UNIVERSITY OF RIJEKA
FACULTY OF INFORMATICS AND DIGITAL TECHNOLOGIES

Matija Burić

APPLICATION OF COMPUTER VISION METHODS IN VETERINARY OPHTHALMOLOGY

DOCTORAL THESIS

Rijeka, 2024

SVEUČILIŠTE U RIJECI
FAKULTET INFORMATIKE I DIGITALNIH TEHNOLOGIJA

Matija Burić

PRIMJENA METODA RAČUNALNOG VIDA U VETERINARSKOJ OFTALMOLOGIJI

DOKTORSKI RAD

Mentor: prof. dr. sc. Marina Ivašić-Kos, Fakultet informatike i digitalnih tehnologija, Sveučilište u Rijeci

Rijeka, 2024.

UNIVERSITY OF RIJEKA
FACULTY OF INFORMATICS AND DIGITAL TECHNOLOGIES

Matija Burić

APPLICATION OF COMPUTER VISION METHODS IN VETERINARY OPHTHALMOLOGY

DOCTORAL THESIS

Supervisor: Professor Marina Ivašić-Kos, PhD, Faculty of Informatics and Digital Technologies, University of Rijeka

Rijeka, 2024

Mentor: prof. dr. sc. Marina Ivašić-Kos, Fakultet informatike i digitalnih tehnologija, Sveučilište u Rijeci

Doktorski rad obranjen je dana ______________________________ u/na

____,

pred povjerenstvom u sastavu:

1. ______________________________
2. ______________________________
3. ______________________________
4. ______________________________
5. ______________________________

Abstract

This dissertation researches computer vision techniques, utilizing deep learning models such as U-Net and GPT-4o, to improve the diagnosis of canine eye diseases in veterinary ophthalmology. One of the the contribution of this work is the development of the DogEyeSeg4 dataset, a collection of real-world clinical images specifically annotated for canine ophthalmology. Additionally, synthetic images based on DogEyeSeg4 dateset images were generated using Stable Diffusion with LoRA fine-tuning to enhance model robustness and generalization.

A modified U-Net model, U-Net+ResNet34(RSD), was trained on DogEyeSeg4 and synthetic images, for precise semantic segmentation of canine eye symptoms such as corneal cloudiness, scleral redness, excessive tearing, and colored mass protrusion in the eye corner. The study also trained individual binary segmentation models for each symptom, utilizing heatmaps from SSD eye detection to reduce false positives. Although these binary models improved symptom isolation, they faced challenges with overlapping conditions and increased complexity. Ultimately, the multiclass U-Net+ResNet34(RSD) model provided better overall performance and efficiency.

GPT-4o interpreted the segmented images, outperforming other Large Language Models (LLMs) in generating accurate diagnostic suggestions, particularly when using segmentation masks from the adjusted U-Net with a ResNet backbone alongside the original images.

Despite promising results, challenges remain in diagnosing complex or subtle conditions like corneal ulcers. Addressing these challenges should be considered when developing more accessible and reliable diagnostic tools. These findings underscore the potential for AI-driven tools to revolutionize veterinary ophthalmology, offering more accurate and efficient diagnostic processes that can ultimately improve animal care.

Keywords: canine ophthalmology, computer vision, U-Net, GPT-4o, image segmentation, deep learning, veterinary diagnostics, DogEyeSeg4, synthetic image augmentation

Prošireni sažetak

Disertacija istražuje primjenu tehnika računalnog vida za poboljšanje dijagnostike očnih bolesti kod pasa, s posebnim naglaskom na veterinarsku oftalmologiju. Cilj rada je unaprijediti proces dijagnostike korištenjem modela dubokog učenja, kao što su U-Net i GPT-4o, te razviti sveobuhvatan sustav koji kombinira segmentaciju slika i interpretaciju dijagnostičkih rezultata. Razvijeni sustav omogućava veterinarima točniju i bržu dijagnozu očnih bolesti, smanjujući potrebu za ručnom analizom i stručnim znanjem u svakodnevnoj kliničkoj praksi. Osnovni cilj istraživanja je poboljšanje postojećih modela računalnog vida za dijagnostiku kroz inovativnu primjenu stvarnih kliničkih slika, sintetičke augmentacije podataka i naprednih modela dubokog učenja.

