sets) for each subset. Finally, each of the volume is further cropped into size $169 \times 208 \times 179$ for training and validation, and test.

3.6.2 Implementation Details

Our proposed MAXNet is implemented in PyTorch and executed on two Nvidia Volta V100 GPUs with 16 GB memory each. It is trained using the Adam optimizer with a

weight decay value of 0.0005, and the batch size is fixed to 8 samples. The initial learning rate is set to be 0.0001 and will be decayed according to a polynomial schedule. We pre-train the model with the cross-entropy loss function $\mathcal{L}_{ce}$ in Eq. (3.13) for the initial 20 epochs and fine-tune it with the cluster and contrastive losses for 50 epochs. The value of hyper-parameter α_1, α_2 are set to be 0.6 and 0.06 respectively after conducting extensive experiments.

3.6.3 Performance of MAXNet

We compare the classification performance of the proposed architecture with other interpretable models. Following [105], we resort to two XAI properties, i.e., continuity and selectivity (more details can be found in [105]), to qualify the interpretability of the MAXNet.

Table 3.2: Comparative results of various interpretable models on ADNI.

Model	Subject-(AD / NC)	ACC	AUC	Continuity	Selectivity
Lee *et al.* [83]	198 / 229	0.9275	0.9804	-	-
3DAN [101]	227 / 305	0.861	0.912	-	-
Kroll *et al.* [95]	153 / 306	-	0.815	-	-
VGGNet 3D [106]	47 / 56	0.766	0.863	-	-
ResNet 3D [106]	47 / 56	0.854	0.794	-	-
AlexNet 2D+C [105]	422 / 826	-	0.923	30.361	-0.059
AlexNet 3D [105]	422 / 826	-	0.898	37.887	0.215
VGG16 2D+C [105]	422 / 826	-	0.892	24.928	0.224
VGG16 3D [105]	422 / 826	-	0.886	41.879	0.039
MAXNet(Subset 1)	422 / 826	**0.928**	**0.959**	**14.61**	**-0.79**
MAXNet(Subset 2)	422 / 826	**0.953**	**0.978**	**15.22**	**-0.87**
MAXNet(Subset 3)	422 / 826	**0.954**	**0.980**	**15.27**	**-0.71**

The comparison results are presented in Table 3.2. It is noted that the existing interpretable models perform evaluation with different cohorts of subjects and the indices of those subjects were not disclosed. To take into account data heterogeneity, we train and evaluate our model on the three non-overlapping subsets extracted from the ADNI dataset. Tables 3.2 reports not only the classification results of the comparison models, but also the number of subjects used by each model. As can be observed from the table, AlexNet 3D [105] shows a slightly better classification performance compared to VGGNet 3D [106]. Lee's model in [83] yields better prediction outcomes compared

to its VGG or AlexNet counterparts, as it derives complex nonlinear relationships for predefined regions. The experimental results presented in Table 3.2 demonstrates that our proposed MAXNet clearly outperforms its competitors, as it obtains the highest accuracy of 95.4%, and second highest AUC of 98.0% on subset 3 from the ADNI subset. Although Lee's model in [83] achieves a slightly higher AUC score than our MAXNet, its accuracy is lower than ours and its results were validated on much less MRI data (427 vs. 1248 subjects). As a result, the experimental results reported in Table 3.2 suggests that our proposed model MAXNet is capable of offering accurate diagnoses for AD vs. NC classification. Last but not the least, it should be noted that the heterogeneity among the data samples drawn from ADNI is not considered in [83], [101], [95], [106], [105], while our work proves to be robust across the samples in the ADNI dataset.

The results of continuity and selectivity metrics are also shown in Fig. 3.2. Note that we did not provide some models' results because they are not reported in relevant papers. Lower values for continuity suggest that similar MRI scans have more similar heatmaps. The continuity value of our proposed model is notably lower compared to other models, it reveals that our MAXNet achieves a continuity of 15.27 on subset 3, and a selectivity of -0.87 on subset 2. Both are consistently superior to the models considered in [105]. This means MAXNet is able to capture distinctive patterns of disease areas by use of the latent feature representations and produce consistent results given similar MRI scans. Moreover, selectivity quantifies the changes in prediction probability of classification when removing the related features gradually, lower values for selectivity means similar MRI volumes have similar relevant features in heatmaps [105]. As is observed, the selectivity value obtained via AlexNet 2D+C, AlexNet 3D, VGG16 2D+C, VGG16 3D [105] varies slightly, this confirms that there is a low correlation between the heatmaps and the predictions in these models. Instead, our model MAXNet achieves the lowest value of selectivity, which indicates the learned latent features are closed related to its final outputs, and well demonstrates its distinctive interpretability.

3.6.4 Qualitative Evaluation of HAM and PCR

Faithfulness and complexity evaluation of HAM

It has been confirmed by neurologists that AD often causes at least moderate cortical atrophy, enlargement of lateral ventricles, and temporal enlarged horns which are most macroscopic in MRI volumes [97]. For fair comparison against recent state-of-the-art methods, we implemented these approaches based on the provided source code on the same network with an AUC of 0.992 [82], the results can be found in Fig. 3.6. Fig. 3.6(a)

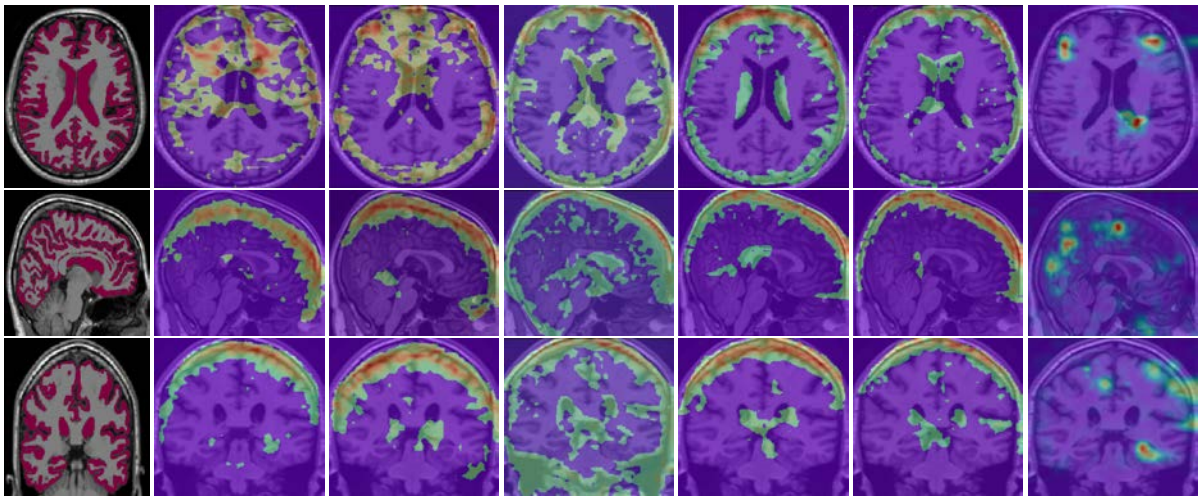
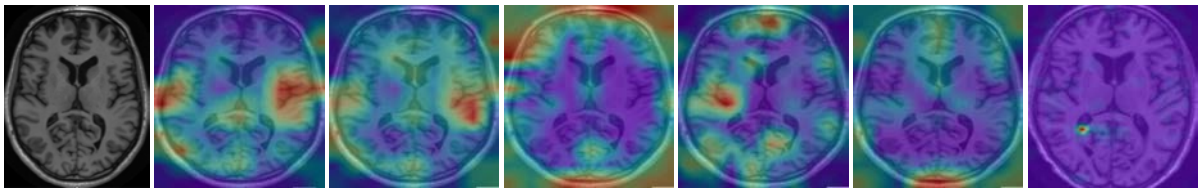


Figure 3.6: Visual results of visualization methods. Note that (a)-(f) were performed over a 3D CNN with an AUC of 0.992 [82]. (a) input with "AD" label. Ground truth of cerebral cortex, lateral ventricle and hippocampus via FreeSurfer are highlighted, (b) Grad-CAM [19], (c) Grad-CAM++ [38], (d) CAMERAS [22], (e) RISE [107], (f) Score-CAM [26], (g) proposed HAM-generated heatmaps which highlight enlarged sulcal spaces caused by atrophy and pathological abnormalities of cerebral cortex and hippocampus.

shows ground truth segmented areas of the cerebral cortex, lateral ventricle, and hippocampus for an MRI volume with AD. Subjective comparisons shown in Fig. 3.6(b) and Fig. 3.6(c) indicate that one nearly cannot identify distinct areas from the heatmaps generated by both the Grad-CAM [19] and Grad-CAM++ [38]. The heatmaps identify most of the white matter, whose lesions are not macroscopic when evaluating the brain disease and thus are not trustable visualization evidence. In addition, the CAMERAS [22] presented in Fig. 3.6(d) shows heatmaps with larger brain areas, but fails to identify pathological abnormalities and discriminative disease areas. Subsequently, RISE [107] and Score-CAM [26] obtain relatively unambiguous views of heatmaps, but still show known patterns of atrophy and fail to highlight the lateral ventricles and hippocampus. Finally, our proposed method HAM generates high-quality heatmaps (Fig. 3.6(g)), which show discriminative localizations of brain abnormalities instead of blurry heatmaps, making it outperform the existing works remarkably. This can be attributed to the effective latent features of aggregated representations from intermediate layers as well as the last convolutional layer.

Meanwhile, we also apply the above visualization methods to our proposed MAXNet. Fig. 3.7(a) shows an example of a normal MRI scan, while Figs. 3.7(b) - (f) present heatmaps produced by the state-of-the-art approaches. As can be observed, these methods do not perform well and some of them even render strange visual outcomes.



Input | Grad-CAM [19] | Grad-CAM++ [38] | CAMERAS [22] | RISE [107] | Score-CAM [26] | Ours

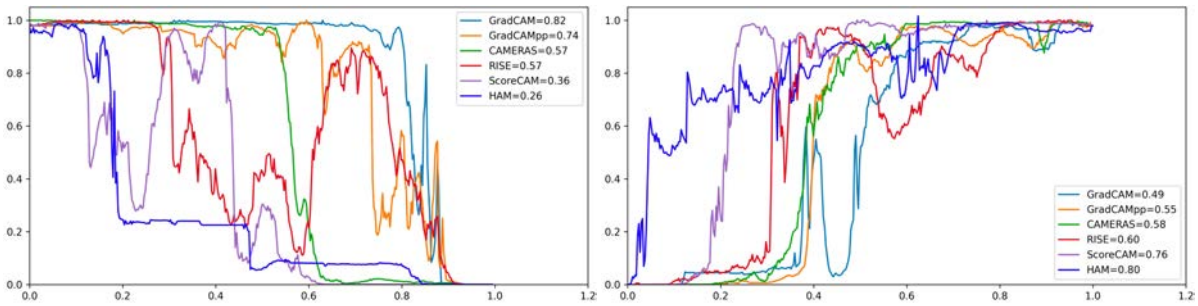
Figure 3.7: (a) Visualization results of different methods given an input with "Normal" label.

Table 3.3: Comparative evaluation of HAM and other methods

Visualization method	Insertion	Deletion	Runtime
Grad-CAM [19]	0.492	0.822	**0.027**
Grad-CAM++ [38]	0.554	0.743	0.030
RISE [107]	0.603	0.576	29.32
Score-CAM [26]	0.761	0.362	19.06
CAMERAS [22]	0.676	0.523	3.25
HAM	**0.801**	**0.263**	0.092

More importantly, some areas such as lateral ventricles and hippocampus are still emphasized even this is a normal case without AD (Fig. 3.7(d)). Therefore, we realize that existing visualization methods do not work well with our proposed explainable model MAXNet. By contrast, the proposed HAM shows significant better visualization results that can highlight small regions and ignore most of non-neuropathy areas as shown in Fig. 3.7(g).

We further evaluate HAM on three metrics: runtime, deletion and insertion, which are adopted in several recent works [105]. An example of the deletion and insertion curves for a test volume is illustrated in Fig. 3.8, and the average performance for 2500 perturbed volumes is given in Table 3.3. As is observed, compared to both Grad-CAM [19] and Grad-CAM++ [38], CAMERAS [22] provides better performance, this is potentially caused by the noisy heatmaps due to up-sampling operation. Score-CAM [26] obtains the state-of-the-art performance on both deletion and insertion metrics, but the deletion curve is steeply convex, which means its feature selection is not stable. Instead, our proposed HAM outperforms other approaches on both deletion and insertion metrics. This is also proof that the heatmaps generated via HAM are able to capture most of the salient features of brain disease areas and contain less noise. For the runtime metric, It can be observed that RISE [107], Score-CAM [26] and CAMERAS [22] all make considerable demands on the time. This is because augmented feature maps or inputs were fed into models multiple times. As expected, Grad-CAM [19] and



(a) (b)

Figure 3.8: (a) The deletion curve for Grad-CAM [19], Grad-CAM++ [38], CAMERAS [22] , RISE [107], Score-CAM [26], and HAM. The x-axis represents the percentage of removed voxels, while the y-axis is the corresponding predicted score. Specifically, a steeper slope indicates a better explanation. (b) The insertion curve for Grad-CAM [19], Grad-CAM++ [38], CAMERAS [22] , RISE [107], Score-CAM [26], and HAM. The x-axis shows the percentage of added voxels, and the y-axis is the corresponding predicted score. Specifically, a fast-rising slope implies a better explanation.

Table 3.4: Comparative evaluation of PCR.

	Swap Deletion Confidence		Swap Insertion Confidence	
	W/O PCR	PCR	W/O PCR	PCR
ρ_{m,r_1}	0.275	**0.784**	0.297	**0.771**
ρ_{m,r_2}	0.182	**0.803**	0.254	**0.755**
ρ_{m,r_3}	0.345	**0.818**	0.278	**0.764**

Grad-CAM++ [38] are the fastest methods, because they do not need to make multiple model predictions with perturbed inputs in theory. Although Grad-CAM [19] achieves a faster runtime than our HAM, it obtains substantially lower insertion (0.492 vs. 0.801 insertion) and higher deletion (0.822 vs. 0.263 deletion) than ours. These promising results suggest that our proposed approach HAM is able to identify the saliency features that are responsible for the model decisions.

Qualitative analysis of PCR

Table 3.4 compares the average values of the proposed metrics Swap Deletion Confidence and Swap Insertion Confidence that are generated with and without PCR (i.e., retrieve samples randomly from MRI scans with the same label). Compared to the baseline which provides relevant samples stochastically, the PCR module is able to find MRI cases with similar pathological abnormalities. As a result, it obtains higher values of the Swap Deletion Confidence and Swap Insertion Confidence. Fig. 3.9 displays an example of a test MRI volume with the label "AD" and three reference samples.

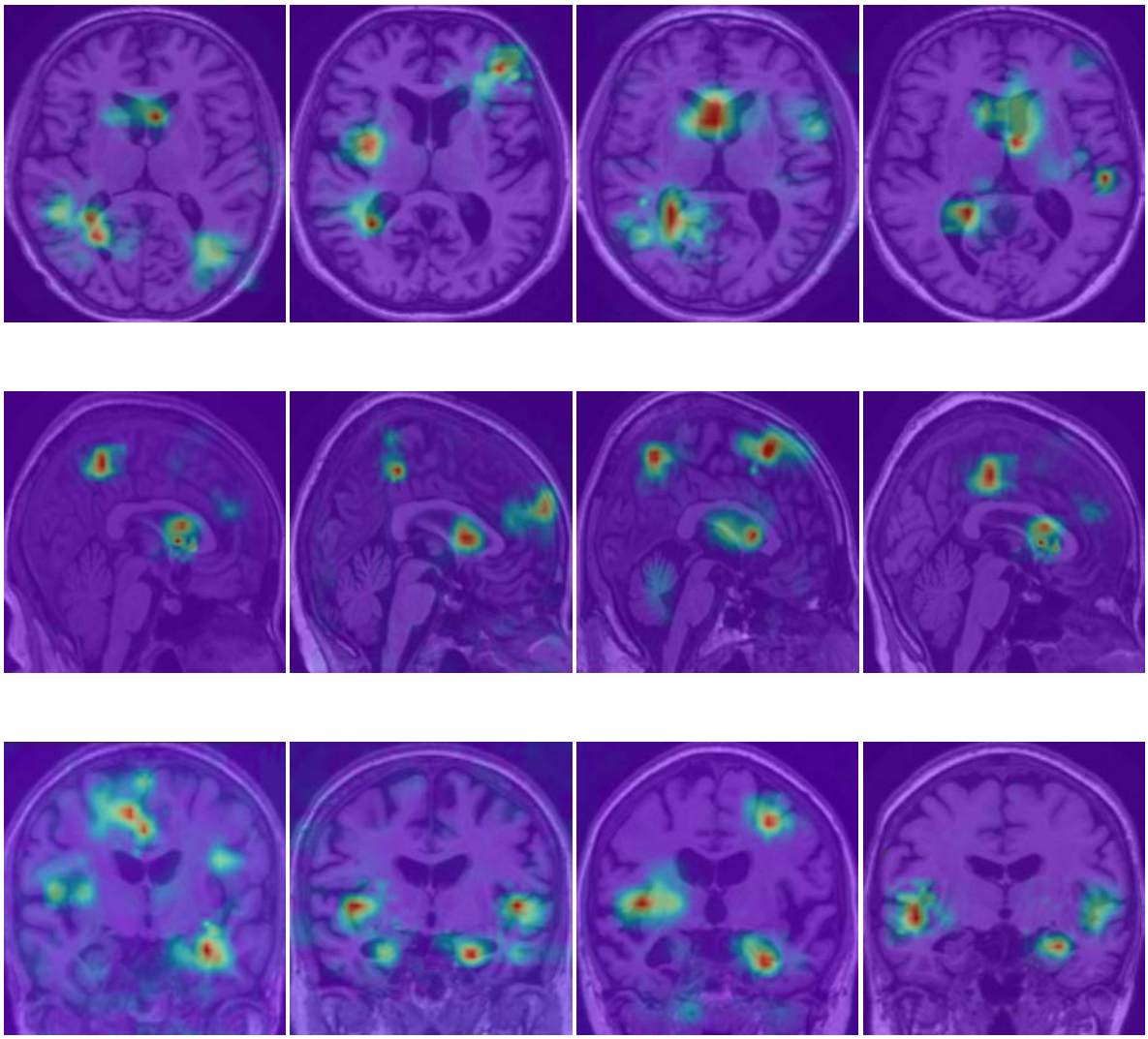


Figure 3.9: Given the m_k^{T}, the explainable tool provides HAM-generated heatmaps and three reference samples $R_c, c \in [1, 2, 3]$ whose latent features are most similar to m_k^{T}.

It is noted that the heatmaps of the reference samples are quite similar to the input's heatmaps. PCR highlights the atrophy of the cerebral cortex, the pathological abnormalities of the hippocampus, and the enlargement of the lateral ventricle, which have salient features related to the model decision. This confirms PCR's ability to retrieve samples with similar disease areas.

3.6.5 Ablation Study

We evaluate the contribution of each constituent module in our proposed MAXNet and HAM method on the subset of ADNI data, i.e., $\mathcal{L}_{\mathrm{cls}} + \mathcal{L}_{\mathrm{ctr}}$, DAM and MFM by removing at least one module at a time. The detailed results are shown in Table 3.5. Several observations are made. (1) If we eliminate the loss $\mathcal{L}_{\mathrm{cls}} + \mathcal{L}_{\mathrm{ctr}}$, then the model

Table 3.5: Contributions of individual modules in the proposed MAXNet on subset 1. Values indicating mask collapse are blank.

$\mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{ctr}}$	DAM	MFM	ACC	Insertion	Deletion
✓			0.725±0.003	0.445	0.398
	✓		0.752±0.075	0.546	0.446
		✓	0.763±0.008	0.327	0.338
✓	✓		0.899±0.140	0.762	0.636
	✓	✓	0.856±0.021	0.526	0.553
✓		✓	0.916±0.012	0.678	0.703
✓	✓	✓	**0.953±0.002**	**0.801**	**0.263**

actually adapts to make predictions or produce heatmaps primarily based on the last convolution features, which fails to achieve high classification accuracy as MRI volumes are difficult to classify using coarse feature maps; (2) If we abandon the DAM, each sample can only be learned with single-resolution activation features stemming from the high-scale level. Consequently, there is a drastic drop on both average accuracy and insertion score; (3) Finally, if we discard the MFM and concatenate all the multi-resolution feature maps, the model is trapped into local optima and cannot be further improved.

3.6.6 Discussions

There are mainly two factors which may limit the performance of our work. First, since the ground-truth annotations of pathological abnormalities such as the bounding boxes in the brain regions are not publicly available, our model learns the class discriminative latent features in a weakly supervised manner (i.e., under the supervision of image-level class labels), which leads to inaccuracy in the identification of different types of pathologies. Second, the proposed model does not incorporate any medical domain knowledge as inputs. As a result, our model may not learn features that exactly match prior knowledge from relevant professionals such as doctors and clinicians.

3.7 Conclusion

In this paper, we integrated several novel modules to constitute a novel explainable framework by employing 3D deep learning techniques. A novel explainable network dubbed MAXNet was proposed to classify AD. Among which, we introduced the

DAM and MFM blocks to aggregate multi-resolution feature activation maps into the latent space, which was not only representative for high-resolution explanations but also crucial for model predictions. Besides, the proposed cluster and contrastive losses encouraged the model to learn interpretable features w.r.t. target labels in the latent space. Additionally, we provided an explainable tool which comprises HAM to generate voxel-wise information, and the PCR module to provide similar samples as the prediction basis. By comparing the proposed model to other state-of-the-art methods through extensive experiments, we validated the effectiveness of our model with good diagnostic accuracy. Moreover, the model was capable of providing insightful explanations about its decisions. Both factors are conducive to applying deep learning models to clinical applications.

Despite the encouraging performance gained by our work, it suffers two limitations. Firstly, the proposed model does not incorporate medical domain knowledge. For future studies, the model should potentially be further improved if prior domain knowledge from medical professionals is integrated. Secondly, the latent features are learned by our model in a weakly supervised manner due to the lack of publicly available annotations of pathological abnormalities, which may neglect possible pathological locations in the brain. Therefore, we plan to develop approaches to learn and integrate expert knowledge in our future work.

Chapter 4

Explainable Vision Transformer with Pixel-wise Attention

In this chapter, we propose the explainable vision transformer model for achieving accuracy-explainability trade-off. An effective way to discover robust interpretable features and perform the prediction accurately is a pivotal step for building an intrinsically interpretable model. We obtain such interpretable representations by explainable attention weights that are able to learn semantically interpretable representations from tokens in terms of model decisions with noise robustness. We also propose a self-supervised attribute-guided loss for our architecture, which utilizes both the attribute discriminability mechanism and the attribute diversity mechanism to enhance the quality of learned representations.

4.1 Introduction

Over the last few years, transformer models have attracted increasing attention with encouraging results in a multitude of challenging domains, such as natural language processing, vision, or graphs [108]. The Multi-Head Attention (MHA) and Multi-Layer Perceptron (MLP) modules in transformers effectively model global representations without convolution [109]. The effectiveness of this framework lies in its ability to capture long-range dependencies. Despite their excellent performance, most transformer architectures are usually expressed as black boxes [110]. Specifically, the large number of parameters and complex interactions between modules make it challenging to provide explanations for the model predictions. Given the high applicability of transformers in high-risk decision-making domains, such as healthcare and autonomous driving, there is a strong necessity for gaining insights into the model's decision-making process [111]. An interpretable solution is able to aid in debugging the models and identifying crucial features for downstream tasks.

XAI is an emerging sub-field of AI pursuing to capture the properties that have influence over the decision of a model [35]. Depending on the phases where predictions

and explanations are performed, these methods can be categorized into two types: intrinsically explainable models and post-hoc explanation methods. Several previous studies have pointed out that explainable models outperform post-hoc methods in faithfulness and stability [112]. Unfortunately, little work has been done so far in the field of explainable transformers. In order to leverage advantages of explainability, recent research efforts have been made to explore the possibility of building inherently explainable transformers [113]. However, the explicit expressive features were not explored to obtain faithful explanations w.r.t. model decisions.

Recently, transformers have shown promising results in weakly supervised semantic segmentation (WSSS) tasks [114]. The generation of pixel-level pseudo segmentation ground-truth labels based on image-level labels is a pivotal step for this task. Transformers employ MHA and MLP to effectively capture long-range semantic correlations, which play a critical role in localizing the target object. Despite the fact that different attention heads in the transformer can attend to diverse semantic areas of an image, it is still unclear how to correctly align these features with a particular semantic class [110]. One common issue among existing transformer-based works is the utilization of a token for each class, which often highlight the most discriminative region of an object instead of the entire object region [108].

In this chapter, we propose the so-called eXplainable Vision Transformer (eX-ViT) with the inherent attribute of explainability and high performance for WSSS tasks. Specifically, the eX-ViT comprises the Explainable Multi-Head Attention (E-MHA) module, which can inherently provide interpretable attention maps that align with informative input patterns with noise robustness. Furthermore, the Attribute-guided Explainer (AttE) module is integrated into the eX-ViT, to learn discriminative attribute features for the target object. Intuitively, we assume each object is made up of several attributes, which could be basic elements including color, shape, and texture, or higher-level local features such as body parts. Our key idea is to decompose the feature representation into a set of learnable attribute features for the target object, capable of capturing diverse and discriminative object features. Besides, a novel attribute-guided loss is designed to promote the learning process inside AttE in a self-supervised manner. More precisely, this loss implicitly adds the regularization to force the representations to focus on various attributes of each target class through the attribute discriminability mechanism and attribute diversity mechanism. We then verify and evaluate our method on several WSSS tasks. To the best of our knowledge, this is the first work to develop an intrinsically explainable vision transformer for WSSS tasks.

The remainder of the chapter is organized as follows. Section 4.2 briefly describes some recent related works on vision transformers, XAI techniques for transformers, and weakly supervised semantic segmentation methodologies. Section 4.3 presents the

explainable architecture, i.e., eX-ViT, and introduces its main modules. Experimental results and discussions are presented in Section 4.4, followed by concluding remarks drawn in Section 4.5.

4.2 Related work

4.2.1 Transformers for Vision

Transformer-based models have recently been introduced to vision tasks and achieved remarkable progress. One of purely transformer-based models is the ViT [109], which has exhibited impressive performance without convolution. However, the ViT is inferior to CNNs when capturing local details. DeiT [115] addressed this issue by employing a strong image classifier as the teacher model to train data-efficient transformer models. Li *et al.* [114] designed the TransCAM, which explicitly utilizes the attention weights produced from the transformer to refine CAM results. Moreover, there are some research studies with modified ViT architectures that benefit downstream vision tasks such as semantic segmentation. However, most of the existing designs focus on efficient and effective frameworks for downstream tasks without considering interpretability. Thus these methods tend to be less faithful to the users. Recently, Peng *et al.* [116] proposed the Conformer to aggregate both the convolutional operations and self-attention mechanisms into a unified framework. However, Conformer results in a more complicated design and additional computational cost. Xu *et al.* [108] added extra class tokens and enforced them learning the activation maps of different classes, it has limited ability to encode more information when it comes to a larger data set, e.g., COCO [117]. In this chapter, we aim to address these issues by proposing the so-called eX-ViT, which exploits explainable features that are robust to noise and provides faithful explanations.

4.2.2 XAI for Transformers

There are mainly two sub-fields of explainable techniques: intrinsically explainable models and post-hoc explanation methods. Unlike post-hoc models, intrinsically explainable models aim to directly incorporate interpretability in the structure of the models, thus revealing the intrinsic reasoning process of the models [112]. Several previous studies have pointed out that explainable models outperform post-hoc methods in faithfulness and stability [112]. Unfortunately, little work has been done so far in the field of explainable transformers. Caron *et al.* [113] utilized a self-supervised approach called DINO based on vision transformers and concluded that the attention

maps contain features about the semantic information of the image. But the expressive features were not explored to obtain faithful explanations. Different from the majority of previous studies, we attempt to build the first explainable transformer architecture with the objective of learning interpretable features.

In terms of post-hoc explanation approaches, there are a variety of recent studies that explore the explainability for transformers. Chefer *et al.* [112] proposed a layer-wise relevance propagation (LRP) method by introducing a relevancy propagation rule that is applicable to both positive and negative contributions. This approach, however, is not able to provide the interpretation for attention modules besides self-attention. Abnar *et al.* [118] proposed to combine the attention scores across multiple layers, but this method fails to distinguish between positive and negative attributions. Recently, Chefer *et al.* [52] also proposed a generic approach to explain transformers including bi-modal ones. However, most of the existing post-hoc methods tend to be fragile, sensitive, and less faithful. Since they cannot faithfully uncover the decision making process of the trained models, and the explanations can be easily impaired by different input schemes (e.g., perturbations or transformations).

4.2.3 Weakly Supervised Semantic Segmentation

Compared to supervised learning methods, WSSS aims at training models with weak labels such as bounding boxes and image-level labels. As the cornerstone of WSSS, The Class Activation Mapping (CAM) technique is widely used in the design of WSSS tasks to extract object localization maps and approximate the segmentation mask [21]. Despite the encouraging results, CAM suffers from the issue of incomplete object activation [108]. To address this drawback, several approaches have been proposed as the CAM expansions to remove the most discriminative parts of CAM and discover more complete object localization maps. Chen *et al.* [119] designed the ReCAM, which a method that leverages CAM to extract pixels belonging to specific classes and subsequently incorporates them into a fully-connected layer along with the corresponding class label for further learning. Yuan *et al.* [120] proposed the multi-strategy contrastive learning framework to discover the similarity and dissimilarity of contrastive sample pairs. Lee *et al.* [121] learned pixel-level feedback by use of saliency map generated from the off-the-shelf detection model. Chen *et al.* [111] introduced several image-specific prototype features for WSSS learning with favorable performance. The above methods are commonly based on CNNs, which reveals the inherent drawback of convolution. Xu *et al.* [108] introduced the transformer attention to learn class-specific localization maps. Ru *et al.* [110] adopted the semantic affinity in self-attentions in transformers to produce more integral pseudo labels for WSSS. However, there is still a

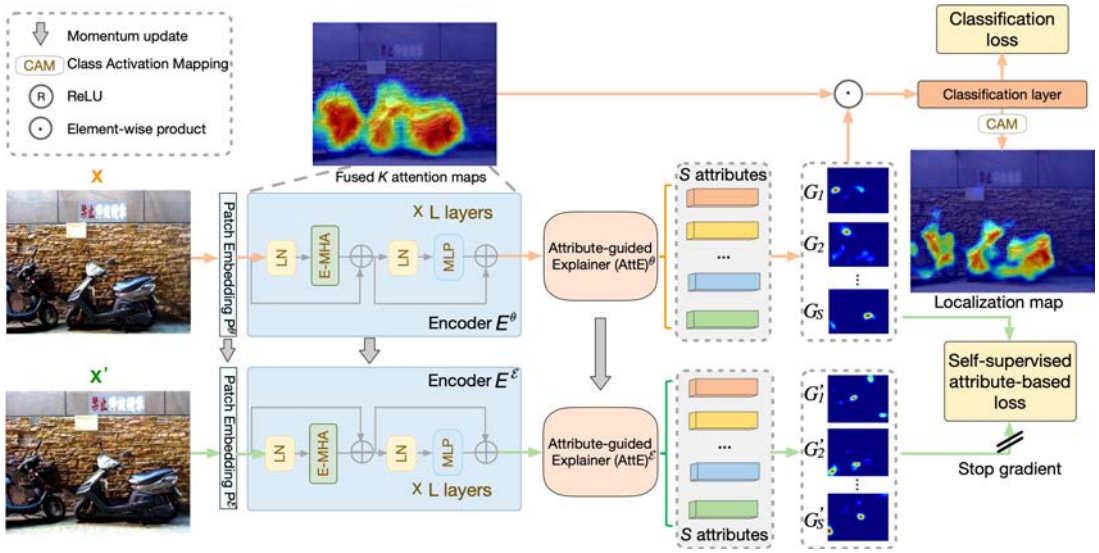


Figure 4.1: Illustration of the proposed eXplainable Vision Transformer (eX-ViT) architecture. x and $x^{'}$ are two different random transformations of an input image. We use a transformer backbone as the encoder to extract feature maps, the backbone contains consecutive L encoding layers with Explainable Multi-Head Attention (E-MHA) as the attention block. θ is the trainable module, while $\mathcal{E}$ is an exponential moving average of θ. The Attribute-guided Explainer (AttE) is proposed atop the encoder to decompose the attention maps into features of attributes through diverse attribute discovery, to facilitate the generation of more faithful and robust interpretations. We also design a self-supervised attribute-guided loss function for our eX-ViT, which aims at learning robust semantic representations via the attribute diversity mechanism and attribute discriminability mechanism.

large gap between fully supervised semantic segmentation and previous transformer-based WSSS methods. In our work, we propose a transformer-based model to extract explainable features to localize class-specific feature maps. We attempt to build a novel transformer architecture with the objective of learning interpretable representations in a self-supervised manner to narrow the supervision gap.

4.3 Method

This section details our proposed network architecture, i.e., the eX-ViT. First, we introduce the overall architecture. Second, we describe the intuition and design of the E-MHA and discuss several important properties of the E-MHA. Furthermore, The AttE is proposed to integrate into our eX-ViT to decompose the attention maps into features of attributes through diverse attribute discovery, and a self-supervised attribute-guided loss is adopted to learn robust semantic representations via the attribute diversity mechanism and attribute discriminability mechanism, which constitutes faithful evidence for model predictions.

4.3.1 Architecture of the eX-ViT

The overall architecture of our proposed eX-ViT is depicted in Fig. 4.1. In particular, the eX-ViT is a siamese network, which comprises two branches for a pair of input images (two data augmentations from an original input) to learn interpretable attention maps in a self-supervised manner. Each branch comprises a transformer encoder with L transformer layers consisting of the novel Explainable Multi-Head Attention (E-MHA) module, and the Attribute-guided Explainer (AttE) module atop the encoder. Specifically, the parameters $\mathcal{E}$ in the lower branch use a momentum update with the upper θ. Empirically, the proposed architecture can conveniently replace the backbone networks in existing methods for WSSS tasks.

4.3.2 Explainable Multi-Head Attention (E-MHA)

In this section, we introduce our novel Explainable Multi-Head Attention (E-MHA) module as shown in Fig. 4.2, which consists of H parallel heads. Specifically, given an input feature map $\boldsymbol{X} \in \mathbb{R}^{T \times d}$ where T is the spatial size and d is the feature dimension, each head H_h holds an explainable attention weight $\boldsymbol{A}_h \in \mathbb{R}^{N \times d}$ (N is the spatial size of $\boldsymbol{A}_h$.) that represents the relative importance of input features. That is, $\boldsymbol{A}_h$ aims to learn explainable features for the output through the proposed E-MHA module.

In particular, we structure this section around two crucial attributes of the E-MHA module: **Noise robustness:** The E-MHA is computed as a dynamic alignment between the input tokens and the attention weight. When the module is optimized, the attention weight is driven to focus on the most discriminative and class-related patterns from the input tokens. Instead of directly removing the irrelevant noises from the input image, we adopt a dynamic alignment mechanism in E-MHA to extract discriminative features from the input, thus reducing the noise information gradually and then preserving the key input patterns.

Inherent explainability: Given input $\boldsymbol{X}$, the E-MHA aims to learn the attention weight that maximizes the alignment between input tokens and the attention weight. During the optimization process, maximizing this alignment means encoding the projected input values as eigenvectors of the attention weight. As a result of this property, the model-inherent attention weight is learned with the discriminative input patterns and thus directly used to explain model decisions without needing any external tools.

First, given input $\boldsymbol{X}$, the projected key, query and value are obtained as follows

$$\boldsymbol{Q} = \boldsymbol{X}\boldsymbol{W}^{Q}, \quad \boldsymbol{K} = \boldsymbol{X}\boldsymbol{W}^{K}, \quad \boldsymbol{V} = \boldsymbol{X}\boldsymbol{W}^{V}, \tag{4.1}$$

where $\boldsymbol{W}^{Q} \in \mathbb{R}^{d \times d}$, $\boldsymbol{W}^{K} \in \mathbb{R}^{d \times d}$, and $\boldsymbol{W}^{V} \in \mathbb{R}^{d \times d}$ are trainable transform matrices.

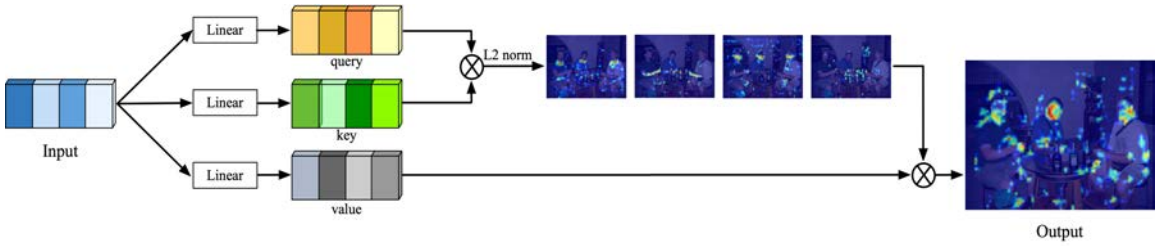


Figure 4.2: The architecture of Explainable Multi-Head Attention (E-MHA). We use $\otimes$ to denote matrix multiplication.

Second, the self-attention operation is constructed by

$$\boldsymbol{W} = \frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d}}, \tag{4.2}$$

where the obtained matrix $\boldsymbol{W}$ implies how much attention is paid on each token.

Third, the attention weight $\boldsymbol{A}$ is defined as follows

$$\boldsymbol{A} = f_\theta(\boldsymbol{W} + \boldsymbol{b})^{\mathrm{T}}, \tag{4.3}$$

where $\boldsymbol{b}$ is a trainable bias term, which is introduced as an initial alignment for the input patterns. $f_\theta(\cdot)$ is a non-linear function that scales the L2 norm of its input, i.e., $f_\theta(\boldsymbol{x}) = \frac{\boldsymbol{x}}{||\boldsymbol{x}||_2}$ and $||f_\theta(\boldsymbol{x})|| \leq 1$. In our case, L2 norm is applied to the vector of $(\boldsymbol{W} + \boldsymbol{b})$.

In what follows, the self-attention feature $\boldsymbol{S}$ is formally expressed as

$$\boldsymbol{S} = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{V}, \tag{4.4}$$

According to Eq. 4.3, $||\boldsymbol{A}|| \leq 1$. Therefore $\boldsymbol{S}$ in Eq. 4.4 is upper-bounded as follows

$$\boldsymbol{S} = ||\boldsymbol{A}||\,||\boldsymbol{V}||\cos(\boldsymbol{A}, \boldsymbol{V}) \leq ||\boldsymbol{V}||. \tag{4.5}$$

Where both $\boldsymbol{A}$ and $\boldsymbol{V}$ are reshaped to a row-wise feature vector before applying the L2 norm function $||\cdot||$. When Eq. 4.5 is optimized, the attention weight $\boldsymbol{A}$ is proportional to $\boldsymbol{V}$. In order to achieve maximal output, $\boldsymbol{A}$ is driven to align with the discriminative features in $\boldsymbol{V}$, instead of the uninformative noise. Therefore, $\boldsymbol{S}$ can only achieve this upper bound if all possible solutions of $\boldsymbol{v} \in \boldsymbol{V}$ are encoded as eigenvectors in the weight $\boldsymbol{A}$. This maximization suggests with the attention weight $\boldsymbol{A}$, we will obtain an inherently explainable decomposition of input patterns.

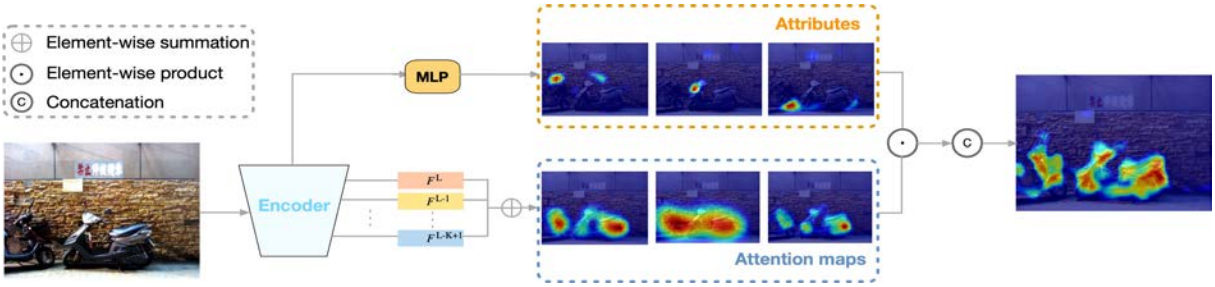


Figure 4.3: Illustration of Attribute-guided Explainer (AttE). We aggregate the interpretable attention maps from the last K transformer layers to generate a fused attention map with good precision on the complete object context information. The attribute features are regarded as the complement information to better guide the localization of the object context, thus producing robust attribute features in a weakly supervised manner.

Overall, the computation in layer l is expressed as

$$\begin{aligned} \boldsymbol{S}^l &= \mathrm{E-MHA}(\mathrm{LN}(\boldsymbol{F}^{l-1})), \\ \boldsymbol{Z}^l &= \boldsymbol{S}^l + \boldsymbol{F}^{l-1}, \\ \boldsymbol{F}^l &= \mathrm{MLP}(\mathrm{LN}(\boldsymbol{Z}^l)) + \boldsymbol{Z}^l, \end{aligned} \tag{4.6}$$

where $\mathrm{LN}(\cdot)$ is the LayerNorm layer, $\mathrm{MLP}(\cdot)$ denotes the multi-layer perceptron layer, and $\boldsymbol{F}^l$ is the output of layer l.

Our key motivation of E-MHA is to dynamically align its attention weights with the discriminative patterns from input values while reducing the impact of noise. The cascade transformer layers in the encoder enable the model to suppress the noise information gradually and learn discriminative input patterns. As a result, the model is able to discover robust representations from the input image. With the attributes of noise robustness and inherent explainability, E-MHA produces the transformer attention map which inherently provides an explainable combination of contributions from discriminative input patterns w.r.t. the model predictions.

4.3.3 Attribute-guided Explainer (AttE)

Although the proposed E-MHA provides the intuitive process for explainable feature learning, it is non-trivial to obtain intrinsically interpretable representations that benefit the WSSS tasks without additional regularization. Inspired by the pixel-wise prediction scheme used in semantic segmentation frameworks to localize objects, we propose the Attribute-guided Explainer (AttE) module for our eX-ViT with the objective of decomposing the attention map into attribute features based on the diverse attribute discovery. By which, the learned feature maps can be viewed as a set of attributes at a granular level that capture more complete object information.

Given that the transformer structure tends to learn more uniform representations across all layers, we propose to utilize the transformer attention maps from the last layer in eX-ViT's encoder, to learn a set of trainable attribute features. Concretely, to model the context attention, given the feature map $\boldsymbol{F}^L \in \mathbb{R}^{H\times W\times d}$ produced by the encoder E^θ, we first calculate a set of spatial feature maps that capture the relative importance of all HW locations as follows

$$\boldsymbol{C}_{i,j} = f_\phi(\boldsymbol{F}^L), \quad \forall\{i,j\} \in H \times W, \tag{4.7}$$

where $f_\phi(\cdot)$ is implemented by a 2-layer MLP block, with one hidden layer followed by a LN layer and the ReLU activation layer. $\boldsymbol{C}_{i,j} \in \mathbb{R}^{H\times W\times c}$ is the obtained feature map with the channel dimension c. We will investigate the influence of c on the model performance in Section 4.4.3.

Furthermore, we apply the ℓ_2-norm function to $\boldsymbol{C}_{i,j}$ along the channel dimension, which is formally expressed as

$$\overline{\boldsymbol{C}}_{i,j} = \frac{\boldsymbol{C}_{i,j}}{||\boldsymbol{C}_{i,j}||_2}, \tag{4.8}$$

where $||\cdot||_2$ denotes the L2 norm, $\overline{\boldsymbol{C}}_{i,j}$ is the normalized representation indicating which spatial features to emphasize or suppress.

Subsequently, $\overline{\boldsymbol{C}}$ is sliced into S groups on the channel dimension, i.e., ($\overline{\boldsymbol{C}}_1$, $\overline{\boldsymbol{C}}_2$,..., $\overline{\boldsymbol{C}}_S$), where $\overline{\boldsymbol{C}}_s \in \mathbb{R}^{H\times W\times \frac{c}{S}}$ stands for the feature map of the s-th attribute, S is the total number of attributes. To this end, we can apply $\overline{\boldsymbol{C}}_s$ of attribute s to the feature $\boldsymbol{F}^L$ by

$$\boldsymbol{G}_s = \overline{\boldsymbol{C}}_s \odot \boldsymbol{F}^L, \tag{4.9}$$

where $\odot$ is the element-wise product, and the $\overline{\boldsymbol{C}}_s$ is broadcast along the channel dimension to match the shape of $\boldsymbol{F}^L$. $\boldsymbol{G} = [\boldsymbol{G}_1, \boldsymbol{G}_2, ..., \boldsymbol{G}_S]$ is the final output that is concatenated along the channel dimension. By this means, each feature map $\boldsymbol{F}^L$ is projected into S attribute representations that explicitly reveal which pixels are related to the attribute s. Likewise, we follow the same procedure described from Eqs. 4.7 - 4.9, the attribute representation $\boldsymbol{G}^{'}$ of the second augmented input can be generated accordingly with the momentum encoder $E^{\mathcal{E}}$. And our $AttE^{\mathcal{E}}$ is also the exponential moving average of the trained $AttE^\theta$.

In summary, the output of AttE can be seen as the decomposed contributions for individual attributes. By this means, our model is able to encode semantically explainable features for the target object in an explicit manner, which facilitates the learning of complete object context information. Moreover, we elaborately design our attribute-guided loss function to guide the learning of AttE, which will be presented in next subsection.

4.3.4 Attribute-guided Loss Function

A challenging problem for typical vision transformers is that they are not intrinsically interpretable due to lack of the representational power. In our work, we propose to improve model interpretability by regularizing its representations with the attribute-guided loss function, i.e., the global-level attribute-guided loss $\mathcal{L}_{\text{global}}$, the local-level attribute discriminability loss $\mathcal{L}_{\text{dis}}$ loss and the attribute diversity loss $\mathcal{L}_{\text{div}}$. On one hand, the $\mathcal{L}_{\text{global}}$ encourages the predicted attribute features to approximate the target object, which ensures the faithfulness of the global representations. On the other hand, the $\mathcal{L}_{\text{dis}}$ and $\mathcal{L}_{div}$ aim to localize fine-grained attributes through the attribute discriminability mechanism and attribute diversity mechanism, thus enabling the robust feature learning.

Since higher layers discover high-level concepts like objects or scenes, we propose to fuse transformer attention maps from the last K encoder layers to achieve good accuracy on the complete object context information. Hence, given the obtained feature map $\boldsymbol{F}^l$ in l-th encoder layer, the fused attention map is expressed as

$$\hat{\boldsymbol{F}} = \frac{1}{K}\sum_{l}^{K}\boldsymbol{F}^l, \tag{4.10}$$

where $\hat{\boldsymbol{F}}$ is the fused transformer attention map. By this means, we aggregate cascaded encoder blocks to produce a reliable attention map for complete object localization. As the aggregated attention map $\hat{\boldsymbol{F}}$ is attribute-agnostic, we propose to couple it with the attribute features $\boldsymbol{G}$ to generate the attribute-guided attention map. The process is defined as follows

$$\boldsymbol{M} = \hat{\boldsymbol{F}} \odot \boldsymbol{G}, \tag{4.11}$$

where $\boldsymbol{M}$ represents the final output of the attribute-guided feature map.

Based upon $\boldsymbol{M}$, the global-level attribute-guided loss $\mathcal{L}_{\text{global}}$ is computed by the multi-label soft margin loss

$$\mathcal{L}_{\text{global}} = \frac{1}{C}\sum_{c=1}^{C}(y^c \log(\hat{y}_c) + (1-y^c)\log(1-\hat{y}_c)), \tag{4.12}$$

where the prediction $\hat{y}_c$ is obtained by feeding the feature map $\boldsymbol{M}$ into a classification layer followed by a generalized mean pooling operation. By optimizing the $\mathcal{L}_{\text{global}}$, the interpretable features are gathered as a summation of the important scores of all attribute features, which ensures the faithfulness of the explanations.

In addition, to improve the ability of network for learning diverse and discriminative attribute representations for the target object, we propose the local-level attribute-guided loss through the attribute discriminability mechanism and attribute diversity mechanism in a self-supervised manner. Intuitively, the attribute discriminability mechanism aims to make attribute features consistently discriminative between two types of input views, while the attribute diversity mechanism enables the model to learn the effective decomposition with the attribute diversity. Formally, the attribute discriminability loss $\mathcal{L}_{\text{dis}}$ is defined by

$$\begin{aligned} \mathcal{L}_{\text{dis}} &= |d - \sum_{s=1}^{S} d^s|, \\ d &= \ell(g(\boldsymbol{G}), g(\boldsymbol{G}^{'})), \\ d_s &= \ell(g(\boldsymbol{G}_s), g(\boldsymbol{G}_s^{'})), \end{aligned} \tag{4.13}$$

where $g(\cdot)$ is the generalized mean pooling. And we adopt the normalized Mean Square Error as the $\ell(\cdot)$ function to calculate the distance between two features. As can be seen from Eq. 4.13, d is leveraged to minimize the difference between attribute features, while d_s is used to guarantee the consistency between $\boldsymbol{G}$ and $\boldsymbol{G}^{'}$ for each individual attribute. Empirically, this attribute discriminability loss function $\mathcal{L}_{\text{dis}}$ is able to facilitate the model to discover discriminative class-specific attributes and obtain more comprehensive localization maps. Meanwhile, we introduce the attribute diversity loss $\mathcal{L}_{\text{div}}$ is formally defined by

$$\mathcal{L}_{\text{div}} = \frac{1}{S(S-1)} \sum_{i=1,j=1}^{S} \sum_{i \neq j}^{S} \frac{< \boldsymbol{G}_i, \boldsymbol{G}_j >}{||\boldsymbol{G}_i||_2 ||\boldsymbol{G}_j||_2}, \tag{4.14}$$

The intuition behind the $\mathcal{L}_{\text{div}}$ is to make attribute features to the maximally independent from each other, so as to make attribute features focus on different discriminative object regions.

Overall, the loss function for the proposed eX-ViT is given below

$$\mathcal{L} = \mathcal{L}_{\text{global}} + \alpha \mathcal{L}_{\text{dis}} + \beta \mathcal{L}_{\text{div}}, \tag{4.15}$$

where $\mathcal{L}_{\text{global}}$ is the multi-label soft margin loss. α and β are the coefficient of $\mathcal{L}_{\text{dis}}$ and $\mathcal{L}_{\text{div}}$, respectively.

As a result, our attribute-guided loss promotes the learning of attribute features. The global-level loss $\mathcal{L}_{\text{global}}$ ensures a faithful transformer model, while the $\mathcal{L}_{\text{dis}}$ and $\mathcal{L}_{\text{div}}$ enable discriminative and robust attribute features. The effectiveness of the loss function is further verified in the experimental section.

4.4 Experimental Results

In this section, we first introduce the experimental settings including datasets and implementation details. Second, we evaluate the efficiency of our proposed eX-ViT and compare it with the recent state-of-the-art methods. Third, we conduct a series of ablation studies to discover the performance contribution from different modules in our framework.

4.4.1 Setup

Datasets

We conduct experiments on PASCAL VOC 2012 dataset [122] and MS COCO 2014 dataset [117]. PASCAL VOC 2012 dataset includes 20 object classes and one background class for the semantic segmentation task. Following the common experimental configuration from others, we adopt the augmented dataset which contains three subsets, training, validation, and testing sets, each having 10582, 1449, and 1464 images, respectively. MS COCO 2014 dataset uses 81 classes, its training and validation sets have 82081 images and 40137 images respectively. Note that image-level labels are only used during training and ground-truth bounding box annotations are solely used during the inference time. In line with previous works [110], we report the mean Intersection-over-Union (mIoU) to evaluate the performance of our proposed model.

Implementation Details

We use PyTorch for implementation and conduct experiments. The encoder parameters are pre-trained on ImageNet. During training, we use the AdamW optimizer. For the transformer encoder E^{θ}, the initial learning rate is set to be 5×10^{-5}, which is further decayed via a polynomial schedule. The learning rate for the rest of the parameters is 5×10^{-4}. For the training on the PASCAL VOC 2012 dataset, the batch size is set as 16, and the training process lasts 40k iterations. On MS COCO 2014 dataset, we trained the models for 80k iterations with a batch size of 8. For data augmentation, we used random scaling with a range of [0.5,2.0], random horizontally flipping, and random cropping.

The default hyper-parameters are set as follows. For encoders E^{θ} and $E^{\mathcal{E}}$, it contains 12 layers, 6 heads within each E-MHA, and the hidden dimension is set to 384. Empirically, we set α and β in Eq. 4.15 as 0.5 and 1.0 respectively throughout this chapter. In line with previous works, we use the ResNet38 [123] as the backbone for semantic segmentation. At test time, only the branch with encoder E^{θ} is needed. Following the

Table 4.1: mIoU (%) of localization maps on the PASCAL VOC 2012 training set.

Method	Local. Maps	+denseCRF
SCE [125]	50.9	55.3
SEAM [124]	55.4	56.8
EDAM [126]	52.8	58.2
AdvCAM [127]	55.6	62.1
ECS-Net [128]	56.6	58.6
CSE [129]	56.0	62.8
SIPE [111]	58.6	64.7
ReCAM [119]	56.6	-
(Ours) eX-ViT	**59.1**	**65.3**

common practice in prior studies [124], we use multi-scale testing and CRFs to obtain pseudo segmentation results.

4.4.2 Comparison with State-of-the-arts

Comparison on Localization Maps

We first evaluate the qualitative results of CAM in mIoU(%) on localization maps. Table 4.1 reports the results of our proposed method as well as other recent state-of-the-art approaches on the PASCAL VOC 2012 training set. As can be seen from the table, SIPE [111] achieves the state-of-the-art result with a mIoU of 58.6%. eX-ViT outperforms all compared methods in terms of both metrics. Concretely, the results show that our eX-ViT improves the mIoU to 59.1%. We also conduct experiments based on eX-ViT with denseCRF post-processing, and the gain becomes up to 65.3%. Fig. 4.4 shows visual comparisons of object localization maps on the PASCAL VOC 2012 training set. As shown in Fig. 4.4, the fused class-specific attribute-guided localization map can effectively capture the discriminative features within the object context of the target class with more useful clues. As a result, the fused localization map by use of our eX-ViT brings notable visual improvements to produce complete and precise localization maps.