Postupak istraživanja uključivao je nekoliko ključnih faza. U početku je stvoren i objavljen novi skup podataka pod nazivom DogEyeSeg4, koji sadrži slike snimljene u stvarnim uvjetima veterinarske klinike. Ovaj skup podataka uključuje različite pasmine, osvjetljenje i specifične kliničke uvjete, što ga čini pogodnim za treniranje modela koji se mogu generalizirati na širok raspon dijagnostičkih scenarija. Budući da su stvarni klinički podaci često ograničeni, korištena je sintetička augmentacija, pri čemu su sintetičke slike dodane kao sloj na stvarne slike kako bi se povećala varijabilnost i broj dostupnih podataka za treniranje. Time je osigurano da model može bolje prepoznati rijetke ili manje uobičajene simptome, čime se poboljšava njegova sposobnost generalizacije.

Unaprijeđen segmentacijski model U-Net+ResNet34(RSD) korišten je za automatsko prepoznavanje simptoma poput zamućenja rožnice, crvenila bjeloočnice, prekomjernog suzenja i protruzija obojene mase u kutu oka. Osim višeklasnog modela, trenirani su i individualni binarni modeli za svaki simptom, koristeći toplinske karte iz SSD detekcije oka kako bi se smanjili lažno pozitivni rezultati. Iako su binarni modeli poboljšali izolaciju pojedinih simptoma, suočili su se s izazovima preklapajućih stanja i većom složenošću, dok je višeklasni U-Net+ResNet34(RSD) model pružio bolju ukupnu učinkovitost.

Nakon segmentacije, slike su proslijeđene modelima GPT-4 i GPT-4o koji su zaduženi za interpretaciju simptoma i generiranje tekstualnih dijagnoza. Usporedba rezultata između GPT-4 i njegove napredne verzije GPT-4o pokazala je da GPT-4o nudi značajno bolje dijagnostičke prijedloge, osobito kada se koriste maske segmentiranih slika zajedno s originalnim slikama. Evaluacija uspješnosti obavljena je korištenjem raznih metrika, uključujući BERTScore, CLIPScore i druge koje se koriste za evaluaciju teksta, pri čemu su rezultati pokazali superiornost kombiniranih pristupa u odnosu na modele koji se oslanjaju isključivo na neobrađene slike.

Rezultati istraživanja ukazuju na visoku učinkovitost razvijenog sustava u prepoznavanju i dijagnosticiranju bolesti oka kod pasa. Modeli su se pokazali posebno uspješnima u dijagnosticiranju bolesti poput uveitisa i prolapsa žlijezde trećeg očnog kapka, gdje je GPT-4o postigao najviše rezultate. Također je uočen problem kod dijagnosticiranja kornealnog ulkusa, gdje GPT-4o nije uspio prepoznati bolest, za razliku od GPT-4.

Ovi rezultati ukazuju na potrebu za daljnjim poboljšanjima u modelima, osobito u slučajevima kada su simptomi suptilni ili atipični. Nadalje, segmentacijski model U-Net+ResNet34(RSD) pokazao se robusnim u širokom spektru slučajeva, no performanse su varirale ovisno o kvaliteti ulaznih slika, što ukazuje na potrebu za dodatnim usavršavanjem kako bi se smanjila ovisnost o vanjskim faktorima poput osvjetljenja i fokusa slike. Zaključci ovog istraživanja ukazuju na veliki potencijal primjene računalnog vida i modela dubokog učenja u svakodnevnoj veterinarskoj praksi. Integracija modela U-Net+ResNet34(RSD) za segmentaciju i GPT-4o za dijagnostičku interpretaciju omogućava brže i preciznije prepoznavanje bolesti oka kod pasa, smanjujući potrebu za ručnom analizom od strane eksperta te omogućujući veterinarima brže donošenje odluka. Daljnji razvoj ovog sustava uključuje proširenje skupa podataka DogEyeSeg4 kako bi se uključile slike iz više izvora i raznih pasmina, širi spektar simptoma, te istraži nove arhitekture modela, poput samokonfigurirajućih modela i transformera, koji bi mogli dodatno unaprijediti preciznost segmentacije i dijagnoze. Također, realna primjena ovog sustava u realnim uvjetima u veterinarskim klinikama predstavlja značajanu pomoć veterinarima ubrzavajući i olakšavajući donošenje preliminarne dijagnoze na temelju automatski segmentiranih slika. Očekuje se da će daljnje usavršavanje modela i prikupljanje povratnih informacija iz stvarnih kliničkih okruženja poboljšati uspješnost sustava i osigurati njegovu širu primjenu.