Comparison on Segmentation Results

The comparison results among the fully-supervised and weakly supervised state-of-the-art methods on PASCAL VOC 2012 validation and test sets are reported in Table 4.2. Among the compared methods, the eX-ViT is able to remarkably improve the segmentation performance using only image-level labels on the validation and test sets, respectively. As can be observed, compared to the fully-supervised methods, the

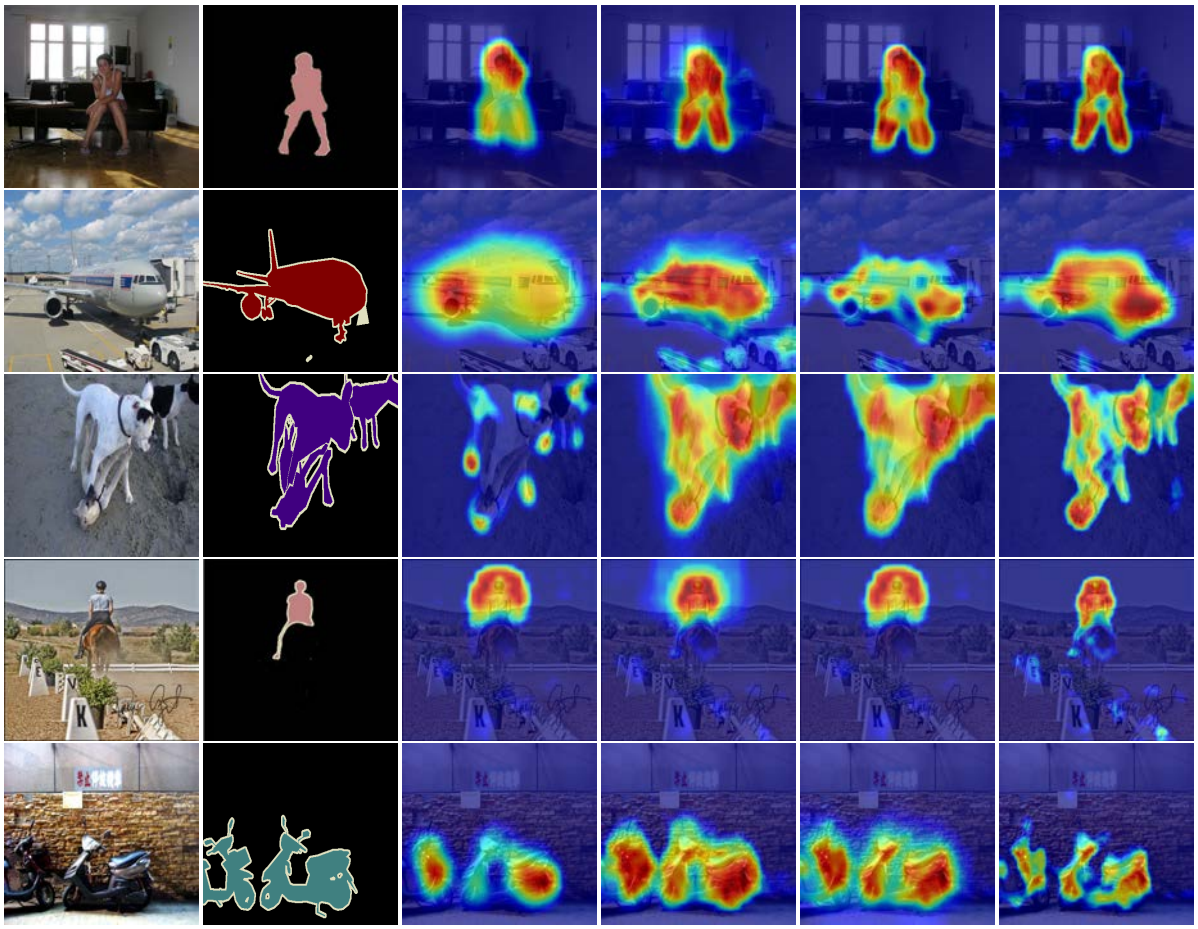


Figure 4.4: Visual comparison of localization maps generated by different methods on PASCAL VOC 2012 training set. From left to right: original image, ground-truth, CAM [21], SIPE [111], AdvCAM [127] and our eX-ViT.

eX-ViT is able to obtain comparable performance with 71.2% mIoU on the validation set and 71.1% mIoU on the test set. Compared with the recent state-of-the-art weakly supervised models, e.g., EPS [121] and EDAM [126] that use both additional saliency maps and image-level labels as supervision signals, eX-ViT still shows superior performance. The qualitative segmentation results on the validation set are shown in Fig. 4.5. Based on our model, DeepLabV2 can produce accurate and complete object segmentation results in various challenging scenarios, including different object scales and multiple objects.

Table 4.3 reports the semantic segmentation results on the MS COCO 2014 dataset. It is observed that methods with the supervision of saliency maps such as DSRG [141] and AuxSegNet [134] do not provide results comparable or superior to the WSSS methods with only image-level labels. The poor performance is caused by the limitation of saliency maps generated by pre-trained models. Instead, our method that leverages image-level labels achieves a segmentation mIoU of 42.9% with ResNet38 backbone,

Table 4.2: Performance comparison of various methods in mIoU (%) on the PASCAL VOC 2012 validation and test sets. *Sup.* indicates supervision type. $\mathcal{F}$: full supervision; $\mathcal{I}$: image-level labels; $\mathcal{S}$: saliency maps.

Method	*Sup.*	Backbone	validation	test
Fully-supervised methods				
DeepLab [130]	$\mathcal{F}$	ResNet101	77.6	79.7
WideResNet38 [123]	$\mathcal{F}$	WR38	80.8	82.5
Segformer [131]	$\mathcal{F}$	MiT-B1	78.7	-
Weakly-supervised methods				
SEAM [124]	$\mathcal{I}$	ResNet38	64.5	65.7
RRM [132]	$\mathcal{I}$	ResNet101	66.3	66.5
CONTA [133]	$\mathcal{I}$	ResNet38	66.1	66.7
AuxSegNet [134]	$\mathcal{I}+\mathcal{S}$	ResNet38	69.0	68.6
EPS [121]	$\mathcal{I}+\mathcal{S}$	ResNet101	70.9	70.8
EDAM [126]	$\mathcal{I}+\mathcal{S}$	ResNet101	70.9	70.6
CDA [135]	$\mathcal{I}$	ResNet38	66.1	66.8
ECS-Net [128]	$\mathcal{I}$	ResNet38	66.6	67.6
CSE [129]	$\mathcal{I}$	ResNet38	68.4	68.2
AdvCAM [127]	$\mathcal{I}$	ResNet101	68.1	68.0
RIB [136]	$\mathcal{I}$	ResNet101	68.3	68.6
A2GNN [137]	$\mathcal{I}$	ResNet101	66.8	67.4
LIID [138]	$\mathcal{I}$	ResNet101	66.5	67.5
SIPE [111]	$\mathcal{I}$	ResNet101	68.8	69.7
ReCAM [119]	$\mathcal{I}$	ResNet101	68.5	68.4
Ru *et al.* [110]	$\mathcal{I}$	MiT-B1	66.0	66.3
MCTformer[108]	$\mathcal{I}$	ResNet38	71.9	71.6
Kho *et al.* [139]	$\mathcal{I}$	ResNet38	66.4	66.8
RRM-ResNet [140]	$\mathcal{I}$	ResNet101	69.3	69.2
MuSCLe [120]	$\mathcal{I}$	EfficientNet	66.6	68.8
TransCAM [114]	$\mathcal{I}$	ResNet38	69.3	69.6
(Ours) eX-ViT	$\mathcal{I}$	ResNet38	**71.2**	**71.1**

which surpasses most recent state-of-the-art WSSS methods including SEAM [124], CSE [129], and MCTformer [108] by a large margin. Several qualitative segmentation results are shown in Fig. 4.6. These results confirm the effectiveness of our model, which is consistent with our intuition. Specifically, our eX-ViT remarkably improves the overall performance with the indispensable block of E-MHA and the AttE module. Adding these modules explicitly encourages eX-ViT to gain insightful clues on the complete object scene, and boost the model efficiency in producing accurate and complete object boundaries. It is noted that both the RIB [136] and SIPE [111] outperform our proposed eX-ViT model on the COCO validation set. This is mainly because that vision Transformers are a relatively new model architecture for WSSS compared to their traditional CNNs counterparts. Therefore, ViTs still require further refinement

Table 4.3: Performance comparison of the state-of-the-art WSSS methods in mIoU (%) on the MS COCO 2014 validation set. *Sup.* indicates supervision type. $\mathcal{I}$: image-level labels; $\mathcal{S}$: saliency maps.

Method	*Sup.*	Backbone	mIoU (%)
CNN			
DSRG [141]	$\mathcal{I}+\mathcal{S}$	VGG16	26.0
AuxSegNet [134]	$\mathcal{I}+\mathcal{S}$	ResNet38	33.9
EPS [121]	$\mathcal{I}+\mathcal{S}$	ResNet101	35.7
CONTA [133]	$\mathcal{I}$	ResNet101	33.4
SEAM [124]	$\mathcal{I}$	ResNet38	31.9
CSE [129]	$\mathcal{I}$	ResNet38	36.4
CDA [135]	$\mathcal{I}$	ResNet38	33.2
ReCAM [119]	$\mathcal{I}$	ResNet101	39.4
SIPE[111]	$\mathcal{I}$	ResNet38	43.6
RIB [136]	$\mathcal{I}$	ResNet101	43.8
Transformer			
Ru *et al.* [110]	$\mathcal{I}$	MiT-B1	38.9
MCTformer [108]	$\mathcal{I}$	ResNet38	42.0
(Ours) eX-ViT	$\mathcal{I}$	ResNet38	**42.9**

and optimization to achieve the state-of-the-art performance. We hope that the eX-ViT's promising performance will inspire further research efforts to enhance ViTs' performance for WSSS tasks.

Comparison on Interpretability

To compare our method with other explainable methods, we also adopt two common metrics, i.e., average precision (AP) and average recall (AR). Which are commonly used in the literature to measure interpretability. We evaluate our method using the DeiT backbone [115] and conduct the weakly-supervised image segmentation experiments, which is in line with earlier work [52]. The quantitative results are shown in Table 4.4. We can see that our model clearly surpasses the ViT model which contains the raw attentions, it reveals that our MAXNet achieves an AP of 15.7%, and an AR of 22.3%. We also observe that the post-hoc interpretability methods such as Rollout [118], GradCAM [19], and partial LRP [142] do not obtain faithful results compared to the counterparts. Which is possibly caused due to the extensive noises introduced by gradients or propagation rules.

Fig. 4.7 shows three cases of visualization results along with their ground truth segmentation label maps. Compared to the original CAM without AttE, attention maps produced with our model perform well in precisely locating both small and large objects with more complete object boundaries. This verifies our intuition with the design

Figure 4.5: Qualitative segmentation results on the validation set of PASCAL VOC 2012. From left to right: original image, ground-truth, SIPE [111] and our eX-ViT.

of eX-ViT and suggests that our proposed model is effective on learning comprehensive features for complete target objects.

Analysis of Misclassified Examples

Fig. 4.8 shows two misclassfied examples along with the learned attributes. In the first row of Fig. 4.8, the object "tv" is misclassified to a similar category "laptop". The importance of the screen as a feature for a laptop could be the reason for this. The second row shows a more complicated example. We can observe while the attention map produced by our model captures most of details in the image, it is unable to distinguish class-specific features required to make accurate predictions for the target class, i.e.,

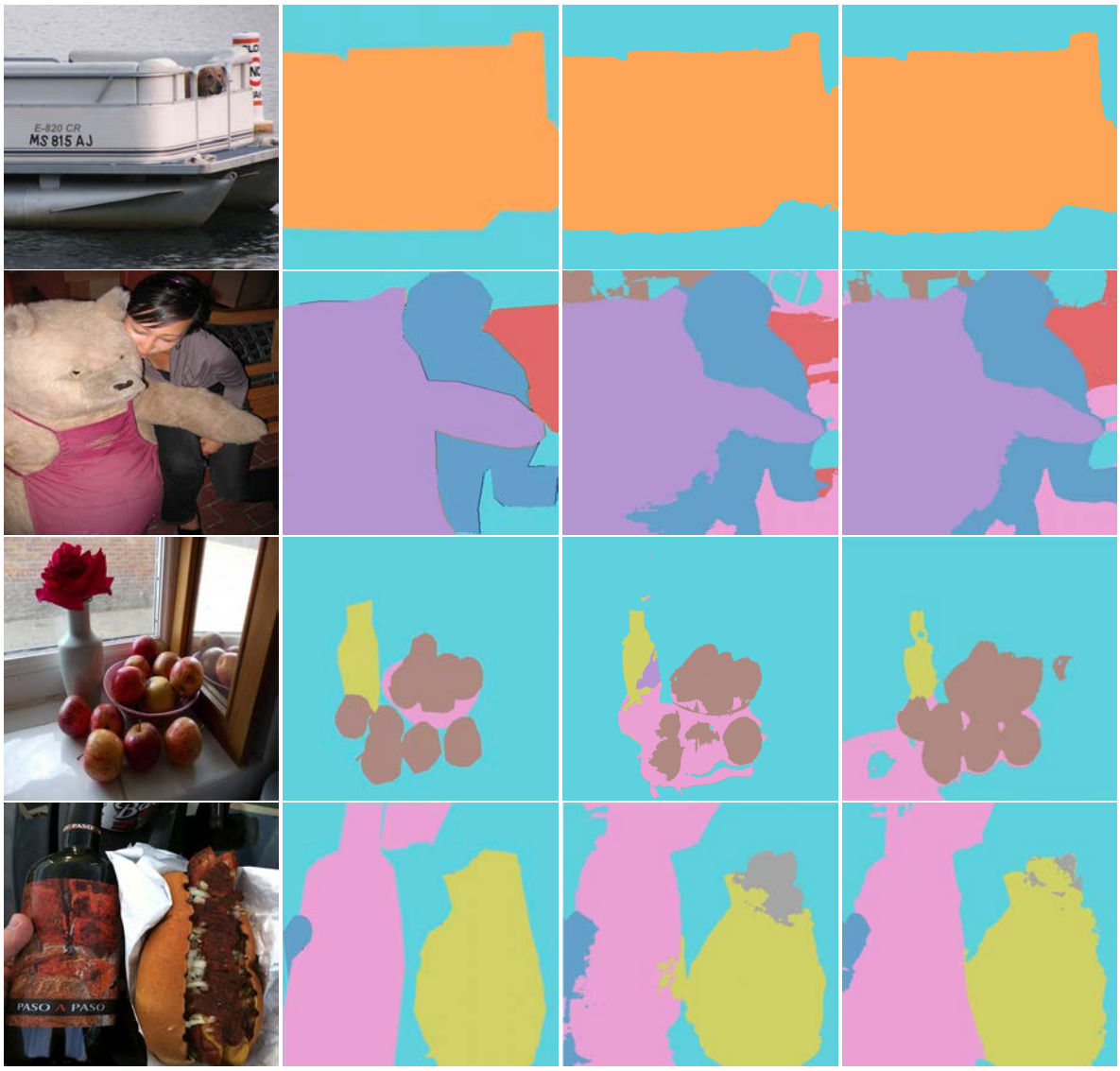


Figure 4.6: Qualitative segmentation results on the validation set of MS COCO 2014. From left to right: original image, ground-truth, SIPE [111] and our eX-ViT.

Table 4.4: Performance comparison of various methods on the MS COCO validation set.

Method	AP	AP_medium	AP_large	AR	AR_medium	AR_large
GradCAM [19]	2.3	2.3	4.7	5.5	5.9	10.7
Partial LRP [142]	4.7	8.0	5.1	10.4	19.9	8.0
ViT [109]	5.6	9.6	6.9	11.7	21.8	10.8
Rollout [118]	0.1	0.1	0.2	0.4	0.1	0.9
Trans. attribution [112]	7.2	10.4	12.4	13.4	21.0	19.4
Chefer et al. [52]	13.1	14.4	24.6	19.3	23.9	33.2
(Ours) eX-ViT	**15.7**	**15.3**	**26.5**	**22.3**	**24.3**	**36.1**

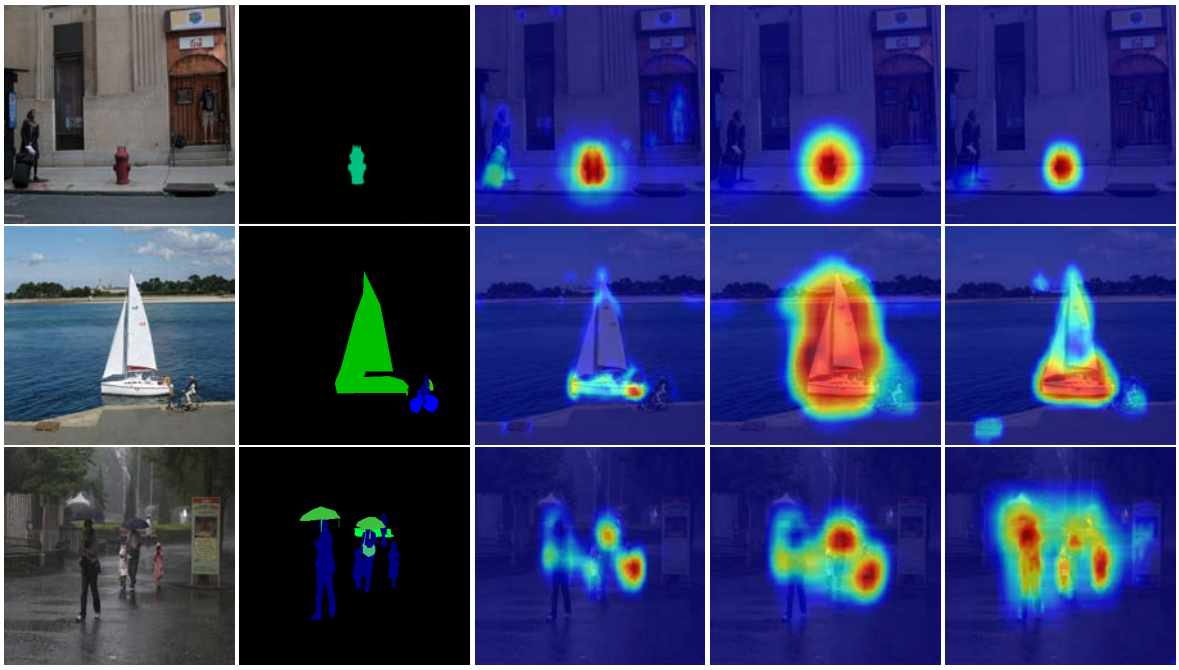


Figure 4.7: Visualization results on the MS COCO 2014 validation set.

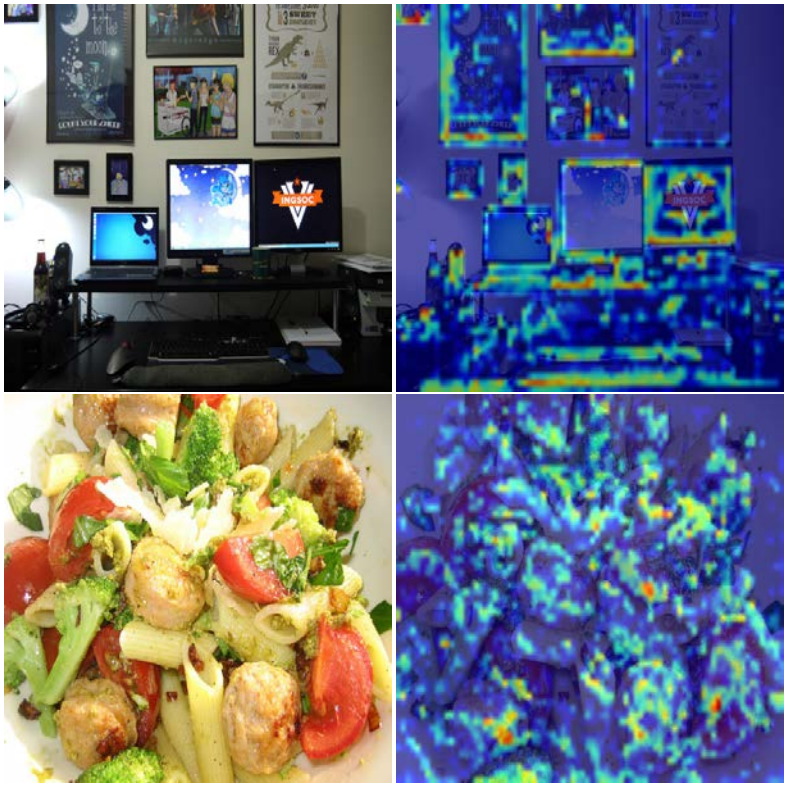


Figure 4.8: Illustration of misclassified samples.

Table 4.5: Performance comparison of various methods in mIoU (%) on the PASCAL VOC 2012 training set.

Method	mIoU(%)
ResNet50-CAM [143]	48.30
ResNet38-CAM [124]	47.43
Conformer-S-CAM [116]	51.70
(Ours) E-MHA	52.31

"broccoli". In future work, we must explore a more compatible feature extractor that can generate more robust local features.

4.4.3 Ablation Studies

This section presents ablation studies to analyze the contributions of each component in our eX-ViT, including the transformer encoder with the proposed Explainable Multi-Head Attention (E-MHA), the Attribute-guided Explainer (AttE), the global-level attribute-guided loss function $\mathcal{L}_{\text{global}}$, the local-level attribute discriminability loss function $\mathcal{L}_{\text{dis}}$, and the attribute diversity loss function $\mathcal{L}_{\text{div}}$.

Effectiveness of E-MHA

It is an intuition that the improved transformer attention mechanism in E-MHA will improve the model's ability to generate pseudo segmentation labels. In order to verify this idea, we simply apply CAM to the last transformer encoder layer. Table 4.5 reports the mIoU results of the pseudo labels generated by CAM with the backbone of ResNet38, ResNet50, Conformer-S [116], and the encoder E^{θ} in our proposed eX-ViT. As can be seen, the backbone of the E-MHA module shows superior performance to its CNN counterparts. Specifically, E-MHA-CAM achieves the mIoU of 52.31%, which is a significant gain of +4.92% and +4.01% over ResNet38-CAM and ResNet50-CAM, respectively. By comparing the E-MHA with the recent state-of-the-art architecture, i.e., Conformer-S [116], we find that our proposed E-MHA still achieves a promising result. In details, compared to CrossFormer-S [116] which explicitly uses multi-scale representations with convolutions to localize object details, E-MHA-CAM achieves the best mIoU of 52.31%, which is 0.61% points higher than CrossFormer-S-CAM. The performance improvement shows that exploiting the most frequent and robust features by use of E-MHA is highly effective for WSSS tasks that require discriminative features to localize instances.

Table 4.6: Effect of the contributions from various modules in mIoU (%) on the PASCAL VOC training set.

E^{θ}	$\mathcal{L}_{\text{global}}$	$\mathcal{L}_{\text{dis}}$	$\mathcal{L}_{\text{div}}$	training	validation
✓				44.72	50.20
✓	✓			53.71	55.43
✓		✓		51.25	54.63
✓			✓	52.13	55.50
✓	✓	✓		55.27	58.10
✓	✓		✓	58.08	59.82
✓	✓	✓	✓	**59.13**	**61.20**

Effectiveness of the AttE and Attribute-guided Loss

Table 4.6 gives an ablation study of each component in our proposed eX-ViT. We consider the first row as a baseline, where the results of the object localization maps are obtained via the CAM approach. As is observed from the table, the baseline can be further improved to 53.71% and 55.43% on the training and validation set, respectively by using the attribute features obtained via AttE to refine the learned transformer attention from the eX-ViT. Empirically, with attribute-guided discriminability loss $\mathcal{L}_{\text{dis}}$ the pseudo segmentation label maps can be improved by +6.53% compared to the baseline on the PASCAL VOC training set (51.25% vs. 44.72%) even without the global supervision $\mathcal{L}_{\text{global}}$. Moreover, the $\mathcal{L}_{\text{dis}}$ further improves the mIoU to 54.63% on the validation set. By incorporating the attribute diversity loss function $\mathcal{L}_{\text{div}}$ to explicitly regularize the attribute structure of the feature space, our full model gains promising results. Particularly, the results in Table 4.6 indicate that the proposed model performs better with the diversity constraint $\mathcal{L}_{\text{div}}$ on the local consistency, which brings +4.37% and 4.39% mIoU improvements on the training and validation sets, respectively compared to the global-level loss. This also confirms our theory that improving the diversity among attributes promotes the learning of comprehensive localization maps.

Influence of the Number of Fused Transformer Layers

We further explore the impact of the number of fused transformer layers in Eq. 4.10 on the PASCAL VOC training set. Following the common practice in the prior work [124], we adopt three metrics to evaluate the performance, i.e., false positives (FP), false negatives (FN), and mIoU. The larger FP and FN values denote higher degrees of over-activated and under-activated areas, respectively. In Fig. 4.9, we compare the performance of the model variants using different numbers of the fused transformer layers. As is observed, when fusing layers with more than 10, we obtain localization maps with a larger FN value, which suggests the generated localization maps have

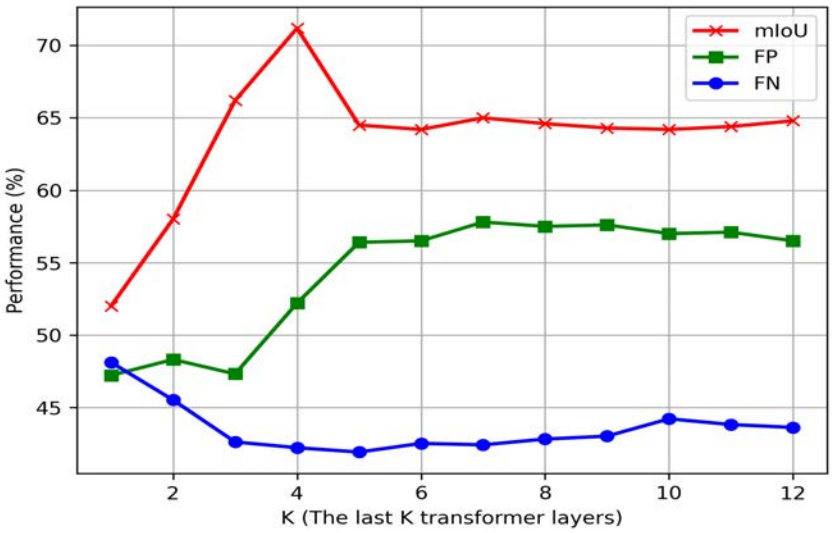


Figure 4.9: Evaluation of object localization maps generated by fusing the class-specific attentions from the last K transformer layers in eX-ViT's encoder E^{θ} in terms of false positives (FP), false negatives (FN) and mIoU. The larger FP and FN values denote having more over-activated pixels, while the higher mIoU value indicates the generated localization maps have fewer over-activated pixels and more complete object coverage.

more over-activated pixels and less complete activation coverage. This is mainly due to the limited ability of lower layers to encode high-level representations. By reducing the number of fused layers from the encoder E^{θ}, the performance of predicted localization maps becomes much better, i.e., more complete activation coverage (lower FN value) or fewer over-activated regions (lower FP value). Overall, the evaluation results indicate that using the last three attention layers can achieve the best mIoU of 71.2% with lower FN and FP values. Therefore we set $K = 3$ throughout this chapter.