Konačno, sustav nudi mogućnosti za daljnja istraživanja u području veterinarske medicine, ne samo za oftalmološke bolesti već i za druge dijagnostičke zadatke, što može dovesti do razvoja novih alata koji će unaprijediti kvalitetu skrbi za životinje.

Ključne riječi: veterinarska oftalmologija, računalni vid, U-Net, GPT-4o, segmentacija slike, duboko učenje, veterinarska dijagnostika, DogEyeSeg4, sintetička augmentacija slika

Contents

1. Introduction 1
  1.1. Problem and Research Subject with Hypothesis 1
  1.2. Purpose and Objectives 2
  1.3. Brief Review of Previous Research 3
  1.4. Scientific Methods 3
  1.5. Structure of the Dissertation 4

2. Literature Review 5

3. Methodology 8
  3.1. Dataset 8
    3.1.1. DogEyeSeg4 Custom Dataset 8
    3.1.2. Synthetic Datasets 12
  3.2. Model Architecture and Training 18
    3.2.1. U-Net 18
    3.2.2. GPT-4 and GPT-4o 26
    3.2.3. Grounding SAM 29
    3.2.4. Single Shot Multibox Detector (SSD) for Eye Detection 31
  3.3. Web Application Development 34

4. Experiments and Results 37
  4.1. Test Dataset 37
  4.2. Metrics 40
    4.2.1. Segmentation Metrics 40
    4.2.2. Text Similarity Metrics 42
  4.3. Performance Evaluation 45
    4.3.1. Experiment 1: Ocular symptom segmentation with U-Net 45
    4.3.2. Experiment 2: Segmentation Comparative Study 57
    4.3.3. Experiment 3: Binary vs. Multiclass Segmentation 60
    4.3.4. Experiment 4: Diagnostic Interpretation 64
    4.3.5. Experiment 5: Combined Approach for Complete Diagnosis 71

5. Discussion 74

6. Conclusion 78

Bibliography 80

List of Published Research Papers in Computer Vision 97

Chapter 1

Introduction

1.1 Problem and Research Subject with Hypothesis

The field of veterinary ophthalmology faces significant challenges in diagnosing and managing ocular conditions in canines, primarily due to the limited availability of advanced diagnostic tools. While human medicine has seen remarkable progress with the integration of computer vision techniques, especially in the domain of ophthalmology[1], these technological advancements have not been fully leveraged in veterinary applications. The disparity between the diagnostic capabilities available in human and veterinary medicine highlights a critical gap that this research aims to address. The core problem of this dissertation is to bridge this gap by developing and applying computer vision methods specifically tailored for canine eye disease detection and diagnosis.

The subject of this research centers on the innovative application of deep learning models, particularly those designed for image segmentation, to help diagnosing ocular conditions in dogs. The research explores the potential of adapting and optimizing image segmentation models, which have proven effective in human medical imaging, to meet the challenges posed by veterinary ophthalmology. This includes dealing with diverse imaging conditions, varying image quality, and the need for precise segmentation of specific ocular features.

To guide this research, the following hypotheses have been formulated:

- **A Computer vision model can recognize certain canine ocular conditions in still images taken in an unconstrained environment.**
- **Modification of the input and architecture of the U-Net network contributes to a better segmentation of canine eye conditions.**

The first hypothesis addresses the core objective of demonstrating that a computer vision model, when applied to images captured in non-ideal, real-world settings, can accurately identify specific ocular conditions in canines. This aspect of the research is crucial because, unlike controlled laboratory environments, real-world veterinary practices often involve images that vary significantly in quality due to factors such as lighting, angle, and the behavior of the animals during imaging.

The second hypothesis related to the technical enhancement of the model itself, positing that by modifying both the input features and the architectural elements of the U-Net network, the model's performance in segmenting canine eye conditions can be improved. This hypothesis suggests that through careful adjustments and refinements, the model can be made more robust and capable of handling the complexities associated with veterinary ophthalmic images, which often include a wide range of conditions and variations.

Together, these hypotheses set the foundation for a comprehensive investigation into the applicability of advanced computer vision techniques in veterinary medicine, aiming to develop tools that could revolutionize the way canine eye diseases are diagnosed and managed in clinical settings.

1.2 Purpose and Objectives

The purpose of this dissertation is to apply advanced computer vision techniques to assist medical staff with the diagnosis of canine oculus conditions. Specifically, the research is directed towards developing a dataset and implementing deep learning models aimed at achieving two primary tasks: **symptom segmentation** and **disease recognition** in real-world veterinary settings. Symptom segmentation here refers to semantic segmentation, where the model distinguishes between specific regions within the eye, such as the cornea, sclera, and conjunctiva, to identify areas with ocular symptoms like corneal cloudiness or scleral redness. This segmentation provides a detailed pixel-level analysis, supporting a more accurate assessment of observable abnormalities in the eye.

Following semantic segmentation, the segmented images and identified symptoms are processed by a large language model (LLM) to generate a diagnostic interpretation. The objective for the LLM is to produce an accurate diagnosis that aligns with reference diagnoses, aiding in identifying the disease (e.g., "cherry eye") based on the segmented symptoms. Additionally, there is potential for the LLM to suggest treatment guidance for the identified ocular condition, though this application has not been tested in this study.

The specific objectives, aligned with the expected scientific contributions, are:

1. **An image dataset for machine learning of canine eye diseases**: Creation of a publicly available, annotated dataset for training machine learning models on canine eye diseases.
2. **Deep convolutional neural network model for recognition of canine eye clinical symptoms and diseases from still images in unconstrained environments**: Development of a CNN-based model for identifying eye symptoms and diseases in images captured under varied conditions.
3. **Deep neural network based on U-Net for segmentation of canine eye clinical symp-**

toms from still images in unconstrained environments: Implementation of a U-Net-based model for segmenting clinical symptoms in canine eye images.

4. **An improved method for segmentation of canine eye conditions based on U-Net**: Refinement of the U-Net model to improve segmentation accuracy for diagnosing canine eye conditions.

1.3 Brief Review of Previous Research

Computer vision has made significant strides in human medical diagnostics, with successful applications in ophthalmology, such as the detection of glaucoma and retinal diseases using Convolutional Neural Networks (CNNs)[2, 3]. However, the application of similar technologies in veterinary medicine, particularly for diagnosing canine eye diseases, is limited. Previous studies have demonstrated the potential of CNNs for disease recognition in controlled environments[4], but there remains a gap in their application under real-world, unconstrained conditions. This research aims to bridge that gap by developing a tailored dataset and a specialized U-Net model for veterinary use. In recent years, models like Grounding SAM[5] have introduced flexible, prompt-driven approaches to object detection and segmentation, offering utility in fields with limited labeled data. Grounding SAM combines Grounding DINO[6] for object detection with SAM (Segment Anything Model)[7] for segmentation, relying on text prompts instead of extensive retraining. This adaptability, beneficial for domains like veterinary ophthalmology, provides solution when conventional labeled datasets are unavailable.

1.4 Scientific Methods

The research employs several scientific methods:

1. **Data Collection and Annotation**: A custom dataset was created using images collected from veterinary clinics and annotated by experts.
2. **Deep Learning Model Development**: The U-Net architecture was modified and trained using transfer learning and data augmentation techniques to improve segmentation accuracy.
3. **Model Evaluation**: The performance of the model was assessed using metrics such as the Jaccard Index and Dice Similarity Coefficient, followed by statistical analysis using ANOVA and Tukey HSD tests.
4. **Application Development**: A web-based tool was developed and deployed using Docker to ensure portability and scalability.