Influence of the Number of Attributes

The attribute-guided scheme allows the model to encode richer semantics into each attribute feature at a granular level. In order to discover the most suitable S concerning different datasets, we conduct extensive experiments to compare the performance of the model variants with different settings of hidden dimension c and the number of attributes S. As shown in Table 4.7, when $c = 128$, the model learns weaker representations for both datasets. In contrast, the performance becomes much better when the hidden dimension is enlarged to 512. However, as the number of attributes increases to 16, the model exhibits poor mIoU accuracy. In the end, we find that $c = 256$ achieves consistently superior performance across a range of attribute numbers. The best performance is achieved when $S = 8$ on the PASCAL VOC 2012 *val* set, and $S = 16$ on

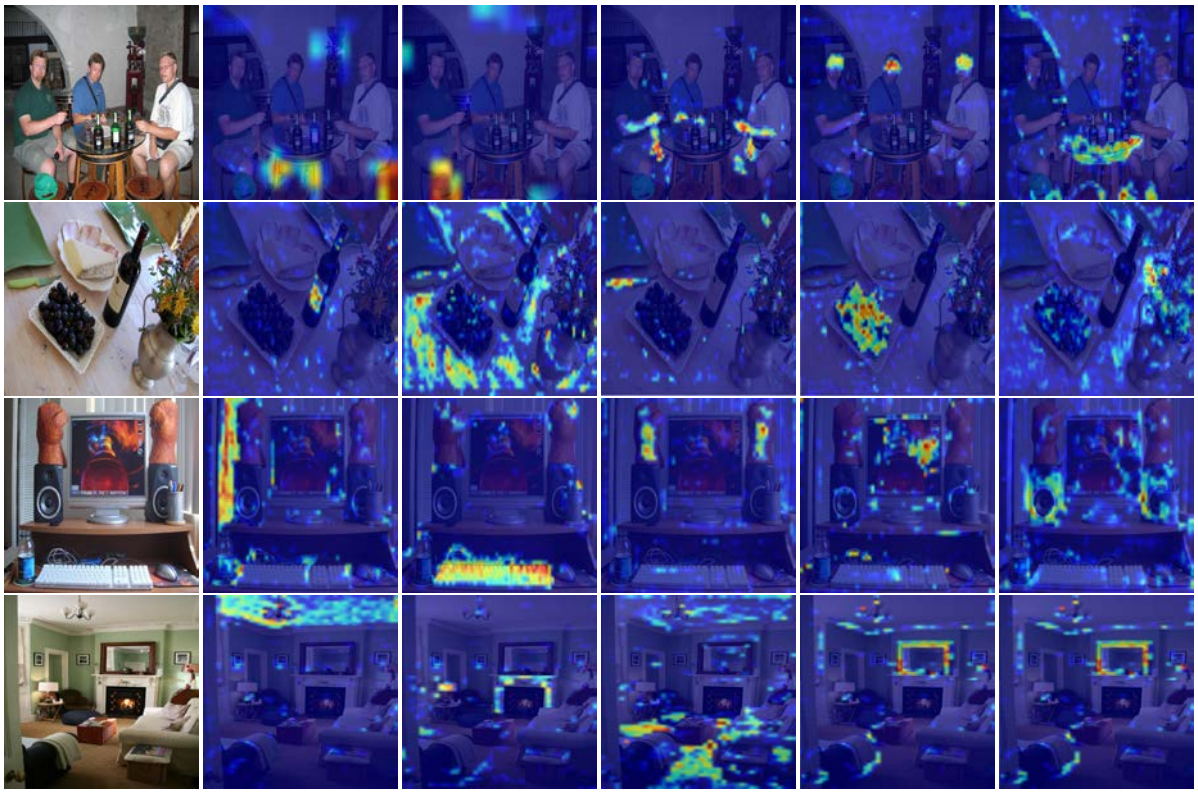


Figure 4.10: Visualization of the learned attributes on the PASCAL VOC 2012 validation set, and MS COCO 2014 validation set, respectively. In each row, the left part is the input image, and the rest of images visualize the top-5 attributes, which shows that AttE attends to the discriminative attributes with a high degree of detail.

the MS COCO 2014 *val* set. These observations suggest that images in MS COCO 2014 tend to contain more local features that are discriminative for object localization.

Additionally, we use the Grad-CAM as a tool to visualize the learned attributes and use Fig. 4.10 to present the visualization results. In each row of Fig. 4.10, the left column is the input image, and the images in the rest columns visualize the top 5 attributes. The first two rows are from the PASCAL validation set, whereas the last two rows are from the COCO validation set. By examining the visualization results presented in Fig. 4.10, several observations can be made regarding the effectiveness of the AttE in localizing object attributes. Firstly, the AttE is able to effectively focus on the compact regions of most objects, which is consistent with human observations. Secondly, for large-area attributes such as the table and ceiling, the learned attributes can accurately attend to the corresponding areas. Finally, the AttE is capable of attending to the regions of small but important attributes such as the fork and head. With these observations, we can ascertain how the AttE decomposes the feature map into different attributes.

Table 4.7: The influence of the number of attributes in mIoU (%) on the PASCAL VOC and MS COCO 2014 validation sets.

c	S	PASCAL VOC 2012 *val*	MS COCO 2014 *val*
512	8	69.42	40.31
512	16	69.03	38.92
256	4	63.48	36.69
256	8	**71.23**	41.23
256	16	70.29	**42.92**
128	4	68.63	37.25
128	8	68.56	38.79

Table 4.8: The influence of hyperparameters in mIoU (%) on the PASCAL VOC validation and test sets.

Hyperparameter	value	PASCAL VOC 2012 *val*	PASCAL VOC 2012 *test*
α	0.1	69.8	69.4
	0.3	70.5	70.6
	0.5	**71.2**	**71.1**
	1.0	70.6	70.4
β	0.1	69.5	69.1
	0.3	70.1	70.3
	0.5	70.6	70.2
	1.0	**71.2**	**71.1**

Influence of Hyperparameters

In this section, we explore how variations of hyperparameters can impact the performance of our model. For this purpose, we train models on PASCAL with each hyperparameter modification and report the accuracy in Table 4.8. It is observed that when α is small (<0.5), there is a slight performance drop. On the other hand, there is a significant accuracy drop when β is smaller than 1.0. This confirms that our model learns better features when our diversity loss enforces the attribute features to the maximally independent of each other to capture broader visual clues of objects.

4.5 Conclusion

In this chapter, we proposed the eX-ViT, a new explainable vision transformer for weakly supervised semantic segmentation. In our framework, a novel Explainable Multi-Head Attention (E-MHA) module is proposed to produce discriminative feature representations with inherent explainability and noise robustness. Which is achieved

by optimizing the dynamic alignment between the input tokens and attention weights. Moreover, a new Attribute-guided Explainer (AttE) module is designed to decompose the attention maps into the contribution of each individual attribute, empowering the feature representation with a set of attribute maps at a granular level. Based on AttE, we develop a self-supervised attribute-based loss to guide the learning of attribute features with the attribute discriminability mechanism and attribute diversity mechanism, which promotes the generation of diverse and discriminative object attributes. Extensive experiments were presented to demonstrate that the eX-ViT surpasses the state-of-the-art CNNs and transformers on two well-known benchmarks. We hope that the eX-ViT's superior performance on WSSS tasks will inspire future research on the exploitation of the explainability of transformers.

Although our work has shown promising results, a limitation is that the proposed model does not incorporate attribute-level ground-truth labels. For future studies, the model should potentially be further improved if prior fine-grained knowledge of various attributes is integrated. Therefore, we plan to develop approaches to learn and integrate the knowledge in our future work.

Chapter 5

Explainability-driven Model Compression for Deep Neural Networks

Recently, deep learning research community has proposed several deep neural networks like transformers that become the cornerstone of many AI applications, powering various tasks such as text summarization, question answering, and visual analysis. If the model's outputs are used by humans to make decisions or take actions, explainability can enhance trust in the system and facilitate its acceptance. However, how to make these models explainable remains underexplored. This chapter investigates a model pruning technique aimed at enhancing the explainability of these well-trained models. We first introduce an explainability-aware mask for each prunable unit in a model, with the goal of quantifying its contribution to predicting each class. Specifically, the proposed mask is fully differentiable and can be learned in an end-to-end manner. We demonstrate many benefits of the proposed mask, including more accurate pruning and fewer computational costs compared with existing black-box pruning methods. Then, this chapter describes how to learn the layer-wise pruning thresholds that differentiate the important and less-important units via a differentiable pruning operation. Lastly, experimental results on various models are provided to demonstrate the efficacy of the proposed method.

5.1 Introduction

Over the last few years, transformers have attracted increasing attention in various challenging domains, such as natural language processing, vision, or graphs [109], [144]. It is composed of two key modules, namely the Multi-Head Attention (MHA) and Multi-Layer Perceptron (MLP). However, similar to CNNs, the major limitations of transformers include the gigantic model sizes with intensive computational costs.

Which severely restricts their deployment in resource-constrained devices like edge platforms. To compress and accelerate transformer models, a variety of techniques naturally emerge. Popular approaches include weight quantization [145], knowledge distillation [146], filter compression [147], and model pruning [148]. Among them, model pruning especially structured pruning has gained considerable interest that removes the least important parameters in pre-trained models in a hardware-friendly manner, which is thus the focus of this chapter.

Due to the significant structural differences between CNNs and transformers, although there is prevailing success in CNN pruning methods, the research on pruning transformers is still in the early stage. Existing studies could empirically be classified into three categories. (1) Criterion-based pruning resorts to preserving the most important weights/attentions by employing pre-defined criteria, e.g., the L1/L2 norm [149], or activation values [150]. (2) Training-based pruning retrains models with handcrafted sparse regularizations [151] or resource constraints [146], [152]. (3) Architecture-search pruning methods directly search for an optimal sub-architecture based on predefined policies [144], [153]. Although these studies have made considerable progress, two fundamental issues have not been fully addressed, i.e., the optimal layer-wise pruning ratio and the weight importance measurement.

For the first issue, the final performance is notably affected by the selection of pruning rates for different layers. To this end, some relevant works have proposed a series of methods for determining the optimal per-layer rate [154], [155]. For instance, Michel *et al.* [156] investigate the effectiveness of attention heads in transformers for NLP tasks and propose to prune attention heads with a greedy algorithm. Yu *et al.* [146] develop a pruning algorithm that removes attention scores below a learned per-layer threshold while preserving the overall structure of the attention mechanism. However, the proposed methods do not take into account the inter-dependencies between weight. Recently, Zhu *et al.* [151] introduce the method VTP with a sparsity regularization to identify and remove unimportant patches and heads from the vision transformers. However, VTP needs to try the thresholds manually for all layers.

For the second issue, previous studies resort to identifying unimportant weights by various importance metrics, including magnitude-based, gradient-based [157], [158], and mask-based [159]. Among them, the magnitude-based approaches usually lead to suboptimal results as it does not take into account the potential correlation between weights [147]. In addition, gradient-based methods often tend to prune weights with small values, as they have small gradients and may not be identified as important by the backward propagation. Finally, the limitation of current mask-based pruning lies in two folds: (1) Most mask-based pruning techniques manually assign a binary

mask w.r.t. a unit according to a per-layer pruning ratio, which is inefficient and sub-optimal. (2) Most works use a non-differentiable mask, which results in an unstable training process and poor convergence.

In this chapter, we propose a novel explainable structured pruning framework for vision transformer models, termed X-Pruner, by considering the explainability of the pruning criterion to solve the above two problems. As stated in the eXplainable AI (XAI) field [35], important weights in a model typically capture semantic class-specific information. Inspired by this theory, we propose to effectively quantify the importance of each weight in a class-wise manner. Firstly, we design an explainability-aware mask for each prunable unit (e.g., an attention head or matrix in linear layers), which measures the unit's contribution to predicting every class and is fully differentiable. Secondly, we use each input's ground-truth label as prior knowledge to guide the mask learning, thus the class-level information w.r.t. each input will be fully utilized. Our intuition is that if one unit generates feature representations that make a positive contribution to a target class, its mask value w.r.t. this class would be positively activated, and deactivated otherwise. Thirdly, we propose a differentiable pruning operation along with a threshold regularizer. This enables the search of thresholds through gradient-based optimization, and is superior to most previous studies that prune units with hand-crafted criteria. Meanwhile, the proposed pruning process can be done automatically, i.e., discriminative units that are above the learned threshold are retained. In this way, we implement our layer-wise pruning algorithm in an explainable manner automatically and efficiently. In summary, we make the following contributions:

- We propose a novel explainable structured pruning framework dubbed X-Pruner, which prunes units that make less contributions to identifying all the classes in terms of explainability. To the best knowledge of the authors, this is the first work to develop an explainable pruning framework for vision transformers;

- We propose to assign each prunable unit an explainability-aware mask, with the goal of quantifying its contribution to predicting each class. Specifically, the proposed mask is fully differentiable and can be learned in an end-to-end manner;

- Based on the obtained explainability-aware masks, we propose to learn the layer-wise pruning thresholds that differentiate the important and less-important units via a differentiable pruning operation. Therefore, this process is done in an explainable manner;

- Comprehensive simulation results are presented to demonstrate that the proposed X-Pruner outperforms a number of state-of-the-art approaches, and shows its superiority in gaining the explainability for the pruned model.

5.2 Related Work

5.2.1 Pruning for Transformers

Pruning has been a popular approach for removing the least important weights in transformer models. The existing methods can be mainly categorized into unstructured and structured pruning. For unstructured pruning, techniques such as magnitude-based and hessian-based have been proposed [154], [160]. However, they result in irregular sparsity, causing sparse tensor computations that are difficult to align with hardware efficiency.

The above problem can be alleviated by structured pruning, where uninformative contiguous structures of a pre-trained model such as attention heads, rows of weight matrix, are removed. For instance, Michel *et al.* [156] found that a large percentage of attention heads can be pruned without scarifying much performance. Fan *et al.* [153] proposed a structured dropout, which selects sub-structures of a model during the inference time. Wang *et al.* [159] pruned rank-1 components inside large language models using a parameterization method. Liu *et al.* [161] assembled several model compression techniques on a range of pre-trained language models, and gained impressive results.However, these works focus on pruning transformers for NLP tasks.

For vision transformers, Chen *et al.* [150] explored unstructured and structured sparsity, and proposed a first-order importance approximation method to remove attention heads. Recently, Yu *et al.* [146] propose a structured pruning method for vision transformers, which involves a $0/1$ mask that differentiates unimportant/important parameters based on the magnitude of the model parameters. Although it uses a differentiable threshold, the mask is non-differentiable, which could cause the gradients to be biased and result in suboptimal results of the remaining weights. Yu *et al.* [152] proposed to integrate three efficient approaches including pruning, layer skipping, and knowledge distillation into a unified framework to produce a compact transformer. Although these existing methods have made significant advances, the designing of importance metrics remains an open problem to explore.

5.3 Method

5.3.1 Problem Definition

Our proposed X-Pruner aims to explore structured pruning by removing prunable units (e.g., rows of weight matrix and attention heads) in vision transformers. Let $\mathcal{D}$ be a training dataset, which consists of N training pairs $\{(\boldsymbol{x}_1, \boldsymbol{y}_1), ..., (\boldsymbol{x}_N, \boldsymbol{y}_N)\}$. Considering an L-layer transformer $f(\boldsymbol{W})$, its parameters are represented by $\boldsymbol{W} = (\boldsymbol{W}^1, \boldsymbol{W}^2, ..., \boldsymbol{W}^L)$,

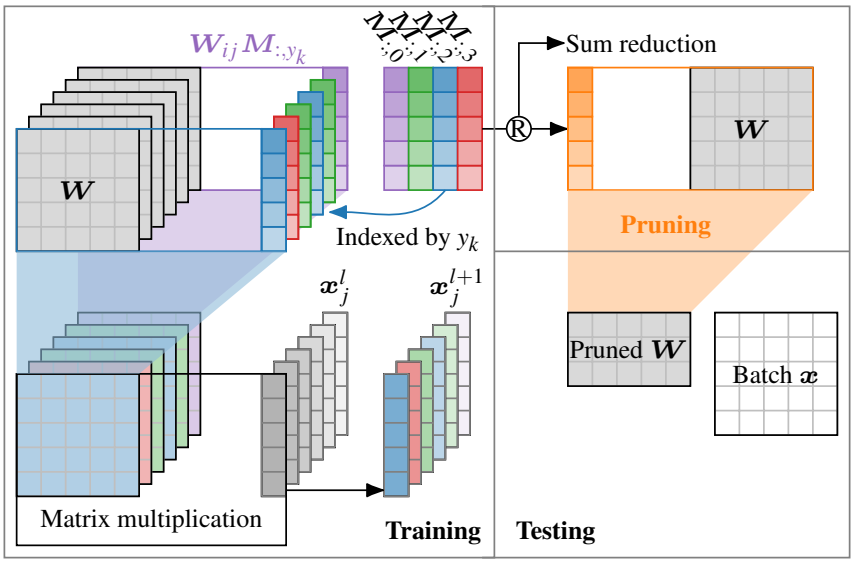


Figure 5.1: Pipeline of our proposed X-Pruner framework. We first train a transformer with the proposed explainability-aware masks, with the goal of quantifying each unit's contribution to predicting each class. Then we explore the layer-wise pruning threshold under a pre-defined cost constraint. Finally, a fine-tune procedure is executed for the pruned model.

where $\boldsymbol{W}^l \in \mathbb{R}^{d_l}, 1 \leq l \leq L$, d_l is the number of prunable parameters in the l-th layer. Given a target pruning ratio α, the pruning process can be regarded as the form of layer-wise operation with pruning rates $\{r_l\}_{l=1}^L$:

$$\begin{aligned}(r_1, r_2, ..., r_L)^* &= \text{argmin}\mathcal{L}(f(\boldsymbol{W}; r_1, r_2, ..., r_L; \boldsymbol{x}), \boldsymbol{y}), \\ s.t. \quad &\frac{P(f(\boldsymbol{W}; \{r_l\}))}{P(f(\boldsymbol{W}))} \geq \alpha,\end{aligned} \tag{5.1}$$

where r_l is the l-th layer's pruning rate, and $P(\cdot)$ is a resource evaluation metric.

5.3.2 The Proposed X-Pruner

Explainability-aware Mask

To fully utilize the class-level information, we propose to assign each prunable unit an explainability-aware mask, which is used to quantify the contribution of each unit to identifying every class. Specifically, the proposed mask is a class-level mask for each class instead of a scalar mask for all classes. For instance, given the weights in the l-th self-attention layer consists of query $\boldsymbol{W}_l^Q \in \mathbb{R}^{n \times d}$, key $\boldsymbol{W}_l^K \in \mathbb{R}^{n \times d}$, and value $\boldsymbol{W}_l^V \in \mathbb{R}^{n \times d}$, where n and d are the number of input and output dimension. The mask for head h is formulated as $\boldsymbol{M}_{l,h}^H \in \mathbb{R}^{C \times d}$, where C is the total number of classes. That is to say, $\boldsymbol{M}_{l,h,i}^H$ is built to quantify the contribution of head h for recognizing the i-th class.

Evidently, a scalar mask used in prior works is a special case of our method where values of $\boldsymbol{M}^H_{l,h,i}$ are the same. Thus, given input $\boldsymbol{x}_i$ with its class label $\boldsymbol{y}_i$, to apply the mask, the product between weight and its corresponding mask is performed. That is, the self-attention operation for head h can be expressed as follows:

$$\boldsymbol{\alpha}_{l,h} = S(\frac{(\boldsymbol{W}^Q_{l,h}\boldsymbol{x}_i)^T \boldsymbol{W}^K_{l,h}\boldsymbol{x}_i}{\sqrt{d}}), \tag{5.2}$$

$$\text{Attn}_{l,h}(\boldsymbol{x}) = \boldsymbol{\alpha}_{l,h}\boldsymbol{W}^V_{l,h}\boldsymbol{x}_i, \tag{5.3}$$

$$\text{MHA}(\boldsymbol{x}, \boldsymbol{M}^H_l) = \textstyle\sum_{h=1}^{H} \boldsymbol{M}^H_{l,h,\boldsymbol{y}_i} \text{Attn}_{l,h}(\boldsymbol{x}), \tag{5.4}$$

where $S(\cdot)$ is the softmax function, $\boldsymbol{\alpha}_h$ is the h-th attention weight, and H is the total number of attention heads.

Meanwhile, we apply the similar idea to the MLP and other linear projection layers. Let us denote the weight matrix in a linear layer by $\boldsymbol{W}_l \in \mathbb{R}^{m\times n}$, where the m and n are the dimensions. Its corresponding mask is defined by $\boldsymbol{M}^F_l \in \mathbb{R}^{C\times m\times n}$. Then, the feed forward process in the linear layer is expressed as:

$$\text{FC}(\boldsymbol{Z}_l, \boldsymbol{M}^F_l) = \boldsymbol{M}_{l,\boldsymbol{y}_i} \boldsymbol{W}^F_l \boldsymbol{Z}_l, \tag{5.5}$$

where $\boldsymbol{Z}_l$ is the input to the l-th layer. We omit the bias across all layers for simplicity.

Recall that our explainability-aware mask aims to identify weights influential to the predicted label. As such, it is desirable for mask $\boldsymbol{M}_{:,c}$ to vary slowly if input images all belong to the same class c, rendering a smooth explainability-aware mask. Therefore, we propose to add a smoothness-aware constraint for the mask. More specifically, we take the second derivative of the mask values w.r.t. the input and predicted class, and choose its L_1 norm as the smoothness-aware constraint:

$$\mathcal{L}_{\text{smooth}}(\boldsymbol{M}) = \sum_{l=1}^{L}\sum_{c=1}^{C} |\nabla^2 \boldsymbol{M}^l_{:,c}|_1. \tag{5.6}$$

Moreover, to address the issue of redundancy among the prunable units, rather than declaring all units as relevant to the model's prediction, we impose the following sparsity constraint on the masks:

$$\mathcal{L}_{\text{sparse}}(\boldsymbol{M}) = \sum_{l=1}^{L}\sum_{c=1}^{C} ||\boldsymbol{M}^l_{:,c}||_2. \tag{5.7}$$

Overall, the total loss function is defined as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{ce}} + \lambda_{\text{sm}}\mathcal{L}_{\text{smooth}} + \lambda_{\text{sp}}\mathcal{L}_{\text{sparse}}(\boldsymbol{M}), \tag{5.8}$$

where $\mathcal{L}_{\text{ce}}$ is the cross-entropy loss, λ_{sm} and λ_{sp} are the hyperparameters.

Unlike prior works that use a binary mask to quantify the contribution of each unit for all classes, we propose to capture the importance of every unit w.r.t. each class with a differentiable mask. After training, the sum value of each learned mask explicitly denotes its contribution to identifying all classes. In this way, our learned explainability-aware masks gain the representation ability for revealing the inner reasoning process in transformers in an end-to-end manner, which essentially offers a global examination of the importance of every single unit in an intuitively explainable manner. Noticeably, the weights of the pre-trained model remain fixed during the training procedure. Therefore, we empirically observe that only a few epochs are required to train our proposed explainability-aware mask.

Explainable Pruning

The goal of the proposed X-Pruner is to preserve the most important units for identifying target classes in a pruned model. This is achieved by removing units with the least-impact explainability-aware masks. Previous works resort to measuring the importance of individual units with a manually chosen per-layer threshold, which is computationally intractable as the parameter search space is exhaustive [152]. In this work, we propose to learn the layer-wise threshold by designing a differentiable pruning operation along with a threshold regularizer, which is superior to most prior studies with better control over the non-uniform sparsity.

Intuitively, with the obtained explainability-aware masks, the less-important units with mask values below a certain threshold should be pruned, while important ones are preserved. However, most of current approaches use a manually selected threshold, which is difficult to optimize in a trainable process. To tackle this issue, we propose a differentiable pruning operation for explainability-aware masks. Mathematically, the differentiable pruning operation is expressed as follows:

$$\hat{\boldsymbol{M}}^l = \begin{cases} \boldsymbol{M}^l \tanh(n(\boldsymbol{M}^l - \theta^l)), & \boldsymbol{M}^l \in \Phi(\boldsymbol{M}^l | 1 - r^l), \\ p \tanh(n(\boldsymbol{M}^l - \theta^l)), & \text{otherwise}, \end{cases} \tag{5.9}$$

where r^l is the pruning ratio for layer l, and $\Phi(\boldsymbol{M}^l|1-r^l)$ is a function that returns the top $(1-r^l)\%$ sorted elements in $\boldsymbol{M}^l$. With a proper setting of n and p, the value of $\tanh(\cdot)$ asymptotically approaches one for $\boldsymbol{M}^l \in \Phi(\boldsymbol{M}^l|1-r^l)$, which results in $\hat{\boldsymbol{M}}^l \approx \boldsymbol{M}^l$. In that case, our proposed differentiable pruning operation implies that discriminative units that contribute more to identifying classes above an adaptive threshold are retained, while those that contribute less are suppressed. By assigning a large positive value to n, our proposed pruning function enables learning the threshold θ^l

with the backward gradient. In our experiments, we empirically verify that letting $p = 500$ and $n = 10$ guarantees a stable training process and yields good results for pruning.

Subsequently, we compute the accumulated pruning rate R across all prunable layers as follows:

$$R = \sum_{l=1}^{L} \frac{r^l * n^l}{N}, \tag{5.10}$$

where n^l represents the total prunable parameters of the layer l and N denotes the number of all unpruned parameters.

To learn the layer-wise pruning rate with the given pruning rate α in an end-to-end manner, we propose a novel regularization term $\mathcal{L}_{\mathrm{R}}$ in the augmented Lagrangian method, which converts the optimization problem in Eq. 5.1 to an unconstrained penalized expression. Specifically, it is expressed as

$$\mathcal{L}_{\mathrm{R}} = \beta(\alpha - R)^2 + \gamma(\alpha - R), \tag{5.11}$$

where β and γ are trainable parameters, and the unconstrained problem of Eq. 5.11 can be solved using gradient descent-based techniques. Overall, the total loss function for the proposed X-Pruner is given by

$$\mathcal{L} = \mathcal{L}_{\mathrm{ce}} + \mathcal{L}_{\mathrm{R}}. \tag{5.12}$$

The optimization problem of Eq. 5.12 allows us to lift up units with discriminative masks that are important to the model decisions while suppressing less-important ones. Moreover, it implies that the layer-wise pruning rate r^l tends to be larger when it has larger n^l, which is natural for exploiting the dynamic sparsity across all layers.

After training, we accordingly remove the least-impact units with the learned pruning rate $\{r^1, r^2, ..., r^L\}$, and integrate the left explainability-aware masks $\boldsymbol{M}$ into the pruned model per layer by setting $\boldsymbol{W} = \boldsymbol{W} * \boldsymbol{M}$, and further fine-tune the pruned model. In summary, our proposed explainable pruning method X-Pruner that is capable of identifying and preserving important units in an explainable and trainable way, which overcomes the drawbacks of existing black-box pruning methods and provides empirical guarantees on the accuracy of the pruned model.