1.5 Structure of the Dissertation

The dissertation is structured as follows:

- **Chapter 2**: Provides a comprehensive literature review, covering the use of computer vision in human and veterinary medicine and an overview of deep learning techniques relevant to this research.
- **Chapter 3**: Details the methodology, including dataset creation, model architecture, training processes, and the development of the web application.
- **Chapter 4**: Presents the experimental results, including model performance metrics, statistical analyses, and a discussion of the prototype application.
- **Chapter 5**: Discusses the implications of the findings, the contribution of the novel dataset, model effectiveness, and limitations, concluding with suggestions for future research.
- **Chapter 6**: Summarizes the dissertation’s contributions, the significance of the research for veterinary practice, and final remarks on the study’s impact.

Chapter 2

Literature Review

The application of computer vision techniques has significantly advanced the field of medicine, particularly in the diagnosis and analysis of various medical conditions. Deep Convolutional Neural Networks (CNNs), a class of deep learning models, have been widely used in ophthalmology for tasks such as detecting glaucoma, diabetic retinopathy, and other retinal disease recognition tasks, providing high accuracy in ocular image analysis and disease classification [8, 9]. These advancements are attributed to CNNs' ability to automatically extract and learn relevant features from ocular images, improving diagnostic accuracy compared to traditional manual methods [10, 11, 12].

The development of these CNN-based models has been supported by a variety of publicly available datasets, which have enabled the training and validation of models in human ophthalmology. Notable datasets include: ORIGA-light dataset focusing on optic nerve head segmentation and glaucoma assessment [13], Drishti-GS designed for glaucoma detection, containing annotated fundus images for training CNNs [14], Retinal Fundus Image for Glaucoma Detection with a focus on glaucoma diagnosis [15], RIM-ONE, a large dataset for retinal image analysis, primarily used for glaucoma detection [16], iBUG focuses on facial landmark detection but has also been used for eye-tracking and ocular disease analysis [17], OpenEDS dataset that includes annotated eye images for eye-tracking and ocular disease applications [18], UBIRIS primarily for biometric purposes, but also applied in ocular disease diagnosis [19], and TEyeD dataset containing eye-tracking data, useful for studying eye movements and diagnosing conditions such as glaucoma [20]. These datasets have significantly contributed to the development of effective models for human ophthalmology.

In contrast, the application of computer vision techniques in veterinary ophthalmology, particularly for canine eye diseases, is much less developed. Datasets and research in this field are scarce. Studies focusing on conditions such as canine glaucoma are limited [21, 22, 23], and the few available datasets are typically small and lack the diversity seen in human datasets. A notable study used CNNs to diagnose ulcerative keratitis in dogs, but it was constrained by a limited dataset and acquired under controlled conditions [24]. Additionally, A recent study further explored CNNs in veterinary ophthalmology, applying transfer learning and data augmentation to enhance diagnostic performance across multiple ocular conditions. However, this work was limited by a non-public dataset[25]. This highlights the need for more extensive datasets and research into canine ophthalmology.

The U-Net architecture, introduced for biomedical image segmentation, has become a popular choice for various medical imaging tasks due to its encoder-decoder structure, which enables precise feature extraction and localization [26]. U-Net's architecture is particularly effective in segmenting medical images where pixel-level accuracy is critical. It has demonstrated robust performance across multiple domains, including ophthalmology, where it has been applied to tasks such as optic disc segmentation, retinal layer segmentation, and detecting diseases like glaucoma and cataracts [27, 28, 29, 30]. Studies have consistently shown that U-Net performs well in medical image analysis, confirming its relevance in both human and veterinary medical contexts [26, 27, 28, 31, 32, 33].

Transformer-based models, such as Swin Transformer [34] and SegFormer [35], have recently gained attention for their superior ability to capture global context, making them particularly effective in fine-grained segmentation tasks across human and veterinary fields. However, U-Net continues to be a formidable option, especially in situations with limited data availability, where it still performs robustly [36]. In fact, a comparison between U-Net and transformer-based architectures for medical image registration demonstrated that U-Net, with minimal adjustments, can surpass the performance of these newer transformer models [36].

One of the key strengths of U-Net is its adaptability, particularly when combined with transfer learning. Transfer learning allows models to leverage pre-trained weights from large datasets, enabling them to perform well even on smaller datasets typical in veterinary applications. U-Net has been successfully combined with advanced CNN backbones like ResNet [37], EfficientNet [38], VGG [39], and Inception [40], significantly improving its feature extraction capabilities [11, 12]. These backbones help enhance the model's ability to learn from limited data, making it well-suited for applications where obtaining large datasets is impractical. Furthermore, studies have shown that U-Net does not require an extensive dataset to achieve good results, especially when augmented with transfer learning techniques [41].