5.4 Experiments

To evaluate the performance of the X-Pruner, we conduct experiments on the CIFAR-10 [162] and ILSVRC-12 datasets [163]. CIFAR-10 includes 10 classes, consisting of 50K

training and 10K validation images. ILSVRC-12 contains images of 1K classes, and its training and validation sets have 1.28M images and 50K images, respectively. For a fair comparison with existing methods, we prune the DeiT [164] and Swin Transformer [165] architectures on classification tasks [146], [152]. Additionally, we conduct a series of ablation studies to discover the performance contribution from different components in our framework.

5.4.1 Implementation Details

All experiments are implemented using PyTorch on NVIDIA Tesla V100 GPUs. We use pre-trained weights to initialize vision transformer models and use them as baseline models. During the training process for explainability-aware masks, the learning rate is set to be 0.01 with a batch size of 128, and we use the SGD optimizer with momentum 0.9. Empirically, the mask training process is 50 epochs for the DeiT and 30 epochs for the Swin Transformer. Which takes around 300 V100 GPU hours. In the explainable pruning process, we initially set all r^l to α. The learning rate for θ^l and r^l is set to be 0.02 and fine-tuned with the AdamW optimizer. The learning rate for the other parameters and momentum are 5 $\times 10^{-4}$ and 0.9, respectively. The DeiT models are trained for 80 epochs and Swin Transformers are trained for 30 epochs. We follow the training strategies used in the original DeiT and Swin Transformers [165] except knowledge distillation. β and γ are initialized to be zero and then optimized during training.

5.4.2 Main Results

Table 5.1 shows the superiority of X-Pruner over other state-of-the-art methods on ILSVRC-12. We observe that most existing pruning methods cannot provide noticeable FLOP savings without too much accuracy degradation. Instead, by learning the differentiable explainability-aware masks, our X-Pruner can reduce the computational costs by 51.3%-66.1% with much lower accuracy drops (0.72%-1.1%). Specifically, when pruning the DeiT-T, compared with WDPruning [146] that can only save 46.2% FLOPs, it is observed that our proposed X-Pruner achieves much larger FLOP saving (66.1% vs. 46.2%) with less accuracy degradation (1.1% vs. 1.86%). For the larger model DeiT-S, while UVC [152] achieves the state-of-the-art top-1 accuracy among the existing methods, which is 78.82% with a 49.6% reduction in FLOPs, the X-Pruner reduces the FLOPs by 51.3% while obtaining the top-1 accuracy of 79.04%. These results demonstrate that the proposed X-Pruner outperforms existing pruning methods with more compact model sizes and better performance.

Meanwhile, we investigate the efficacy of our proposed method on another popular transformer, *i.e.*, Swin Transformer [165]. The experimental results are presented in

Table 5.1: Comparison with the state-of-the-art methods on the ILSVRC-12 dataset. FLOPs remained denotes the remained ratio of FLOPs to the full-model FLOPs. $*$ indicates utilizing knowledge distillation in the training process.

Model	Method	Top-1 Acc. (%)	Top-5 Acc. (%)	FLOPs (G)	FLOPs remained (%)
DeiT-T	Baseline	72.2	91.10	1.3	100
	SCOP [147]	68.9	89.00	0.8	61.5
	HVT [148]	69.7	89.40	0.7	53.8
	UVC* [152]	70.6	-	0.5	39.1
	WDPruning [146]	70.3	89.82	0.7	53.8
	X-Pruner	**71.1**	**90.11**	**0.6**	**49.2**
DeiT-S	Baseline	79.8	95.00	4.6	100
	SCOP [147]	77.5	93.50	2.6	56.5
	HVT [148]	78.0	93.83	2.4	52.2
	UVC* [152]	78.82	-	2.3	50.4
	WDPruning [146]	78.38	94.05	2.6	56.5
	X-Pruner	**78.93**	**94.24**	**2.4**	**52.1**
DeiT-B	Baseline	81.8	95.59	17.6	100
	SCOP [147]	79.7	94.50	10.2	58.3
	UVC* [152]	80.57	-	8.0	45.5
	WDPruning [146]	80.76	95.36	9.9	56.3
	X-Pruner	**81.02**	**95.38**	**8.5**	**48.5**

Table 5.2: Pruning results of Swin Transformer on the ILSVRC-12 dataset.

	Method	Top-1	FLOPs	Top-1 ↓	FLOPs ↓ (%)
Swin-T	Baseline	81.2	4.5	0.0	0.0
	STEP [166]	77.2	3.5	4.0	22.2
	ViT-Slim [144]	80.7	3.4	0.5	24.4
	X-Pruner (Ours)	**80.7**	**3.2**	**0.5**	**28.9**
Swin-S	Baseline	83.2	8.7	0.0	0.0
	STEP [166]	79.6	6.3	3.6	27.6
	WDPruning [146]	81.8	6.3	1.4	27.6
	X-Pruner (Ours)	**82.0**	**6.0**	**1.2**	**31.0**

Table 5.2. For the Swin-T, the X-Pruner yields significantly better top-1 accuracy with substantially fewer FLOPs. More specifically, our method obtains 28.9% FLOPs saving, and the top-1 accuracy only drops by 0.5%. When pruning the Swin-S, compared to the state-of-the-art method WDPruning [146] which considers the dimensions for pruning, our X-Pruner also shows impressive superiority thanks to the use of the explainability-aware mask.

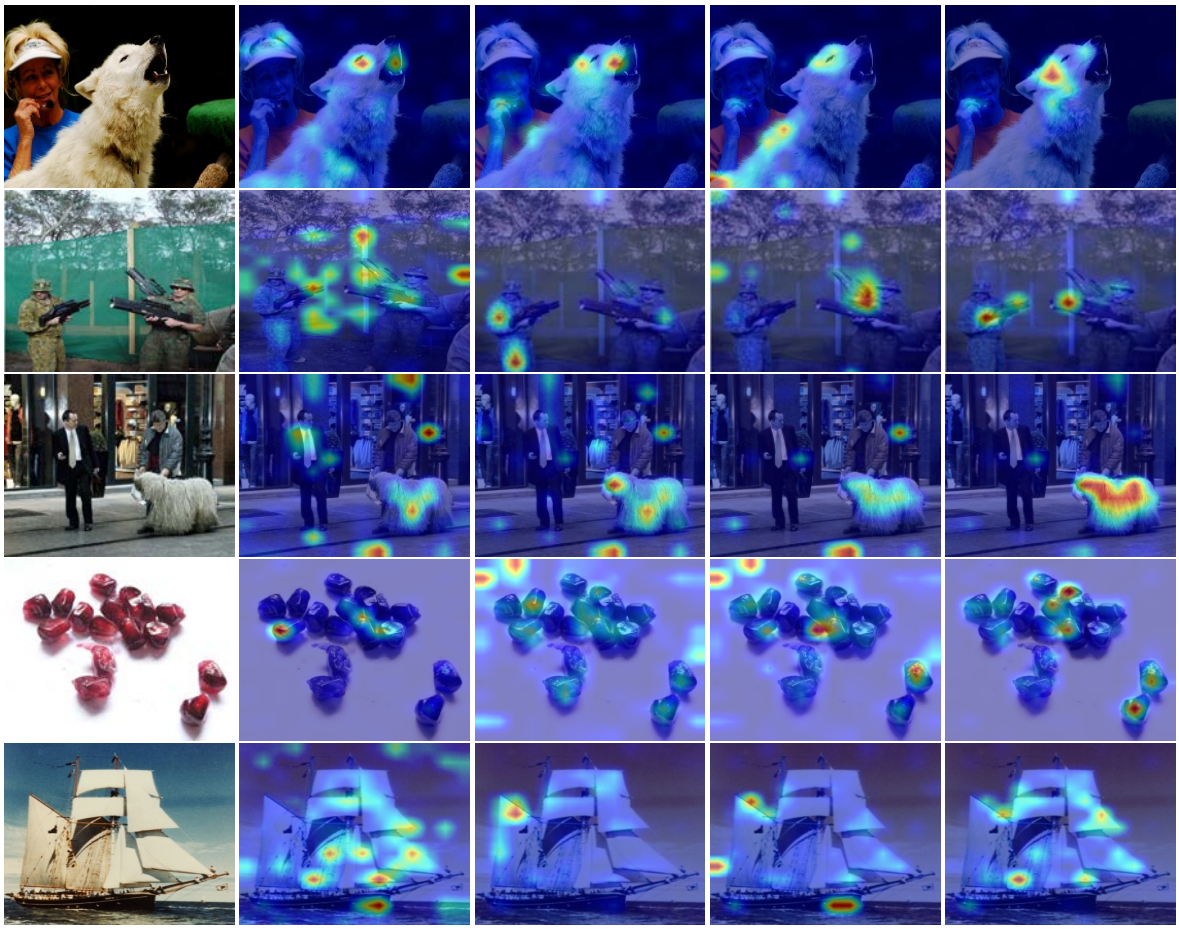


Figure 5.2: Visual explanations generated by a variety of pruned networks on the ILSVRC-12 validation set.

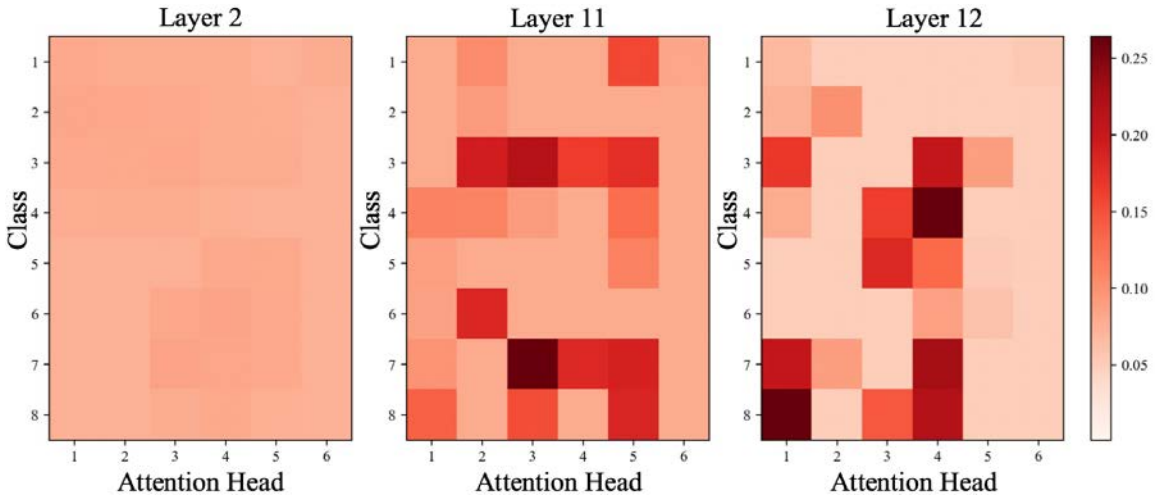


Figure 5.3: Explainability-aware mask values in varying layers for DeiT-S.

Table 5.3: Main results for pruning Swin-T under different configurations on ILSVRC-12.

Setting	Top-1 ↓ (%)	FLOPs ↓ (%)
w/o mask	2.65	28.9
w/o $\mathcal{L}_{\text{smooth}}$	1.02	29.3
w/o $\mathcal{L}_{\text{sparse}}$	1.92	29.1
X-Pruner	**0.51**	**28.9**

5.4.3 Visualization and Analysis

We visualize the class-level visual explanation maps based on the DeiT-S as well as its pruned models by the LRP-based relevance method [118]. Fig. 5.2 provides a visual comparison based on randomly chosen ILSVRC-12 validation images. As can be seen from the figure, most of the visual explanation results of the full model still appear noise-like patterns to humans. However, the maps produced on the pruned model obtained by WDPruning [146] and UVC [152] are distorted. Though the predictions of the pruned models are correct, they produce incorrect explanation maps after the pruning process. Instead, we observe that the visual explanation maps produced on the pruned model of our X-Pruner are more compact and contain less noise.

Moreover, the learned mask values of attention layers shown in Fig. 5.3 demonstrate that the proposed X-Pruner discovers the head importance appropriately without per-layer pruning ratio. Notably, the masks at higher layers (Layers 11 and 12) have higher values compared to the masks in Layer 2. Which indicates that in transformers, the lower layers attend to both local and global information, whereas the higher layers attend to global information. Thus rich semantic-level features are captured at higher layers, which are essential for the final predictions.

We further compare our X-Pruner with the state-of-the-art method WDPruning [146] on CIFAR-10. Fig. 5.4 depicts the top-1 accuracy of the DeiT-S with various pruning rates. As can be seen from the figure, at lower pruning rates, e.g., 10%, both methods achieve slightly higher accuracy compared to the baseline. When it comes to larger pruning rates, compared to WDPruning [146], our X-Pruner suffers less accuracy loss with the same pruning rates (e.g., 50% or 70%).

5.4.4 Ablation Studies

In this subsection, we first evaluate the effectiveness of explainability-aware masks in our proposed method based on the Swin-T model. Table 5.3 shows the detailed results, all of which are pruned using similar FLOPs pruning rates for a fair comparison.

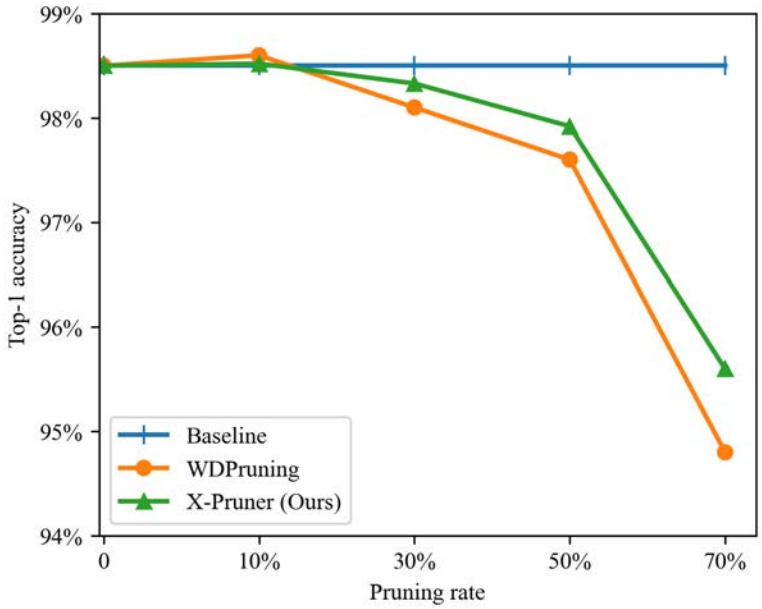


Figure 5.4: Top-1 accuracy for DeiT-S on CIFAR-10 with various pruning rates. "Baseline" denotes the unpruned baseline model.

Table 5.4: Main results of learnable pruning rate on DeiT-S.

Method	Top-1 ↓ (%)	FLOP ↓ (%)
Random pruning	2.28	47.2
Uniform pruning	4.05	47.4
X-Pruner	**0.87**	**47.9**

We first employ a class-agnostic strategy to train the explainability-aware mask, denoted as a w/o explainability-aware mask. That is, use the same mask for all the input given different classes. However, this strategy causes serious performance degradation (2.65%) since it loses the class-wise signal to identify each unit's contribution. We further explore the impact of optimization constraints. Moreover, as is observed from Table 5.3, when the masks are trained without the sparrse regularizer λ_{sparse}, the trained model suffers a drop of 1.92% in top-1 accuracy. Which proves our method effectively alleviates the problem of over-fitting and improves the performance. Finally, if the smooth constraint λ_{smooth} is removed, the top-1 accuracy is decreased by 1.02%. Overall, our proposed method X-Pruner is able to prune models effectively with desirable accuracy.

In Table 5.4, we further investigate the layer-wise pruning rate on ILSVRC-12 and

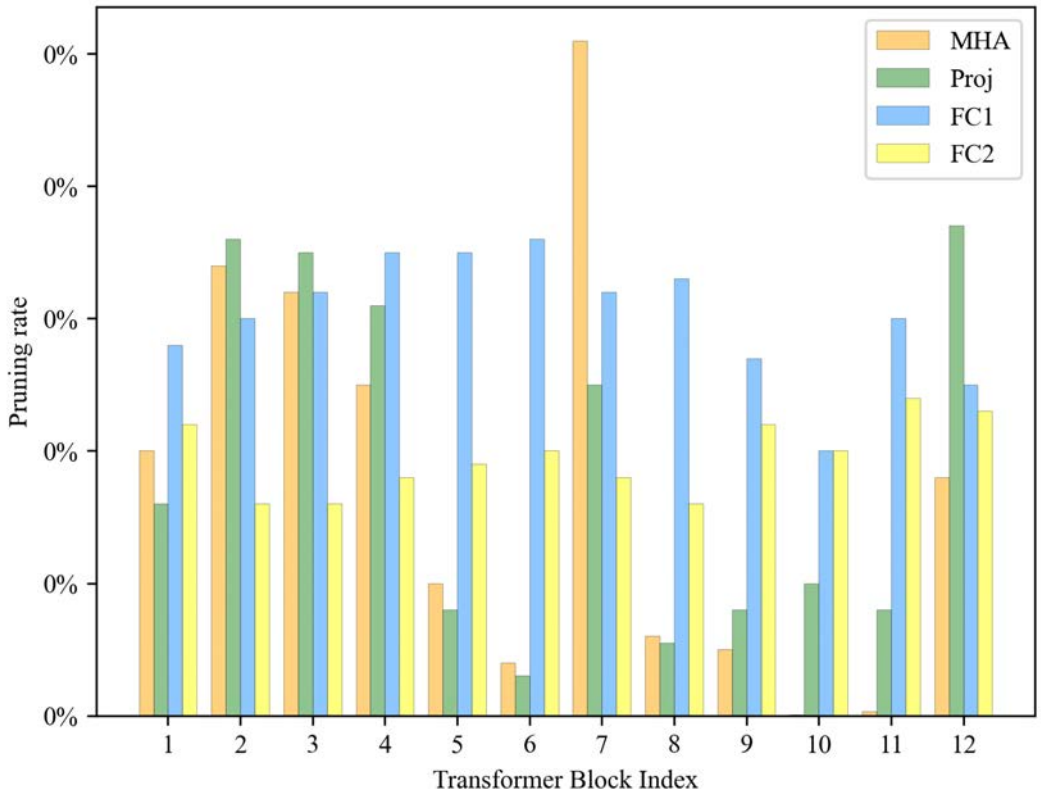


Figure 5.5: The pruning rate of units on each block when the pruning rate is set at 0.3 for DeiT-S.

compare it with both random pruning and uniform pruning. In our method, the number of pruned units for each individual layer is determined adaptively according to the global budget. The top-1 accuracy of uniform pruning is decreased by 4.05%. We also apply the random pruning to the DeiT-S, which also achieves an inferior performance. Lastly, our proposed X-Pruner outperforms these two methods with minor top-1 accuracy drop (0.87%).

We visualize the layer-wise pruning rate for the DeiT-S in Fig. 5.5. We observe that our method automatically learns the pruned architecture by taking into account the explainability-aware mask values, which is superior to estimating the importance of individual prunable units. Moreover, by visualizing the attention maps produced by the 4-th layer in DeiT-B model in Fig. 5.6, we observe that the proposed X-Pruner indeed removes the redundant heads that mainly focus on background and contribute less to the final prediction.

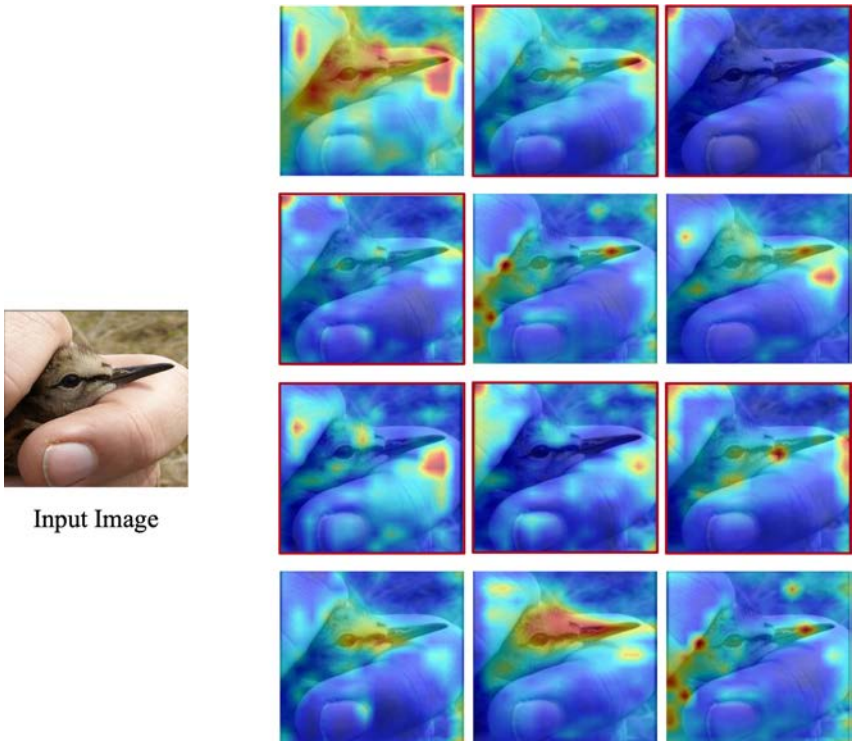


Figure 5.6: Visualization of the attention maps produced by the 4-th layer for DeiT-B. Red box means the head is pruned based on our learned mask values.

5.5 Conclusion

We proposed the X-Pruner, a novel explainable transformer pruning framework. In X-Pruner, a novel explainability-aware mask is proposed to evaluate each prunable unit's contribution to predicting every class, which is fully differentiable and learned with a proposed class-wise regularizer to mitigate over-fitting. Then, a new explainable pruning process was introduced to learn layer-wise pruning rate until a resource constraint is reached. Extensive experiments demonstrate that the X-Pruner is able to significantly reduce the computational costs of several transformers in terms of model explainability. Moreover, it surpasses the state-of-the-art pruning methods with a minor accuracy drop.

Chapter 6

Improving Sample Quality in Generative Models via Explainable Techniques

In this chapter, we focus on evaluating and improving the quality of samples generated by generative models based on visual explanations. Given the overwhelming popularity of text prompting in numerous Generative AI scenarios, how to effectively support such inputs with explanation and guidance presents a significant challenge. The application of explainable methods to enhance the quality of samples generated by generative models establishes a crucial link between interpretability and performance improvement. To address this challenge, we undertake a comprehensive analysis of the factors shaping sample generation. This exploration serves as the foundation for our pioneering solution: the delta module. This module is introduced as a means to refine the generation process, leveraging attention maps derived from an explainable method. Then, this chapter describes the multi-view score consistency method to enable 3D editing by use of a diffusion prior, which is effective in providing additional supervision signals for learning 3D-consistent geometry. By capitalizing on these insights, our framework is empowered to optimize sample generation, enhancing both its coherence and fidelity. Lastly, experimental results on various real-world datasets are presented to illustrate the efficacy of the proposed framework across a range of text prompts.

6.1 Introduction

With the growing prevalence of efficient 3D reconstruction techniques like Neural Radiance Fields (NeRFs), generating photo-realistic synthetic views of real-world 3D scenes has become increasingly feasible [167]. Meanwhile, there is a surging demand for the manipulation of 3D scenes driven by the broad range of content re-creation use

cases [168]. Recent research endeavors have introduced various methods to enhance the capabilities of NeRFs for editing purposes. These approaches include operating on explicit 3D representations [169], training models to enable color modifications or the removal of certain objects [170], or utilizing diffusion models to perform text-to-3D generation [171]. These advancements have significantly expanded the versatility of implicit volumetric representations. However, it has been observed that the edited scenes produced by those methods may not always follow the semantic meanings of the provided prompt accurately.

We observe two key issues in state-of-the-art methods for editing NeRF scenes. (1) "Over-editing", where excessive or unnecessary alterations of the prompt are generated; and (2) "inconsistent editing", where the large inconsistencies within 2D edits lead to the model's failure to consolidate in 3D. Illustrative cases demonstrating the aforementioned issues are provided in Figure 2, where the edited scenes are generated employing the state-of-the-art IN2N [172] method. In the left column, we present an example of over-editing, a scenario in which the model erroneously associates the "blue" attribute with the wrong subject. In the right column, we provide a demonstration of the inconsistent editing, where the model produces renderings that exhibit inconsistencies when viewed from different perspectives.

Recently, diffusion models have been proposed as a promising approach to generate multi-view consistent images [173]. Notably, the diffusion process aims at producing high-fidelity images through successive denoising steps applied to images initialized with Gaussian noise. This method effectively addresses prevailing limitations of traditional generative models, marking a notable advancement in image generation techniques. Editing 3D scenes using diffusion models can be achieved through two primary methods: either by optimizing NeRFs via a pretrained diffusion model such as DreamFusion [171], or via employing an unconditional generative diffusion model that can be trained with 2D images [174]. Although these approaches can generate 3D models from any given text prompts, they currently lack fine-grained control over the synthesized views. More importantly, they cannot be directly used to edit real-captured NeRFs of fully observed 3D scenes. Haque et al. [172] propose Instruct-NeRF2NeRF, which extracts shape and appearance features from a pretrained 2D diffusion model (i.e., InstructPix2Pix [175]) to gradually edit the rendered images while optimizing the underlying 3D scene. This method, however, shares several limitations with InstructPix2Pix [175], such as significant multi-view inconsistencies and notable rendering artifacts including noise and blurring. Although recent diffusion variants show improved performances, the diffusion process still makes mistakes, such as unrealistic images, artifacts, biases, and drops for required concepts [175]. Therefore, the decision-making process of the diffusion model should be interpretable to trust the

outcomes of the algorithm. However, existing explainable methods mostly specialize in classifiers. How to interpret the diffusion process in terms of generated visual concepts and attended regions at each time step remains unexplored.

Our method. To solve the aforementioned problems and perform high-fidelity text-to-3D editing with pretrained NeRFs, we propose a framework dubbed Edit-DiffNeRF to optimize a pretrained diffusion model based on attention maps generated by an explainable method to controllable and 3D-consistent scene edits. Which enables us to make fine-grained modifications to the rendered views of NeRF, offering enhanced control and accuracy in the optimization process of an underlying scene. Our Edit-DiffNeRF is composed of a frozen diffusion model, a proposed delta module to edit the latent space of a diffusion model, and a NeRF. Our proposed framework for text-to-3D editing involves two key steps. First, in order to perform the editing for the rendered views, we utilize the diffusion prior to generate a latent semantic embedding for each view and then apply our proposed delta module to edit the embedding. This delta module is optimized using a CLIP distance loss function. After training, it is able to produce edited images based on the input text instruction. Second, we proceed to train the NeRF using those edited images, leveraging the modifications made through the delta module. In order to ensure the multi-view consistency of a 3D scene, we propose a multi-view semantic consistency loss to reconstruct consistent latent features in the latent space from different views. To evaluate its effectiveness, we evaluate our Edit-DiffNeRF on a variety of real-captured NeRF scenes published by [172]. Extensive experimental results demonstrate 25% improvement in the alignment of the performed 3D edits with the text instructions compared to Instruct-NeRF2NeRF [172].

6.2 Related Work

6.2.1 Neural 2D & 3D Scene Editing

Recent advances in neural networks have opened new frontiers in content editing, providing users with a range of user-friendly editing options. These include the manipulation of facial attributes [176]–[179], stroke-based editing techniques [180], [181], or style transfer between images [182]. However, similar capabilities in editing 3D scenes represented by NeRF are still limited. Existing methods either depend on meticulous manual annotation [183]–[185], explicit object deformation [186], [187], or global style transfer [188]. The advent of 3D Generative Adversarial Networks (GANs) [189], [190] and semantic NeRF editing [191]–[193] have marked significant progress in the field. For example, EditNeRF [170] enables modifications to be made to both the shape and color of objects in a scene. CLIP-NeRF [168] and NeRF-Art [194] further extend this by

promoting alignment between the CLIP representations of a scene and an input text prompt. Nonetheless, these methods mainly focus on texture or color modification. Blended-NeRF [195] performs localized editing in a NeRF along with a predefined 3D region of interest (ROI) box. Despite its effectiveness, the rigid box shape of the 3D ROI can sometimes be restrictive. SINE [192] requires one image from a scene edited by the user and edits a NeRF with a mesh for geometric supervision. IN2N [172] is another method for editing NeRFs with text prompts based on a pre-trained model IP2P [175]. Although promising, the edited scenes suffer from excessive or unnecessary alterations and view inconsistency. Moreover, these methods cannot assess or evaluate the quality of generated samples and identify areas for improvement. In this work, we focus on view consistent editing and the manipulation of the editing direction that aligns closely with the text prompt.

6.2.2 NeRF 3D generation

The recent advances in pre-trained large-scale models have significantly accelerated the field of 3D content generation from scratch, allowing for fast and effective creation of 3D scenes. Some works optimize NeRFs by vision-language models such as CLIP [196]. One such method is Dream Fields [197], which leverages multi-modal image and text representations from CLIP to train NeRFs for synthesizing 3D objects. CLIP-Forge [198] uses a two-stage training process based on an unlabeled shape dataset and a pre-trained image-text model CLIP to generate 3D shapes. CLIP has also been used in the recent state-of-the-art (SOTA) model DreamFusion [171] and its variants [199]–[201] to generate 3D contents. While these SOTA models are able to produce 3D scenes based on arbitrary text inputs, they often grapple with challenges such as inadequate detailing and unrealistic outputs. To address these issues, IT3D [202] utilizes powerful text-to-image diffusion models to generate high-quality 3D scenes. In parallel, recent works such as SparseFusion [203] leverages a view-conditioned latent diffusion model to distill a 3D consistent scene. In our work, we aim to edit real-world 3D scenes by a pre-trained 2D diffusion prior. Furthermore, we propose a novel multi-view score distillation method to ensure 3D consistency across different viewpoints of a scene.

6.3 Preliminaries

6.3.1 Denoising diffusion probabilistic models (DDPMs)

Given a set of training views $\{\boldsymbol{x}^i\}_{i=1}^N \in \mathcal{I}$ for a 3D scene, The goal of generative models, e.g., Denoising Diffusion Probabilistic Models (DDPMs) [204] is to optimize the

parameters θ of a model that closely approximates the data distribution $p(\boldsymbol{x})$. DDPM proposes to learn the data distribution by gradually transforming a sample from a tractable noise distribution toward a target distribution. Diffusion models typically include a deterministic forward process $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ that gradually adds noise to the sample such that:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) := \mathcal{N}(\boldsymbol{x}_t; \sqrt{1-\beta_t}\boldsymbol{x}_{t-1}, \beta_t \boldsymbol{I}), \quad t \in (0, T], \tag{6.1}$$

where β_t is the t-th variance schedule. The model then learns the reverse (denoising) process with a neural network $\mathcal{D}_\theta(\boldsymbol{x}_t, t)$, which performs denoising steps by progressively removing noise and predicts $\hat{\boldsymbol{x}}_0$ from $\boldsymbol{x}_t$. The denoising process similarly uses a Gaussian distribution:

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) := \mathcal{N}(\boldsymbol{x}_{t-1}; \mu_\theta(\boldsymbol{x}_t, t), \Sigma_\theta(\boldsymbol{x}_t, t)), \tag{6.2}$$

where μ_θ and Σ_θ are the mean and variance, respectively.

6.3.2 Latent diffusion models

Latent diffusion models [173] obtain efficiency improvements compared to DDPMs [204] by leveraging the latent space of a pretrained variational autoencoder. In particular, given an input image $\boldsymbol{x}^i$, the forward process adds noise to the encoded latent embedding $\boldsymbol{z} = \boldsymbol{\varepsilon}(\boldsymbol{x}^i)$, where $\boldsymbol{\varepsilon}(\cdot)$ is an encoder and $\boldsymbol{z}_t$ is the noisy latent embedding at timestep t. The neural network $\mathcal{D}_\theta(\cdot)$ is optimized to predict the presented noise based on image and text instruction conditioning inputs. Formally, the latent diffusion objective is expressed as follows:

$$\mathcal{L} = \mathbb{E}_{\boldsymbol{\varepsilon}(\boldsymbol{x}^i), \boldsymbol{\varepsilon}(\boldsymbol{C}_I), \boldsymbol{c}_T, \epsilon\sim\mathcal{N}(0,1), t}\left[||\epsilon - \mathcal{D}_\theta(\boldsymbol{z}_t, t, \boldsymbol{\varepsilon}(\boldsymbol{C}_I), \boldsymbol{c}_T)||_2^2\right], \tag{6.3}$$

where $\boldsymbol{C}_I$ is the conditioned image, $\boldsymbol{C}_T$ is the text editing instruction, and $\hat{\epsilon}_t = \mathcal{D}_\theta(\boldsymbol{z}_t, t, \boldsymbol{\varepsilon}(\boldsymbol{C}_I), \boldsymbol{c}_T)$ is the predicted noise at timestep t. Once trained, the estimated latent $\hat{\boldsymbol{z}}_{t-1}$ can be derived with a noisy input $\boldsymbol{z}_t$ and a predicted noise $\hat{\epsilon}_t$ at timestep t.

6.4 Method

Given a pretrained NeRF scene along with the input text instruction, we aim to edit the NeRF scene through controlled manipulation of a pretrained diffusion model as a score function estimator. Which allows us to produce an edited version of the NeRF scene in accordance with the provided edit instruction. At the core of our method lies

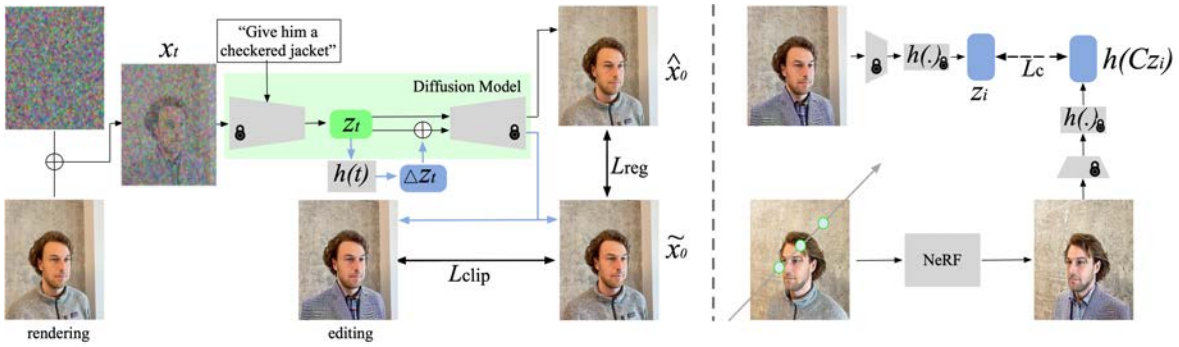


Figure 6.1: Our pipeline of Edit-DiffNeRF, which is a two-stage framework consisting of a frozen diffusion model, a proposed delta module, and a NeRF. In the first stage, we train the delta module $h(t)$ to edit the latent space of a pretrained diffusion model. After training, it is able to produce edited images based on the input text instruction. Then we freeze the weights of the delta module and train the NeRF using those edited images, leveraging the modifications made through the delta module.

the concept of semantic-aware diffusion, wherein we gradually guide the noised latent code at each timestep denoted by t towards a more semantically faithful and consistent generation.

We describe the calculation of the relevancy score in Section 6.4.1. The relevancy score is to determine subject tokens in the text prompt, and is used as a guidance in Section 6.4.2 to strengthen the activations of the neglected subject tokens at different timesteps. In Section 6.4.3, we introduce our view-consistent rendering to allow similar edit localization for 3D scene editing.

6.4.1 Computing 2D Score on 3D Scene

Given a source model with parameters θ_{src}, which is an implicit representation of a source 3D scene, our goal is to model and convert θ_{src} to θ_{tgt} along with the target text prompt.

Given an image, denoted as I belonging to an image space $\mathcal{I}$ representing a scene, and a text editing instruction $\boldsymbol{C}_T$, we employ the IP2P framework to calculate the relevancy score associated with each token present within $\boldsymbol{C}_T$. This score serves as an indicator that a pixel should be changed based on the specific token. First, we obtain the noisy latent z_t with noise level t:

$$z_t = \sqrt{\boldsymbol{\alpha_t}}\boldsymbol{\varepsilon}(I) + \sqrt{1-\boldsymbol{\alpha_t}}\boldsymbol{\epsilon}, \tag{6.4}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(0,1)$ is a random noise sample, α_t is a pre-defined variable that controls the noise schedule. Subsequently, we employ IP2P's noise approximation model denoted as ϵ_θ to yield two different predictions: 1) the predicted noised $\epsilon_{\boldsymbol{I},\boldsymbol{C}_T}(z_t) = \epsilon_\theta(z_t, t, \boldsymbol{I}, \boldsymbol{C}_T)$, and 2) the predicted noise under the absence of the specific token $\boldsymbol{C}_k$,

$\epsilon_{\boldsymbol{I},\boldsymbol{C}_T\setminus\boldsymbol{C}_k}(z_t) = \epsilon_\theta(z_t, t, \boldsymbol{I}, \boldsymbol{C}_T\setminus\boldsymbol{C}_k)$. Notably, the key difference between $\epsilon_{\boldsymbol{I},\boldsymbol{C}_T}$ and $\epsilon_{\boldsymbol{I},\boldsymbol{C}_T\setminus\boldsymbol{C}_k}$ lies in the fact that only the former is aware of the token $\boldsymbol{C}_k$. We then compute the relevancy score for all tokens $\boldsymbol{C}_1, \boldsymbol{C}_2, ..., \boldsymbol{C}_K$.

Furthermore, an intuitive idea to edit $\boldsymbol{x}$ is simply updating $\boldsymbol{x}_t$ to optimize NeRF based on the score distillation sampling loss $\nabla\mathcal{L}_{SDS}$ as proposed in DreamFusion [171]. However, it results in a 3D scene with more artifacts or distorted views. We believe this is due to a pretrained diffusion model with the fixed semantic latent space may not achieve feasible results in realistic scenarios [205].

6.4.2 Editing semantic latent space in diffusion models

In order to manipulate the latent semantic space with the input text instruction and circumvent the process of training an entire diffusion model, we propose to utilize a delta module $\boldsymbol{h}(t)$, which learns the shifted latent semantic space Δz_t given a frozen and pretrained diffusion model and offers significantly reduced computational demands. $\boldsymbol{h}(t)$ is implemented as a small compact neural network with two convolutional layers. Which have the same number of channels as the bottleneck layer of the U-Net in the diffusion model. Formally, the parameterization for $\mu_\theta(\boldsymbol{x}_t, t)$ given the delta module $h(\cdot)$ becomes:

$$\mu_\theta(\boldsymbol{x}_t, t, \Delta z_t) = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\mathcal{D}_\theta(z_t|\Delta z_t, t, \boldsymbol{\varepsilon}(\boldsymbol{C}_I), \boldsymbol{c}_T)\right). \tag{6.5}$$

Essentially, by introducing the information of $\boldsymbol{c}_T$ to the latent semantic space with Δz_t, the predicted $\hat{\epsilon} = \mathcal{D}_\theta(z_t|\Delta z_t, t, \boldsymbol{\varepsilon}(\boldsymbol{C}_I), \boldsymbol{c}_T)$ is modified. Which, in turn, produces the shifted mean value $\mu_\theta(\boldsymbol{x}_t, t, \Delta z_t)$ to bridge the gap and facilitate the reverse process of reconstructing the provided instructional information in the samples.

We utilize the architecture of CLIP [206], which comprises an image encoder $\boldsymbol{\varepsilon}_I$ and a text encoder $\boldsymbol{\varepsilon}_T$ that project inputs to a shared latent space. Building on this, we propose a cross-modal CLIP distance function to evalute the cosine similarity between the input text instruction and the edited image:

$$\mathcal{L}_{\text{clip}} = 1 - \left\langle \boldsymbol{\varepsilon}_T(\boldsymbol{t}_{\text{src}}) - \boldsymbol{\varepsilon}_I(\boldsymbol{t}_{\text{tgt}}), \boldsymbol{\varepsilon}_I(\boldsymbol{x}_{\text{src}}) - \boldsymbol{\varepsilon}_I(\boldsymbol{x}_{\text{tgt}}) \right\rangle, \tag{6.6}$$

where the input image and its text description are represented by $\boldsymbol{x}_{\text{src}}$ and $\boldsymbol{t}_{\text{src}}$. The text instruction and the edited image are denoted as $\boldsymbol{t}_{tgt}$ and $\boldsymbol{x}_{tgt}$, respectively, and $\langle\cdot\rangle$ is the cosine similarity operator. In addition, we add an L1 loss to regulate the produced $\boldsymbol{x}_0$ from the original input and the edited one by:

$$\mathcal{L}_{\text{reg}} = \lambda_{\text{reg}}|\hat{\boldsymbol{x}}_0 - \tilde{\boldsymbol{x}}_0|, \tag{6.7}$$

where $\hat{x_0}$ is obtained via the frozen diffusion prior, $\widetilde{x}_0$ is generated with the modified latent embedding, and λ_{reg} is the hyper-parameter.

6.4.3 View-consistent rendering

Another fundamental challenge when it comes to editing a 3D scene with a 2D diffusion model is that the diffusion models lack 3D awareness, which leads to a generated NeRF with inconsistent and distorted views. Seo et al. [207] tried to close this gap by utilizing viewpoint-specific depth maps from a coarse 3D structure. However, this embedding needs to optimize a 3D model for each text prompt. This is a computationally intensive process that can still produce blurred and distorted images despite optimization.

To account for the inconsistency challenge, we propose to encode the latent semantic embedding for each view using the pretrained diffusion model and our proposed delta function $\boldsymbol{h}(\cdot)$. Specifically, given a pretrained diffusion model and the optimized $\boldsymbol{h}(\cdot)$, each rendered image is encoded into a latent embedding $\boldsymbol{z}_i$. Inspired by conditional NeRFs [168], the color $\boldsymbol{c}$ in NeRF [167] is extended to a latent-dependent emitted color $\boldsymbol{c}_z$:

$$(\sigma, \boldsymbol{c}_{z_i}) = F_\theta(\boldsymbol{x}, \boldsymbol{d}, \boldsymbol{z}_i), \tag{6.8}$$

where the embedding $\boldsymbol{z}_i$ is a conditional input, and F_θ is the NeRF model.

6.4.4 Multi-view semantic consistency loss

To achieve our goal, the optimized delta module $\boldsymbol{h}(\cdot)$ is supposed to produce consistent latent semantic embeddings as much as possible across different camera poses. Therefore, we propose a novel multi-view semantic consistency loss, denoted as $\mathcal{L}_c$. Which involves using a latent embedding $\boldsymbol{z}_i$ extracted from image $\boldsymbol{x}^i$ as the conditional input to reconstruct images from different views. The formal expression of our proposed loss $\mathcal{L}_c$ is given as follows:

$$\mathcal{L}_c = ||\boldsymbol{h}(\boldsymbol{C}_{z_i}) - \boldsymbol{z}_i||_1, \tag{6.9}$$

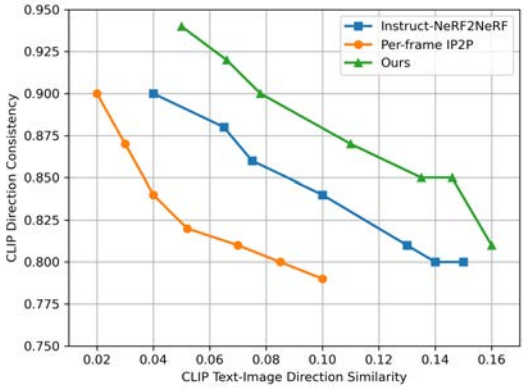
where $\boldsymbol{C}_{z_i}$ is the rendered output based on Eq. 6.8. Thus the total loss for training a NeRF becomes:

$$\mathcal{L} = \mathcal{L}_{\text{photo}} + \lambda_c \mathcal{L}_c, \tag{6.10}$$

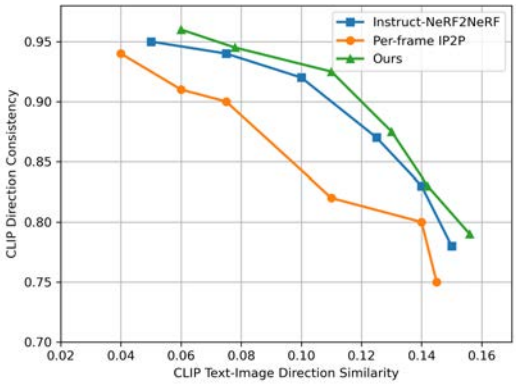
where λ_c is the hyper-parameter. By leveraging this strategy, our method ensures consistency across different views while also preserving the integrity of the underlying latent embeddings.

6.5 Experiments

In this section, we first introduce the datasets and implementation details in Section 6.5.1. Subsequently, we evaluate the performance of our approach against other methods in Section 6.5.2. Finally, to further understand the impact of the key designs, we conduct an ablation study in Section 6.5.3.



(a) Results with larger training datasets



(b) Results with smaller training datasets (less than 100 images)

Figure 6.2: We plot the trade-off between the CLIP Direction Consistency and the CLIP Text-Image Direction Similarity. For both metrics, higher is better.

Table 6.1: Quantiative evaluation on real-captured scenes

	CLIP Text-Image Direction Similarity	CLIP Direction Consistency
Per-frame IP2P [175]	0.1603	0.8185
SDS w/ IP2P [175]	0.0266	0.9160
One-time DU [172]	0.1157	0.8823
Instruct-NeRF2NeRF [172]	0.1600	0.9191
Ours	**0.2031**	**0.9376**

6.5.1 Experimental setup

Datasets. We conduct 3D editing on a set of scenes with varying degrees of complexity, including 360-degree scenes of environments and objects, faces, and full-body portraits that are released by [172]. These scenes were captured using two types of cameras: a smartphone and a mirrorless camera. We use the camera poses that were extracted via the COLMAP [208]. Following CLIP-NeRF [168], we also evaluate the effectiveness of our approach on two publicly available datasets: Photoshape [209] with a collection of 150K chairs and Carla [210], [211] consisting of 10K cars.

Table 6.2: FID scores on real-captured scenes from Instruct-NeRF2NeRF [172]

	Chairs		Cars	
	Before	After	Before	After
EditNeRF [170]	36.8	40.2	102.8	118.7
CLIP-NeRF [168]	47.8	48.4	66.7	67.8
Ours	**32.1**	**32.5**	**48.6**	**49.0**

Implementation details. We choose the official InstructPix2Pix [175] as our diffusion prior, which contains a large-scale text-to-image latent diffusion model StableDiffusion [173]. During the training stage, we uniformly sample timesteps ranging from $t = 1$ to $T = 1000$ for all experiments. The variances of the diffusion process are linearly increased, starting from $\beta_1 = 0.00085$ and reaching $\beta_1 = 0.012$. We optimize our proposed delta module $\boldsymbol{h}(t)$ using 50 steps for each view. The training process requires approximately 15 minutes and is performed on four RTX 3090 GPUs. After training, we use the edited images as the supervision to train our NeRF. As the underlying NeRF implementation, we use the nerfacto model from NeRFStudio [212], which is a recommended real-time model tuned for real captures. We follow the training strategy in NeRFStudio and the NeRFs are optimized for 30000 steps with L1 and directional CLIP losses in [206].

Metrics. It should be noted that unlike dynamic NeRF methods, acquiring ground truth views for view synthesis results after editing poses significant challenges, particularly when performing with real scenes. This is primarily because the edited views as a product of user manipulation do not physically exist. Following the evaluation metrics employed in Instruct-NeRF2NeRF [172], we evaluate two crucial quantitative metrics, namely (1) CLIP Text-Image Direction Similarity, i.e., the alignment between the edited 3D views and the corresponding text instruction, and (2) CLIP Direction Consistency, the temporal consistency of the edit across multiple views [175]. Besides, we compute the Fréchet Inception Distance (FID) scores [213] for 2000 rendered images before and after the editing process, which allows us to quantitatively assess the quality and fidelity of the edited scenes.

6.5.2 Quantiative evaluation

Tables 6.1 and 6.2 show the superiority of Edit-DiffNeRF over other state-of-the-art methods on real scenes. We observe that by learning the optimal delta module to edit the latent semantic space, our Edit-DiffNeRF can have notably higher CLIP Text-Image Direction Similarity and Consistency. In Fig. 6.2, we also plot the trade-off between the CLIP Direction Consistency and the CLIP Text-Image Direction Similarity

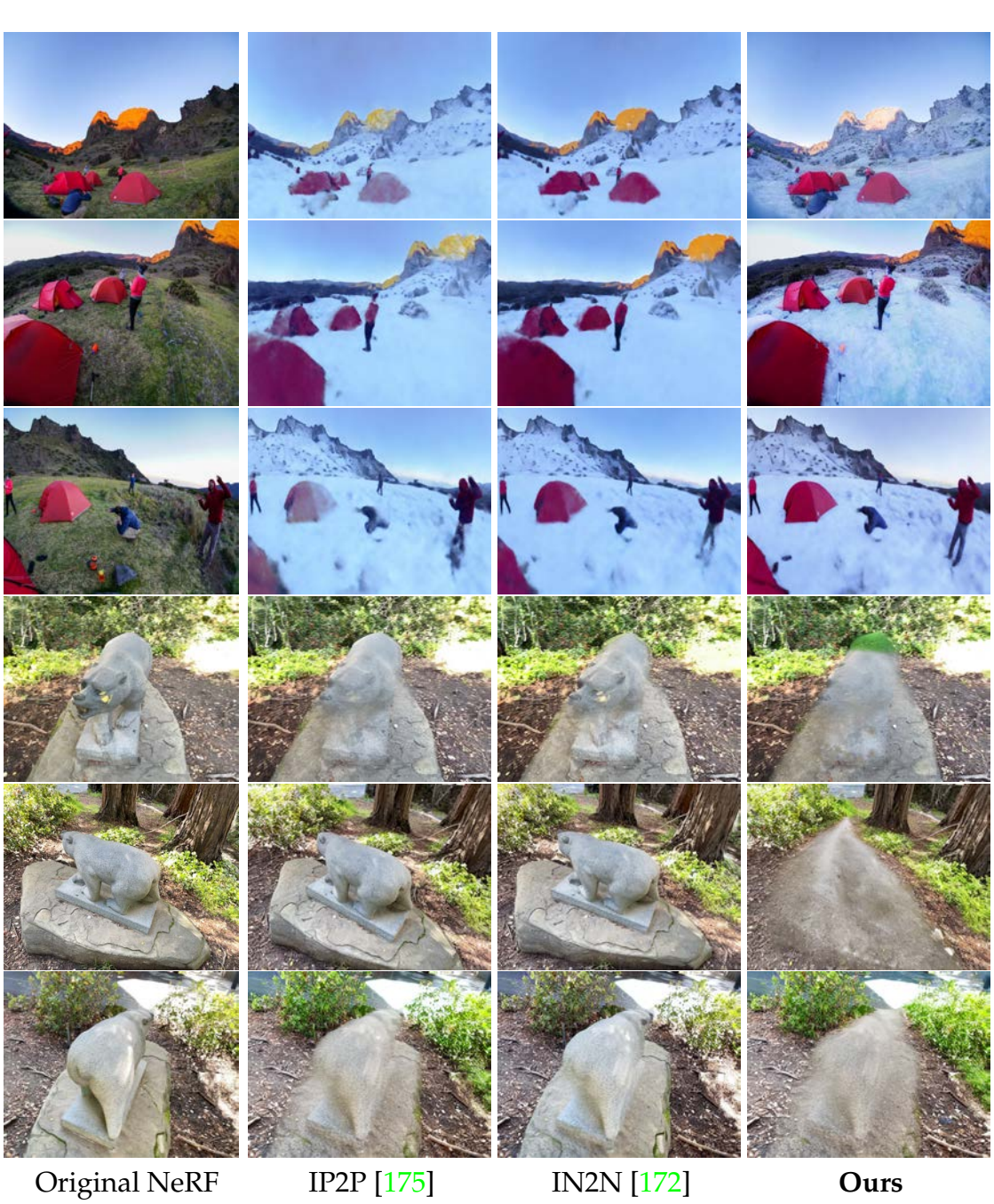


Figure 6.3: Visual comparisons with a collection of recent state-of-the-art methods.

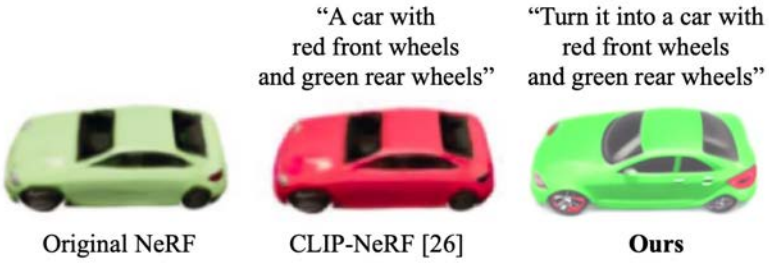


Figure 6.4: Comparisons of editing results between CLIP-NeRF [168] and our Edit-DiffNeRF.

over two scenes. As these two metrics compete with each other, when the degree to which the output images align with the desired edit increases, the consistency with the input image decreases. As is observed from Fig. 6.2, compared to the recent state-of-the-art method Instruct-NeRF2NeRF [172], the CLIP Direction consistency obtained by our Edit-DiffNeRF is significantly higher, even with similar CLIP Text-Image Direction Similarity values. Furthermore, we observe that when the training dataset for a scene is smaller (consisting of less than 100 images), both the Instruct-NeRF2NeRF [172] and our method yield similar results, with lower directional similarity.

In Table 6.2, we report the FID scores for measuring the image quality of synthesized views before and after editing. To calculate the FID scores for rendered images, we employ a set of 2000 randomly selected test images. Subsequently, we apply various edit instructions similar to CLIP-NeRF [168] to these images and recompute the FID scores for the edited results. On the chair dataset, When evaluated on the chair dataset, EditNeRF [170] demonstrates improved performance in terms of reconstruction compared to CLIP-NeRF [168]. However, it is worth noting that the quality of the edited images noticeably decreases after the editing process. When evaluated on the car dataset, CLIP-NeRF [168] exhibits a significant improvement over EditNeRF [170] in terms of reconstruction quality before and after editing. Finally, compared to those two methods, our Edit-DiffNeRF not only greatly improves the overall quality but also effectively preserves the image quality after the editing process.

Editing results. We show edited results rendered from different views in Fig. 6.3 for real-captured scenes. For comparison, we also show the original NeRF rendering results under the same views before editing. In Fig. 6.3, the first set is a campsite scene. We edited it by a text instruction "Make it look like just snowed". On the one hand, as is observed from Fig. 6.3, the generated results via the InstructPix2Pix [175] are with ambiguity on what exactly to edit and exhibit considerable inconsistencies across different views. On the other hand, although the Instruct-NeRF2NeRF [172] appears to produce feasible views, some of them tend to show significant variance, resulting in a 3D scene that is blurry and highly distorted. Instead, the rendered multiple views obtained via our Edit-DiffNeRF show its capability to effectively edit real-world scenes,

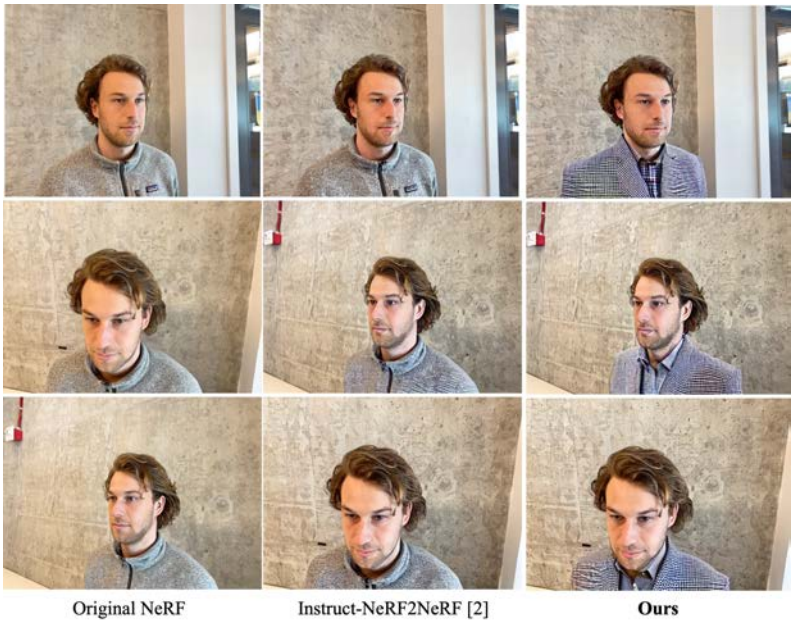


Figure 6.5: Comparison with Instruct-NeRF2NeRF [172]. Edits were performed with a text instruction "Give him a checkered jacket".

surpassing the achievements of previous approaches by delivering photo-realistic results while maintaining 3D consistency. In the second set of Fig. 6.3, the images were edited by instruction "Delete the bear statue and the stage". According to the figure, neither InstructPix2Pix [175] nor Instruct-NeRF2NeRF [172] are able to obtain the intended editing results effectively. This limitation primarily arises from the incapacity of InstructPix2Pix to handle large spatial manipulations.

Moreover, we present additional experimental results along with NeRF rendering results in Figs. 6.4 and 6.5 to further illustrate our findings. The comparisons in Fig. 6.4 were performed with a text instruction “Turn it into a car with red front wheels and green rear wheels”. It can be noted that CLIP-NeRF [168] is not able to handle fine-grained edits. Instead, our edited result (right column) is quite consistent with the instruction. In Fig. 6.5, we can further observe the presence of inconsistent edits within Instruct-NeRF2NeRF [172] that fail to consolidate in a 3D scene (middle column). Instead, our model provides a superior solution (right column), rendering consistent results and allowing for significant visual manipulations.

6.5.3 Ablation study

We validate the effectiveness of our Edit-DiffNeRF by conducting a comprehensive comparative analysis between our approach and several other variants.

Impact of delta module. We first compare our model with against Instruct-NeRF2NeRF [172]. We use the official code released by [175] and fine-tune the entire UNet model

Table 6.3: Ablation study results

	CLIP Text-Image Direction Similarity	CLIP Direction Consistency
Instruct-NeRF2NeRF [172]	0.1120	0.7805
Ours w/o $\mathcal{L}_c$	0.1703	0.9198
Ours	**0.2031**	**0.9376**

in InstructPix2Pix [175] to edit real images. Table 6.3 demonstrates that our proposed Edit-DiffNeRF outperforms InstructPix2Pix in all aspects. We attribute this superiority to the fact that fine-tuning the entire model for each scene is challenging, thereby resulting in inferior results.

Impact of multi-view semantic consistency loss. In addition, we perform experiments where we exclude the consistency loss $\mathcal{L}_c$. As is observed from Table 6.3, even in the absence of the multi-view semantic consistency loss $\mathcal{L}_c$, our method still surpasses Instruct-NeRF2NeRF [172] with a trained delta module. Nevertheless, without this loss, our method still lacks 3D consistency and only achieves a slight performance gain. In contrast, incorporating this loss yields substantial improvements in the results.

6.6 Limitations

Despite the encouraging performance gained by our work, it suffers from two limitations. First, our model is subject to the visual quality of the rendered images generated using NeRF techniques, as well as the diffusion model's ability to generalize to arbitrary edits. Second, the quality of edited images decreases when the input images have low resolution or are out of focus.

6.7 Conclusion

In this paper, we outlined the underlying challenges in achieving accurate NeRF scene modifications with pretrained 2D diffusion models. To address this limitation, we introduce the Edit-DiffNeRF framework, which specifically targets editing the semantic latent space within pretrained diffusion models. Specifically, the Edit-DiffNeRF framework is devised to learn latent semantic directions using a delta module, guided by provided text instructions, which allows for the effective consolidation of these instructions within a 3D scene through NeRF training. Furthermore, we introduce a multi-view semantic consistency loss to ensure semantic consistency across different

views. Extensive experiments demonstrate that our approach consistently and effectively enables edits across a wide range of real-captured scenes. Moreover, it significantly improves the text-image consistency of the edited results.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

This thesis presented elaborately designed and learning-based algorithms to address model explainability for the broad deployment of AI applications. Our contribution to the field includes the development of innovative visual explainable tools and empirical studies aimed at identifying unclear or inaccurate explanations within deep learning algorithms. Additionally, we investigated the role of AI explanations in facilitating the model compression process and explored their application in the scene generation domain.

In this thesis, we demonstrated the potent capabilities of deep learning models in accurately and efficiently classifying medical images. Despite the notable potential that deep learning holds, the inherent complexity of these models poses a considerable challenge in establishing the necessary trust in their decision-making processes for clinical implementation. Our contribution takes a stride towards understanding these intricate models and gaining trust through the introduction of our proposed explainable framework, including an explainable tool and a prediction basis module in Chapter 3.

In Chapter 4, we have taken crucial strides towards understanding existing intricate models and fostering humans' trust. We initiated the process by showcasing how visual attribution can serve to validate the classification decisions made by deep learning. Subsequently, we introduced an explainable model, named eX-ViT, representing our innovative approach for visually explaining weakly supervised segmentation models. Through validation on various datasets, we illustrated how our proposed model is able to provide superior comprehensive explanations for model decisions.

In Chapter 5, we illustrated the effectiveness of an explainable pruning method that we developed, showcasing its direct utility in model compression. Our explanations, coupled with empirical validation, highlight that over-parameterization in deep networks, especially in the spatial convolution layer, is not essential for achieving high

performance. A substantial number of spatial weights can be removed from the network before both training and inference. The creation of compact and accurate networks aligns seamlessly with the importance of explanations, as discussed in Chapter 1. In terms of legal relevance, compact and accurate models address the data processing principles of "data minimization" and "accuracy" outlined in the General Data Protection Regulation (GDPR). Our explainable approach offers solutions to the GDPR's requirement for transparency. The future application of compression or smaller models holds the potential to counteract the prevailing trend in deep learning toward excessively large models. Understanding which parameters are dispensable contributes to eliminating the lack of transparency in model structures.

In Chapter 6, we demonstrated how to incorporate explanations into generative models instead of discriminative models, resulting in notable improvements in the quality of generated samples. Despite the promise held by Generative AI in seamlessly delivering on-demand content throughout a user's workflow, the optimal use, methodologies, and the comparative utility of incorporating generated content over traditional approaches remain uncertain. For instance, there may be a need for explanation when specific prompts fail to yield desired generated content. A successful explanation could empower users to enhance their prompts, leading to greater satisfaction with the newly generated samples.

To sum up, this dissertation presents several novel contributions to the field of explainable AI, these contributions have been validated across multiple application domains. In the domain of medical imaging, we focused on enhancing diagnostic accuracy and interpretability in models used for Alzheimer's disease analysis. And our methods demonstrated a significant improvement in model interpretability without compromising diagnostic accuracy on ADNI dataset. In the domain of object detection, the focus was on improving the accuracy and reliability of AI models used in safety-critical applications, such as autonomous driving. Our methods were evaluated on the COCO and Pascal VOC datasets, demonstrating a significant reduction in false positives. For generative models, the thesis explored XAI techniques to enhance the quality and diversity of generated samples. Using real-captured scenes, the proposed methods resulted in a 25% increase in image quality, as measured by the Fréchet Inception Distance (FID). The integration of XAI facilitated targeted model adjustments, leading to more realistic and diverse generated outputs. It is our aspiration that the frameworks and methods introduced in this dissertation will lay a robust foundation for constructing explainable models within the computer vision community.

7.2 Future Work

In future research endeavors within the realm of explainable AI, there emerges a pressing need for the development of unified representations or data structures tailored to organize explainable features. This necessity particularly arises to effectively grapple with the challenges posed by increasingly large and intricate models. As AI models continue to expand in both size and complexity, the conventional methods for organizing and interpreting explainable features may prove insufficient. Hence, there is a call for innovative approaches that can effectively capture and communicate the intricate decision-making processes underlying these sophisticated models. Such novel representations or data structures would not only facilitate a deeper understanding of AI model behavior but also pave the way for more robust and comprehensive explanations that can be readily understood and utilized by stakeholders across various domains.

Furthermore, while our proposed methods have yielded excellent results, current exploration has primarily focused on improving visual explanations. However, the landscape of AI systems is dynamically evolving towards the integration of multi-modal capabilities. This transition introduces a significant layer of complexity, demanding a fundamental rethinking of how we conceive and implement explainability. It would be intriguing to extend the concept of explainability beyond visual modalities and explore how humans can effectively leverage multi-modal explanations to interact with AI models. In addition, in scenarios where users engage in text-based conversations with chatbots, further studies should strive to provide empirical evidence that offers tangible insights into the effectiveness of prompting strategies. By providing explanations on how humans can harness prompts to facilitate more meaningful and intuitive interactions with AI systems, future research can significantly enhance our understanding of explainability in AI environments, ultimately fostering more transparent and trustworthy AI systems.

Bibliography

[1] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[2] P. Schramowski, W. Stammer, S. Teso, A. Brugger, F. Herbert, X. Shao, H. Luigs, A. Mahlein, and K. Kersting, "Making deep neural networks right for the right scientific reasons by interacting with their explanations," *Nature Machine Intelligence*, vol. 2, no. 8, pp. 476–486, 2020. [Online]. Available: https://doi.org/10.1038/s42256-020-0212-3.

[3] E. Tjoa and C. Guan, "A survey on explainable artificial intelligence (XAI): toward medical XAI," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 11, pp. 4793–4813, 2021. [Online]. Available: https://doi.org/10.1109/TNNLS.2020.3027314.

[4] D. Gunning and D. W. Aha, "Darpa's explainable artificial intelligence (XAI) program," *AI Magazine*, vol. 40, no. 2, pp. 44–58, 2019.

[5] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter, "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 1528–1540. [Online]. Available: https://doi.org/10.1145/2976749.2978392.

[6] A. Renda, P. Ducange, F. Marcelloni, D. Sabella, M. C. Filippou, G. Nardini, G. Stea, A. Virdis, D. Micheli, D. Rapone, *et al.*, "Federated learning of explainable ai models in 6g systems: Towards secure and automated vehicle networking," *Information*, vol. 13, no. 8, p. 395, 2022.

[7] Y. Zhang and H. Yu, "Lr-xfl: Logical reasoning-based explainable federated learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, 2024, pp. 21 788–21 796.

[8] Z. Wang, H. Wang, S. Ramkumar, P. Mardziel, M. Fredrikson, and A. Datta, "Smoothed geometry for robust attribution," *Advances in neural information processing systems*, vol. 33, pp. 13 623–13 634, 2020.

[9] S. Ghalebikesabi, L. Ter-Minassian, K. DiazOrdaz, and C. C. Holmes, "On locality of local explanation models," *Advances in neural information processing systems*, vol. 34, pp. 18 395–18 407, 2021.

[10] P. Chalasani, J. Chen, A. R. Chowdhury, X. Wu, and S. Jha, "Concise explanations of neural networks using adversarial training," in *International Conference on Machine Learning*, 2020, pp. 1383–1391.

[11] A. Boopathy, S. Liu, G. Zhang, C. Liu, P.-Y. Chen, S. Chang, and L. Daniel, "Proper network interpretability helps adversarial robustness in classification," in *International Conference on Machine Learning*, 2020, pp. 1014–1023.

[12] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, 2020.

[13] R. Mihalcea and C. W. Leong, "Toward communicating simple sentences using pictorial representations," *Machine Translation*, vol. 22, no. 3, pp. 153–173, 2008. [Online]. Available: https://doi.org/10.1007/s10590-009-9050-0.

[14] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with CLIP latents," 2022. arXiv: 2204.06125. [Online]. Available: https://doi.org/10.48550/arXiv.2204.06125.

[15] R. Dale, "GPT-3: what's it good for?" *Natural Language Engineering*, vol. 27, no. 1, pp. 113–118, 2021. [Online]. Available: https://doi.org/10.1017/S1351324920000601.

[16] M. Wermelinger, "Using github copilot to solve simple programming problems," in *Proceedings of the ACM Technical Symposium on Computer Science Education*, 2023, pp. 172–178. [Online]. Available: https://doi.org/10.1145/3545945.3569830.

[17] W. F. Godoy, P. Valero-Lara, K. Teranishi, P. Balaprakash, and J. S. Vetter, "Evaluation of openai codex for HPC parallel programming models kernel generation," in *Proceedings of the International Conference on Parallel Processing*, 2023, pp. 136–144. [Online]. Available: https://doi.org/10.1145/3605731.3605886.

[18] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature machine intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[19] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 618–626.

[20] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, "Pixel-wise explanations of non-linear classifier decisions with deep taylor decomposition," in *International conference on machine learning*, PMLR, 2015, pp. 286–294.

[21] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2921–2929.

[22] M. A. Jalwana, N. Akhtar, M. Bennamoun, and A. Mian, "Cameras: Enhanced resolution and sanity preserving class activation mapping for image saliency," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 16 327–16 336.

[23] A. Ghorbani, A. Abid, and J. Zou, "Interpretation of neural networks is fragile," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, 2019, pp. 3681–3688.

[24] A. Singh, S. Sengupta, and V. Lakshminarayanan, "Explainable deep learning models in medical image analysis," *Journal of Imaging*, vol. 6, no. 6, p. 52, 2020. [Online]. Available: https://doi.org/10.3390/jimaging6060052.

[25] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the North American Chapter of the Association for Computational Linguistics*, 2019, pp. 4171–4186. [Online]. Available: https://doi.org/10.18653/v1/n19-1423.

[26] H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, and X. Hu, "Score-cam: Score-weighted visual explanations for convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 24–25.

[27] P. Hacker, A. Engel, and M. Mauer, "Regulating ChatGPT and other large generative AI models," in *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*, 2023, pp. 1112–1123.

[28] A. Zador, S. Escola, B. Richards, B. Ölveczky, Y. Bengio, K. Boahen, M. Botvinick, D. Chklovskii, A. Churchland, C. Clopath, *et al.*, "Catalyzing next-generation artificial intelligence through neuroai," *Nature communications*, vol. 14, no. 1, p. 1597, 2023.

[29] I. E. Kumar, S. Venkatasubramanian, C. Scheidegger, and S. Friedler, "Problems with shapley-value-based explanations as feature importance measures," in *Proceedings of the International conference on machine learning*, 2020, pp. 5491–5500.

[30] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee, "From local explanations to global understanding with explainable ai for trees," *Nature machine intelligence*, vol. 2, no. 1, pp. 56–67, 2020.

[31] Q. Ai and L. Narayanan. R, "Model-agnostic vs. model-intrinsic interpretability for explainable product search," in *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 2021, pp. 5–15.

[32] P. Wang and N. Vasconcelos, "A generalized explanation framework for visualization of deep learning model predictions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 8, pp. 9265–9283, 2023.

[33] A. Kumar, K. Sehgal, P. Garg, V. Kamakshi, and N. C. Krishnan, "MACE: model agnostic concept extractor for explaining image classification networks," *IEEE Transactions on Artificial Intelligence*, vol. 2, no. 6, pp. 574–583, 2021.

[34] T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," *Artificial Intelligence*, vol. 267, pp. 1–38, 2019.

[35] A. B. Arrieta, N. D. Rodríguez, J. D. Ser, A. Bennetot, S. Tabik, *et al.*, "Explainable artificial intelligence (XAI): concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, pp. 82–115, 2020.

[36] P. Voigt and A. Von dem Bussche, "The EU general data protection regulation (GDPR)," *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, vol. 10, pp. 10–5555, 2017.

[37] F. Doshi-Velez, M. Kortz, R. Budish, C. Bavitz, S. Gershman, D. O'Brien, S. Schieber, J. Waldo, D. Weinberger, and A. Wood, "Accountability of AI under the law: The role of explanation," 2017. arXiv: 1711.01134. [Online]. Available: http://arxiv.org/abs/1711.01134.

[38] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks," in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, 2018, pp. 839–847.

[39] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should I trust you?": Explaining the predictions of any classifier," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1135–1144. [Online]. Available: https://doi.org/10.1145/2939672.2939778.

[40] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in Neural Information Processing Systems*, vol. 30, pp. 4765–4774, 2017.

[41] M. T. Ribeiro, S. Singh, and C. Guestrin, "Anchors: High-precision model-agnostic explanations," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018, pp. 337–348.

[42] S. Wachter, B. Mittelstadt, and C. Russell, "Counterfactual explanations without opening the black box: Automated decisions and the gdpr," *Harvard Journal of Law & Technology*, vol. 31, no. 2, pp. 841–887, 2017.

[43] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," in *Annals of statistics*, 2001, pp. 1189–1232.

[44] C. J. Anders, L. Weber, D. Neumann, W. Samek, K. Müller, and S. Lapuschkin, "Finding and removing clever hans: Using explanation methods to debug and improve deep models," *Information Fusion*, vol. 77, pp. 261–295, 2022.

[45] Y. Fei, L. Cui, S. Yang, W. Lam, Z. Lan, and S. Shi, "Enhancing grammatical error correction systems with explanations," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, 2023, pp. 7489–7501.

[46] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proceedings of the 34th International Conference on Machine Learning*, 2017, pp. 3319–3328.

[47] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," *arXiv preprint arXiv:1704.02685*, 2017.

[48] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, *et al.*, "Sanity checks for saliency maps," *Advances in Neural Information Processing Systems*, vol. 31, pp. 9505–9515, 2018.

[49] A. Malhotra, S. Mittal, P. Majumdar, S. Chhabra, K. Thakral, M. Vatsa, R. Singh, S. Chaudhury, A. Pudrod, and A. Agrawal, "Multi-task driven explainable diagnosis of COVID-19 using chest X-ray images," *Pattern Recognition*, vol. 122, pp. 1–13, 2022.

[50] Z. C. Lipton, "The mythos of model interpretability," *Communications of the ACM*, vol. 61, no. 10, pp. 36–43, 2018.

[51] M. Zhou, X. Wei, W. Jia, and S. Kwong, "Joint decision tree and visual feature rate control optimization for VVC UHD coding," *IEEE Transactions on Image Processing*, vol. 32, pp. 219–234, 2023.

[52] H. Chefer, S. Gur, and L. Wolf, "Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers," in *Proceedings of the IEEE International Conference on Computer Vision*, 2021, pp. 387–396.

[53] A. Shaban-Nejad, M. Michalowski, J. S. Brownstein, and D. L. Buckeridge, "Guest editorial explainable AI: towards fairness, accountability, transparency and trust in healthcare," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 7, pp. 2374–2375, 2021.

[54] X. Kong, S. Liu, and L. Zhu, "Toward human-centered xai in practice: A survey," *Machine Intelligence Research*, pp. 1–31, 2024.

[55] J. Colin, T. Fel, R. Cadène, and T. Serre, "What I cannot predict, i do not understand: A human-centered evaluation framework for explainability methods," in *Advances in neural information processing systems*, 2022, pp. 2832–2845.

[56] F. Doshi-Velez, M. Kortz, R. Budish, C. Bavitz, S. Gershman, D. O'Brien, K. Scott, S. Schieber, J. Waldo, D. Weinberger, *et al.*, "Accountability of ai under the law: The role of explanation," *arXiv preprint arXiv:1711.01134*, 2017.

[57] F. Doshi-Velez and B. Kim, "Towards a rigorous science of interpretable machine learning," *arXiv preprint arXiv:1702.08608*, 2017.

[58] C. Agarwal, S. Krishna, E. Saxena, M. Pawelczyk, N. Johnson, I. Puri, M. Zitnik, and H. Lakkaraju, "Openxai: Towards a transparent evaluation of model explanations," in *Advances in Neural Information Processing Systems*, 2022, pp. 15 784–15 799.

[59] Y. Liu, H. Li, Y. Guo, C. Kong, J. Li, and S. Wang, "Rethinking attention-model explainability through faithfulness violation test," in *International Conference on Machine Learning*, 2022, pp. 13 807–13 824.

[60] W. Huang, X. Zhao, G. Jin, and X. Huang, "Safari: Versatile and efficient evaluations for robustness of interpretability," in *Proceedings of the IEEE International Conference on Computer Vision*, 2023, pp. 1988–1998.

[61] R. Machlev, M. Perl, J. Belikov, K. Y. Levy, and Y. Levron, "Measuring explainability and trustworthiness of power quality disturbances classifiers using XAI—explainable artificial intelligence," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 8, pp. 5127–5137, 2021.

[62] J. van der Waa, E. Nieuwburg, A. Cremers, and M. Neerincx, "Evaluating XAI: A comparison of rule-based and example-based explanations," *Artificial intelligence*, vol. 291, p. 103 404, 2021.

[63] Q. V. Liao, Y. Zhang, R. Luss, F. Doshi-Velez, and A. Dhurandhar, "Connecting algorithmic research and usage contexts: A perspective of contextualized evaluation for explainable AI," in *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, vol. 10, 2022, pp. 147–159.

[64] S. Jesus, C. Belém, V. Balayan, J. Bento, P. Saleiro, P. Bizarro, and J. Gama, "How can i choose an explainer? an application-grounded evaluation of post-hoc explanations," in *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, 2021, pp. 805–815.

[65] K. Morrison, M. Jain, J. Hammer, and A. Perer, "Eye into AI: Evaluating the interpretability of explainable AI techniques through a game with a purpose," *Proceedings of the ACM on Human-Computer Interaction*, vol. 7, pp. 1–22, 2023.

[66] U. Ehsan, P. Wintersberger, Q. V. Liao, E. A. Watkins, C. Manger, H. Daumé III, A. Riener, and M. O. Riedl, "Human-centered explainable ai (HCXAI): Beyond opening the black-box of ai," in *CHI conference on human factors in computing systems extended abstracts*, 2022, pp. 1–7.

[67] B. Mittelstadt, C. Russell, and S. Wachter, "Explaining explanations in AI," in *Proceedings of the conference on fairness, accountability, and transparency*, 2019, pp. 279–288.

[68] L. Weber, S. Lapuschkin, A. Binder, and W. Samek, "Beyond explaining: Opportunities and challenges of xai-based model improvement," *Information Fusion*, vol. 92, pp. 154–176, 2023.

[69] D. Kollias, A. Arsenos, and S. Kollias, "Domain adaptation explainability & fairness in ai for medical image analysis: Diagnosis of covid-19 based on 3-d chest ct-scans," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 4907–4914.

[70] C. Tang, N. Srishankar, S. Martin, and M. Tomizuka, "Grounded relational inference: Domain knowledge driven explainable autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, 2024.

[71] C.-H. Chang, J. Yoon, S. Ö. Arik, M. Udell, and T. Pfister, "Data-efficient and interpretable tabular anomaly detection," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023, pp. 190–201.

[72] C.-S. Lin and Y.-C. F. Wang, "Describe, spot and explain: Interpretable representation learning for discriminative visual reasoning," *IEEE Transactions on Image Processing*, vol. 32, pp. 2481–2492, 2023.

[73] Z. Zhang, L. Yilmaz, and B. Liu, "A critical review of inductive logic programming techniques for explainable ai," *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[74] W. Tang, J. Liu, Y. Zhou, and Z. Ding, "Causality-guided counterfactual debiasing for anomaly detection of cyber-physical systems," *IEEE Transactions on Industrial Informatics*, 2023.

[75] G. Cornacchia, V. W. Anelli, G. M. Biancofiore, F. Narducci, C. Pomo, A. Ragone, and E. Di Sciascio, "Auditing fairness under unawareness through counterfactual reasoning," *Information Processing & Management*, vol. 60, no. 2, p. 103 224, 2023.

[76] K. E. Mokhtari, B. P. Higdon, and A. Başar, "Interpreting financial time series with shap values," in *Proceedings of the annual international conference on computer science and software engineering*, 2019, pp. 166–172.

[77] J. Wexler, M. Pushkarna, T. Bolukbasi, M. Wattenberg, F. B. Viegas, and J. Wilson, "The What-If tool: Interactive probing of machine learning models," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 1, pp. 56–65, 2020. [Online]. Available: https://doi.org/10.1109/TVCG.2019.2934619.

[78] N. Kokhlikyan, V. Miglani, M. Martin, E. Wang, B. Alsallakh, J. Reynolds, A. Melnikov, N. Kliushkina, C. Araya, S. Yan, and O. Reblitz-Richardson, "Captum: A unified and generic model interpretability library for pytorch," 2020. arXiv: 2009.07896. [Online]. Available: https://arxiv.org/abs/2009.07896.

[79] B. Shi, Y. Chen, P. Zhang, C. D. Smith, J. Liu, A. D. N. Initiative, *et al.*, "Nonlinear feature transformation and deep fusion for Alzheimer's disease staging analysis," *Pattern Recognition*, vol. 63, pp. 487–498, 2017.

[80] Z. Yang, I. M. Nasrallah, H. Shou, J. Wen, J. Doshi, M. Habes, G. Erus, A. Abdulkadir, S. M. Resnick, M. S. Albert, *et al.*, "A deep learning framework identifies dimensional representations of Alzheimer's disease from brain structure," *Nature Communication*, vol. 12, no. 1, pp. 1–15, 2021.

[81] B. Yu, L. Zhou, L. Wang, W. Yang, M. Yang, P. Bourgeat, and J. Fripp, "Sa-lutnets: Learning sample-adaptive intensity lookup tables for brain tumor segmentation," *IEEE Transactions on Medical Imaging*, vol. 40, no. 5, pp. 1417–1427, 2021.

[82] S. Basaia, F. Agosta, L. Wagner, E. Canu, G. Magnani, R. Santangelo, M. Filippi, Alzheimer's Disease Neuroimaging Initiative, *et al.*, "Automated classification of Alzheimer's disease and mild cognitive impairment using a single MRI and deep neural networks," *NeuroImage: Clinical*, vol. 21, pp. 1–8, 2019.

[83] E. Lee, J. Choi, M. Kim, and H. Suk, "Toward an interpretable Alzheimer's disease diagnostic model with regional abnormality representation via deep learning," *NeuroImage*, vol. 202, pp. 1–15, 2019.

[84] D. Zhang, Y. Wang, L. Zhou, H. Yuan, and D. Shen, "Multimodal classification of Alzheimer's disease and mild cognitive impairment," *NeuroImage*, vol. 55, no. 3, pp. 856–867, 2011.

[85] F. Liu, C. Wee, H. Chen, and D. Shen, "Inter-modality relationship constrained multi-modality multi-task feature selection for Alzheimer's disease and mild cognitive impairment identification," *NeuroImage*, vol. 84, pp. 466–475, 2014.

[86] X. Zhu, H. Suk, and D. Shen, "Multi-modality canonical feature selection for Alzheimer's disease diagnosis," in *Medical Image Computing and Computer-Assisted Intervention*, P. Golland, N. Hata, C. Barillot, J. Hornegger, and R. D. Howe, Eds., vol. 8674, 2014, pp. 162–169.

[87] P. Cao, X. Shan, D. Zhao, M. Huang, and O. Zaiane, "Sparse shared structure based multi-task learning for MRI based cognitive performance prediction of Alzheimer's disease," *Pattern Recognition*, vol. 72, pp. 219–235, 2017.

[88] E. Gerardin, G. Chételat, M. Chupin, R. Cuingnet, B. Desgranges, H.-S. Kim, M. Niethammer, B. Dubois, S. Lehéricy, L. Garnero, *et al.*, "Multidimensional classification of hippocampal shape features discriminates Alzheimer's disease and mild cognitive impairment from normal aging," *NeuroImage*, vol. 47, no. 4, pp. 1476–1486, 2009.

[89] L. Sørensen, C. Igel, A. Pai, I. Balas, C. Anker, M. Lillholm, M. Nielsen, A. D. N. Initiative, *et al.*, "Differential diagnosis of mild cognitive impairment and Alzheimer's disease using structural MRI cortical thickness, hippocampal shape, hippocampal texture, and volumetry," *NeuroImage: Clinical*, vol. 13, pp. 470–482, 2017.

[90] Y. Chen and Y. Xia, "Iterative sparse and deep learning for accurate diagnosis of Alzheimer's disease," *Pattern Recognition*, vol. 116, pp. 1–10, 2021.

[91] J. Su, H. Shen, L. Peng, and D. Hu, "Few-shot domain-adaptive anomaly detection for cross-site brain images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–18, 2021.

[92] Y. Pan, M. Liu, Y. Xia, and D. Shen, "Disease-image-specific learning for diagnosis-oriented neuroimage synthesis with incomplete multi-modality data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–15, 2021.

[93] B. Lei, M. Yang, P. Yang, F. Zhou, W. Hou, W. Zou, X. Li, T. Wang, X. Xiao, and S. Wang, "Deep and joint learning of longitudinal data for Alzheimer's disease prediction," *Pattern Recognition*, vol. 102, pp. 1–13, 2020.

[94] C. Lian, M. Liu, J. Zhang, and D. Shen, "Hierarchical fully convolutional network for joint atrophy localization and Alzheimer's disease diagnosis using structural MRI," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 4, pp. 880–893, 2020.

[95] J. Kröll, S. B. Eickhoff, F. Hoffstaedter, and K. R. Patil, "Evolving complex yet interpretable representations: Application to Alzheimer's diagnosis and prognosis," in *Proceedings of the IEEE Congress on Evolutionary Computation*, 2020, pp. 1–8.

[96] K. Gopinath, C. Desrosiers, and H. Lombaert, "Learnable pooling in graph convolutional networks for brain surface analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 2, pp. 864–876, 2022.

[97] A. Y. Hannun, P. Rajpurkar, M. Haghpanahi, G. H. Tison, C. Bourn, M. P. Turakhia, and A. Y. Ng, "Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network," *Nature Medicine*, vol. 25, no. 1, pp. 65–69, 2019.

[98] P. Afshar, F. Naderkhani, A. Oikonomou, M. J. Rafiee, A. Mohammadi, and K. N. Plataniotis, "Mixcaps: A capsule network-based mixture of experts for lung nodule malignancy prediction," *Pattern Recognition*, vol. 116, pp. 1–29, 2021.

[99] Y. Xie, M. Chen, D. Kao, G. Gao, and X. Chen, "CheXplain: Enabling physicians to explore and understand data-driven, AI-enabled medical imaging analysis," in *Proceedings of the CHI Conference on Human Factors in Computing Systems*, 2020, pp. 1–13.

[100] D. R. Chittajallu, B. Dong, P. Tunison, R. Collins, K. Wells, J. Fleshman, G. Sankaranarayanan, S. Schwaitzberg, L. Cavuoto, and A. Enquobahrie, "Xai-cbir: Explainable AI system for content based retrieval of video frames from minimally invasive surgery videos," in *Proceedings of the IEEE International Symposium on Biomedical Imaging*, 2019, pp. 66–69.

[101] D. Jin, B. Zhou, Y. Han, J. Ren, T. Han, B. Liu, J. Lu, C. Song, P. Wang, D. Wang, *et al.*, "Generalizable, reproducible, and neuroscientifically interpretable imaging biomarkers for Alzheimer's disease," *Advanced Science*, vol. 7, no. 14, pp. 1–12, 2020.

[102] W. Hu, X. Meng, Y. Bai, A. Zhang, G. Qu, B. Cai, G. Zhang, T. W. Wilson, J. M. Stephen, V. D. Calhoun, *et al.*, "Interpretable multimodal fusion networks reveal mechanisms of brain cognition," *IEEE Transactions on Medical Imaging*, vol. 40, no. 5, pp. 1474–1483, 2021.

[103] C. R. Jack Jr, M. A. Bernstein, N. C. Fox, P. Thompson, G. Alexander, D. Harvey, B. Borowski, P. J. Britson, J. L. Whitwell, C. Ward, *et al.*, "The Alzheimer's disease neuroimaging initiative (ADNI): MRI methods," *Journal of Magnetic Resonance Imaging*, vol. 27, no. 4, pp. 685–691, 2008.

[104] S. G. Mueller, M. W. Weiner, L. J. Thal, R. C. Petersen, C. R. Jack, W. Jagust, J. Q. Trojanowski, A. W. Toga, and L. Beckett, "Ways toward an early diagnosis in Alzheimer's disease: The Alzheimer's disease neuroimaging initiative (ADNI)," *Alzheimer's & Dementia*, vol. 1, no. 1, pp. 55–66, 2005.

[105] E. Nigri, N. Ziviani, F. Cappabianco, A. Antunes, and A. Veloso, "Explainable deep cnns for MRI-based diagnosis of Alzheimer's disease," in *Proceedings of the IEEE International Joint Conference on Neural Networks*, 2020, pp. 1–8.

[106] C. Yang, A. Rangarajan, and S. Ranka, "Visual explanations from deep 3D convolutional neural networks for Alzheimer's disease classification," in *Proceedings of the AMIA Annual Symposium*, 2018, pp. 1571–1580.

[107] V. Petsiuk, A. Das, and K. Saenko, "Rise: Randomized input sampling for explanation of black-box models," in *Proceedings of the British Machine Vision Conference*, 2018, pp. 1–13.

[108] L. Xu, W. Ouyang, M. Bennamoun, F. Boussaïd, and D. Xu, "Multi-class token transformer for weakly supervised semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 1–19.

[109] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2020, pp. 1–21.

[110] L. Ru, Y. Zhan, B. Yu, and B. Du, "Learning affinity from attention: End-to-end weakly-supervised semantic segmentation with transformers," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 1–17.

[111] Q. Chen, L. Yang, J. Lai, and X. Xie, "Self-supervised image-specific prototype exploration for weakly supervised semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 1–11.

[112] H. Chefer, S. Gur, and L. Wolf, "Transformer interpretability beyond attention visualization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 782–791.

[113] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *Proceedings of the IEEE International Conference on Computer Vision*, 2021, pp. 9630–9640.

[114] R. Li, Z. Mai, Z. Zhang, J. Jang, and S. Sanner, "TransCAM: Transformer attention-based cam refinement for weakly supervised semantic segmentation," *Journal of Visual Communication and Image Representation*, vol. 92, pp. 1–8, 2023.

[115] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *International Conference on Machine Learning*, 2021, pp. 10 347–10 357.

[116] Z. Peng, W. Huang, S. Gu, L. Xie, Y. Wang, J. Jiao, and Q. Ye, "Conformer: Local features coupling global representations for visual recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, 2021, pp. 357–366.

[117] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in context," in *Proceedings of the European Conference on Computer Vision*, 2014, pp. 740–755.

[118] S. Abnar and W. H. Zuidema, "Quantifying attention flow in transformers," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 4190–4197.

[119] Z. Chen, T. Wang, X. Wu, X. Hua, H. Zhang, and Q. Sun, "Class re-activation maps for weakly-supervised semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 1–14.

[120] K. Yuan, G. Schaefer, Y.-K. Lai, Y. Wang, X. Liu, L. Guan, and H. Fang, "A multi-strategy contrastive learning framework for weakly supervised semantic segmentation," *Pattern Recognition*, pp. 1–12, 2023.

[121] S. Lee, M. Lee, J. Lee, and H. Shim, "Railroad is not a train: Saliency as pseudo-pixel supervision for weakly supervised semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 5495–5505.

[122] M. Everingham, S. M. A. Eslami, L. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *International Journal of Computer Vision*, vol. 111, no. 1, pp. 98–136, 2015.

[123] Z. Wu, C. Shen, and A. van den Hengel, "Wider or deeper: Revisiting the resnet model for visual recognition," *Pattern Recognition*, vol. 90, pp. 119–133, 2019.

[124] Y. Wang, J. Zhang, M. Kan, S. Shan, and X. Chen, "Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12 272–12 281.

[125] Y. Chang, Q. Wang, W. Hung, R. Piramuthu, Y. Tsai, and M. Yang, "Weakly-supervised semantic segmentation via sub-category exploration," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8988–8997.

[126] T. Wu, J. Huang, G. Gao, X. Wei, X. Wei, X. Luo, and C. H. Liu, "Embedded discriminative attention mechanism for weakly supervised semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 16 765–16 774.

[127] J. Lee, E. Kim, and S. Yoon, "Anti-adversarially manipulated attributions for weakly and semi-supervised semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4071–4080.

[128] K. Sun, H. Shi, Z. Zhang, and Y. Huang, "ECS-Net: Improving weakly supervised semantic segmentation by using connections between class activation maps," in *Proceedings of the IEEE International Conference on Computer Vision*, 2021, pp. 7263–7272.

[129] H. Kweon, S. Yoon, H. Kim, D. Park, and K. Yoon, "Unlocking the potential of ordinary classifier: Class-specific adversarial erasing framework for weakly supervised semantic segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2021, pp. 6974–6983.

[130] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2018.

[131] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "SegFormer: Simple and efficient design for semantic segmentation with transformers," in *Advances in Neural Information Processing Systems*, 2021, pp. 1–18.

[132] B. Zhang, J. Xiao, Y. Wei, M. Sun, and K. Huang, "Reliability does matter: An end-to-end weakly supervised semantic segmentation approach," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, pp. 12 765–12 772.

[133] D. Zhang, H. Zhang, J. Tang, X. Hua, and Q. Sun, "Causal intervention for weakly-supervised semantic segmentation," in *Advances in Neural Information Processing Systems*, 2020, pp. 1–12.

[134] L. Xu, W. Ouyang, M. Bennamoun, F. Boussaïd, F. Sohel, and D. Xu, "Leveraging auxiliary tasks with affinity learning for weakly supervised semantic segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2021, pp. 6964–6973.

[135] Y. Su, R. Sun, G. Lin, and Q. Wu, "Context decoupling augmentation for weakly supervised semantic segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2021, pp. 6984–6994.

[136] J. Lee, J. Choi, J. Mok, and S. Yoon, "Reducing information bottleneck for weakly supervised semantic segmentation," in *Advances in Neural Information Processing Systems*, 2021, pp. 27 408–27 421.

[137] B. Zhang, J. Xiao, J. Jiao, Y. Wei, and Y. Zhao, "Affinity attention graph neural network for weakly supervised semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, no. 1, pp. 1–15, 2021.

[138] Y. Liu, Y. Wu, P. Wen, Y. Shi, Y. Qiu, and M. Cheng, "Leveraging instance-, image- and dataset-level information for weakly supervised instance segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 3, pp. 1415–1428, 2022.

[139] S. Kho, P. Lee, W. Lee, M. Ki, and H. Byun, "Exploiting shape cues for weakly supervised semantic segmentation," *Pattern Recognition*, vol. 132, pp. 1–13, 2022.

[140] B. Zhang, J. Xiao, Y. Wei, K. Huang, S. Luo, and Y. Zhao, "End-to-end weakly supervised semantic segmentation with reliable region mining," *Pattern Recognition*, vol. 128, pp. 1–13, 2022.

[141] Z. Huang, X. Wang, J. Wang, W. Liu, and J. Wang, "Weakly-supervised semantic segmentation network with deep seeded region growing," in *Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7014–7023.

[142] E. Voita, D. Talbot, F. Moiseev, R. Sennrich, and I. Titov, "Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned," in *Proceedings of the Association for Computational Linguistics*, 2019, pp. 5797–5808.

[143] J. Ahn, S. Cho, and S. Kwak, "Weakly supervised learning of instance segmentation with inter-pixel relations," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2209–2218.

[144] A. Chavan, Z. Shen, Z. Liu, Z. Liu, K. Cheng, and E. P. Xing, "Vision transformer slimming: Multi-dimension searching in continuous optimization space," in *Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 1–11.

[145] Q. Zhang, S. Zuo, C. Liang, A. Bukharin, P. He, W. Chen, and T. Zhao, "PLATON: pruning large transformer models with upper confidence bound of weight importance," in *International Conference on Machine Learning*, 2022, pp. 26 809–26 823.

[146] F. Yu, K. Huang, M. Wang, Y. Cheng, W. Chu, and L. Cui, "Width & depth pruning for vision transformers," in *Proceeding of the AAAI Conference on Artificial Intelligence*, 2022, pp. 3143–3151.

[147] Y. Tang, Y. Wang, Y. Xu, D. Tao, C. Xu, C. Xu, and C. Xu, "SCOP: scientific control for reliable neural network pruning," in *Advances in Neural Information Processing Systems*, 2020, pp. 1–11.

[148] Z. Pan, B. Zhuang, J. Liu, H. He, and J. Cai, "Scalable vision transformers with hierarchical pooling," in *Proceedings of the IEEE International Conference on Computer Vision*, 2021, pp. 367–376.

[149] B. Pan, R. Panda, Y. Jiang, Z. Wang, R. Feris, and A. Oliva, "IA-RED^2: Interpretability-aware redundancy reduction for vision transformers," in *Advances in Neural Information Processing Systems*, 2021, pp. 24 898–24 911.

[150] T. Chen, Y. Cheng, Z. Gan, L. Yuan, L. Zhang, and Z. Wang, "Chasing sparsity in vision transformers: An end-to-end exploration," in *Advances in Neural Information Processing Systems*, 2021, pp. 19 974–19 988.

[151] M. Zhu, Y. Tang, and K. Han, "Vision transformer pruning," in *Proceedings of the Knowledge Discovery and Data Mining*, 2021, pp. 1–4.

[152] S. Yu, T. Chen, J. Shen, H. Yuan, J. Tan, S. Yang, J. Liu, and Z. Wang, "Unified visual transformer compression," in *International Conference on Learning Representations*, 2022, pp. 1–17.

[153] A. Fan, E. Grave, and A. Joulin, "Reducing transformer depth on demand with structured dropout," in *International Conference on Learning Representations*, 2020, pp. 1–15.

[154] T. Chen, J. Frankle, S. Chang, S. Liu, Y. Zhang, Z. Wang, and M. Carbin, "The lottery ticket hypothesis for pre-trained BERT networks," in *Advances in Neural Information Processing Systems*, 2020, pp. 1–13.

[155] E. Frantar and D. Alistarh, "SPDY: accurate pruning with speedup guarantees," in *International Conference on Machine Learning*, 2022, pp. 6726–6743.

[156] P. Michel, O. Levy, and G. Neubig, "Are sixteen heads really better than one?" In *Advances in Neural Information Processing Systems*, 2019, pp. 14 014–14 024.

[157] L. Huang, W. Wang, J. Chen, and X. Wei, "Attention on attention for image captioning," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 4633–4642.

[158] B. Na, J. Mok, H. Choe, and S. Yoon, "Accelerating neural architecture search via proxy data," in *International Joint Conference on Artificial Intelligence*, 2021, pp. 2848–2854.

[159] Z. Wang, J. Wohlwend, and T. Lei, "Structured pruning of large language models," in *Proceedings of the Empirical Methods in Natural Language Processing*, 2020, pp. 6151–6162.

[160] V. Sanh, T. Wolf, and A. M. Rush, "Movement pruning: Adaptive sparsity by fine-tuning," in *Advances in Neural Information Processing Systems*, 2020, pp. 1–12.

[161] Y. Liu, Z. Lin, and F. Yuan, "ROSITA: refined BERT compression with integrated techniques," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021, pp. 8715–8722.

[162] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," University of Toronto, Tech. Rep., 2009.

[163] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[164] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *International Conference on Machine Learning*, 2021, pp. 10 347–10 357.

[165] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE International Conference on Computer Vision*, 2021, pp. 10 012–10 022.

[166] J. Li, R. Cotterell, and M. Sachan, "Differentiable subset pruning of transformer heads," *Transactions of the Association for Computational Linguistics*, pp. 1442–1459, 2021.

[167] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," in *Proceedings of the European Conference on Computer Vision*, 2020, pp. 405–421.

[168] C. Wang, M. Chai, M. He, D. Chen, and J. Liao, "CLIP-NeRF: Text-and-image driven manipulation of neural radiance fields," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 3825–3834.

[169] Y. Yuan, Y. Sun, Y. Lai, Y. Ma, R. Jia, and L. Gao, "NeRF-editing: Geometry editing of neural radiance fields," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 332–18 343.

[170] S. Liu, X. Zhang, Z. Zhang, R. Zhang, J. Zhu, and B. Russell, "Editing conditional radiance fields," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021, pp. 5753–5763.

[171] B. Poole, A. Jain, J. T. Barron, and B. Mildenhall, "DreamFusion: Text-to-3D using 2D diffusion," in *International Conference on Learning Representations*, 2023.

[172] A. Haque, M. Tancik, A. A. Efros, A. Holynski, and A. Kanazawa, "Instruct-NeRF2NeRF: Editing 3D scenes with instructions," in *Proceedings of the IEEE International Conference on Computer Vision*, 2023.

[173] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.

[174] A. Karnewar, A. Vedaldi, D. Novotny, and N. J. Mitra, "Holodiffusion: Training a 3D diffusion model using 2D images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2023, pp. 18 423–18 433.

[175] T. Brooks, A. Holynski, and A. A. Efros, "Instructpix2pix: Learning to follow image editing instructions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2023, pp. 1–15.

[176] Z. He, W. Zuo, M. Kan, S. Shan, and X. Chen, "AttGAN: Facial attribute editing by only changing what you want," *IEEE Transactions on Image Processing*, vol. 28, no. 11, pp. 5464–5478, 2019.

[177] Y. Xu, Y. Yin, L. Jiang, Q. Wu, C. Zheng, C. C. Loy, B. Dai, and W. Wu, "TransEditor: Transformer-based dual-space GAN for highly controllable facial editing," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 7673–7682.

[178] H. Xu, G. Song, Z. Jiang, J. Zhang, Y. Shi, J. Liu, W. Ma, J. Feng, and L. Luo, "OmniAvatar: Geometry-guided controllable 3D head synthesis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12 814–12 824.

[179] K. Jiang, S. Chen, F. Liu, H. Fu, and L. Gao, "NeRFFaceEditing: Disentangled face editing in neural radiance fields," in *ACM SIGGRAPH*, 2022, pp. 1–9.

[180] H. Ling, K. Kreis, D. Li, S. W. Kim, A. Torralba, and S. Fidler, "EditGAN: High-precision semantic image editing," in *Advances in Neural Information Processing Systems*, 2021, pp. 16 331–16 345.

[181] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J. Zhu, and S. Ermon, "SDEdit: Guided image synthesis and editing with stochastic differential equations," in *International Conference on Learning Representations*, 2022.

[182] G. Parmar, K. K. Singh, R. Zhang, Y. Li, J. Lu, and J. Zhu, "Zero-shot image-to-image translation," in *ACM SIGGRAPH*, 2023, pp. 1–11.

[183] K. Kania, K. M. Yi, M. Kowalski, T. Trzcinski, and A. Tagliasacchi, "CoNeRF: Controllable neural radiance fields," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 602–18 611.

[184] B. Yang, C. Bao, J. Zeng, H. Bao, Y. Zhang, Z. Cui, and G. Zhang, "NeuMesh: Learning disentangled neural mesh-based implicit field for geometry and texture editing," in *Proceedings of the European Conference on Computer Vision*, 2022, pp. 597–614.

[185] A. Mirzaei, T. Aumentado-Armstrong, K. G. Derpanis, J. Kelly, M. A. Brubaker, I. Gilitschenski, and A. Levinshtein, "SPIn-NeRF: Multiview segmentation and perceptual inpainting with neural radiance fields," in *Proceedings of the IEEE International Conference on Computer Vision*, 2023, pp. 20 669–20 679.

[186] C. Zheng, W. Lin, and F. Xu, "EditableNeRF: Editing topologically varying neural radiance fields by key points," in *Proceedings of the IEEE International Conference on Computer Vision*, 2023, pp. 8317–8327.

[187] Y.-J. Yuan, Y.-T. Sun, Y.-K. Lai, Y. Ma, R. Jia, L. Kobbelt, and L. Gao, "Interactive NeRF geometry editing with shape priors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–17, 2023.

[188] D. Lahiri, N. Panse, and M. Kumar, "S2RF: semantically stylized radiance fields," in *Proceedings of the IEEE International Conference on Computer Vision*, 2023.

[189] E. R. Chan, C. Z. Lin, M. A. Chan, K. Nagano, B. Pan, S. D. Mello, O. Gallo, L. J. Guibas, J. Tremblay, S. Khamis, T. Karras, and G. Wetzstein, "Efficient geometry-aware 3D generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 102–16 112.

[190] J. Gu, L. Liu, P. Wang, and C. Theobalt, "StyleNeRF: A style-based 3D aware generator for high-resolution image synthesis," in *International Conference on Learning Representations*, 2022.

[191] H. Do, E. Yoo, T. Kim, C. Lee, and J. Y. Choi, "Quantitative manipulation of custom attributes on 3D-aware image synthesis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2023, pp. 8529–8538.

[192] C. Bao, Y. Zhang, B. Yang, T. Fan, Z. Yang, H. Bao, G. Zhang, and Z. Cui, "SINE: semantic-driven image-based NeRF editing with prior-guided editing field," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2023, pp. 20 919–20 929.

[193] J. Zhuang, C. Wang, L. Liu, L. Lin, and G. Li, "DreamEditor: Text-driven 3D scene editing with neural fields," in *ACM SIGGRAPH*, 2023.

[194] C. Wang, R. Jiang, M. Chai, M. He, D. Chen, and J. Liao, "NeRF-Art: Text-driven neural radiance fields stylization," *IEEE Transactions on Visualization and Computer Graphics*, 2023.

[195] O. Gordon, O. Avrahami, and D. Lischinski, "Blended-nerf: Zero-shot object generation and blending in existing neural radiance fields," in *Proceedings of the IEEE International Conference on Computer Vision*, 2023.

[196] N. M. Khalid, T. Xie, E. Belilovsky, and T. Popa, "CLIP-Mesh: Generating textured meshes from text using pretrained image-text models," in *ACM SIGGRAPH*, 2022, 25:1–25:8.

[197] A. Jain, B. Mildenhall, J. T. Barron, P. Abbeel, and B. Poole, "Zero-shot text-guided object generation with dream fields," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 857–866.

[198] A. Sanghi, H. Chu, J. G. Lambourne, Y. Wang, C. Cheng, M. Fumero, and K. R. Malekshan, "CLIP-Forge: Towards zero-shot text-to-shape generation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 582–18 592.

[199] H. Wang, X. Du, J. Li, R. A. Yeh, and G. Shakhnarovich, "Score jacobian chaining: Lifting pretrained 2D diffusion models for 3D generation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12 619–12 629.

[200] C. Lin, J. Gao, L. Tang, T. Takikawa, X. Zeng, X. Huang, K. Kreis, S. Fidler, M. Liu, and T. Lin, "Magic3D: High-resolution text-to-3D content creation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2023, pp. 300–309.

[201] G. Metzer, E. Richardson, O. Patashnik, R. Giryes, and D. Cohen-Or, "Latent-NeRF for shape-guided generation of 3D shapes and textures," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12 663–12 673.

[202] Y. Chen, C. Zhang, X. Yang, Z. Cai, G. Yu, L. Yang, and G. Lin, *IT3D: Improved text-to-3D generation with explicit view synthesis*, 2023. arXiv: 2308.11473.

[203] Z. Zhou and S. Tulsiani, *SparseFusion: Distilling view-conditioned diffusion for 3d reconstruction*, 2022. arXiv: 2212.00792.

[204] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in neural information processing systems*, vol. 33, 2020, pp. 6840–6851.

[205] C. Shi, H. Ni, K. Li, S. Han, M. Liang, and M. R. Min, "Exploring compositional visual generation with latent classifier guidance," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 853–862.

[206] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the International Conference on Machine Learning*, 2021, pp. 8748–8763.

[207] J. Seo, W. Jang, M.-S. Kwak, J. Ko, H. Kim, J. Kim, J.-H. Kim, J. Lee, and S. Kim, "Let 2D diffusion model know 3D-consistency for robust text-to-3D generation," *arXiv preprint arXiv:2303.07937*, 2023.

[208] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4104–4113.

[209] K. Park, K. Rematas, A. Farhadi, and S. M. Seitz, "PhotoShape: Photorealistic materials for large-scale shape collections," vol. 37, no. 6, p. 192, 2018.

[210] A. Dosovitskiy, G. Ros, F. Codevilla, A. M. López, and V. Koltun, "CARLA: an open urban driving simulator," in *CoRL*, 2017, pp. 1–16.

[211] K. Schwarz, Y. Liao, M. Niemeyer, and A. Geiger, "GRAF: Generative radiance fields for 3D-aware image synthesis," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 20 154–20 166.

[212] M. Tancik, E. Weber, E. Ng, R. Li, B. Yi, J. Kerr, T. Wang, A. Kristoffersen, J. Austin, K. Salahi, A. Ahuja, D. McAllister, and A. Kanazawa, "Nerfstudio: A modular framework for neural radiance field development," 2023. eprint: 2302.04264.

[213] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local nash equilibrium," vol. 30, 2017.