While U-Net can perform well with small datasets, the creation of synthetic datasets can further enhance model performance by augmenting the available training data. Synthetic data generation techniques, such as those using Generative Adversarial Networks (GANs) and diffusion models, have become valuable tools for augmenting datasets in various domains [42, 43]. Diffusion models, particularly Stable Diffusion, have shown promise in generating high-quality synthetic images by iteratively refining noisy images [44]. Stable Diffusion, which employs a U-Net-like architecture for image synthesis, can be used to create realistic synthetic images [45, 46, 47] of diseased eyes, which can significantly improve the training of CNNs for canine ophthalmology. These synthetic datasets could address the shortage of annotated data in veterinary ophthalmology and enhance model robustness [48].

A web-based application utilizing a U-Net model trained on real and synthetic datasets could provide a valuable diagnostic tool for veterinarians. This application could assist in the early detection and treatment of canine eye diseases. In combination with image segmentation, the integration of Large Language Models (LLMs) could further enhance the diagnostic process. LLMs such as ChatGPT [49], Mistral [50], Gemini [51], Llama [52], and Claude [53] have demonstrated significant potential in medical data interpretation. These models, when integrated with image analysis tools, could help in interpreting symptoms, guiding veterinarians through diagnostic workflows, and improving decision-making. Evaluation metrics like BERTScore [54], CLIPScore [55], BLEU [56], METEOR [57], ROUGE [58], and SPICE [59] have shown that LLMs can effectively process complex medical information, making them suitable for integration into diagnostic applications.

A mobile web application will be developed as part of this research, utilizing advanced AI methods to support canine eye disease diagnosis. The application combines U-Net-based image segmentation, trained on a custom dataset *DogEyeSeg4* [60] along with synthetic data, to identify and analyze ocular symptoms.

Chapter 3

Methodology

3.1 Dataset

In veterinary ophthalmology, especially in the niche of canine eye diseases, obtaining suitable datasets poses significant challenges. Unlike human medical datasets, which benefit from more standardized and widely available sources, veterinary datasets are relatively scarce. The images needed for training models in this domain must capture a variety of conditions across multiple breeds, often under non-ideal circumstances. As a result, a custom dataset was developed to support this research, addressing the scarcity of available data. The following sections describe the *DogEyeSeg4* dataset, constructed to overcome these challenges, as well as a synthetic dataset generated to augment the available real-world data.

3.1.1 DogEyeSeg4 Custom Dataset

Given the limited availability of publicly accessible canine ophthalmic datasets, the development of the *DogEyeSeg4* dataset became essential. The process of gathering suitable data from real-world clinical environments posed several difficulties. First, patient compliance during eye examinations often impacted image quality. Dogs, being non-cooperative subjects, frequently moved during assessments, resulting in blurry images or requiring multiple attempts to capture usable data. Second, the diversity of breeds presented additional variability in eye structure, fur coloration, and size, all of which affected the clarity and focus of the images. These factors, while reflective of real-world conditions, made it difficult to obtain consistent, high-quality images that were necessary for robust model training.

The images in the *DogEyeSeg4* dataset were collected from two specialized veterinary ophthalmology clinics and a veterinary eye disease atlas [61]. Clinical environments, unlike controlled laboratory settings, introduce uncontrollable variables such as inconsistent lighting, non-standardized camera equipment, and varying angles of capture. These factors result in natural but challenging conditions for machine learning models. The lighting in veterinary clinics often varies depending on the location of the examination, leading to different levels of contrast and exposure in the images. Additionally, images were captured without staged conditions, meaning the dataset includes a range of natural scenarios rather than perfectly lit or artificially enhanced images. This variety adds to the complexity but also enhances the dataset's applicability to real-world diagnostic conditions.

The *DogEyeSeg4* dataset consists of 145 images, which include both close-up images of the canine eye and full headshots. Example images with corresponding masks are shown in Figure 3.1. This diversity is crucial because veterinary practitioners often capture images that show either the entire head of the animal or focus specifically on the affected eye, depending on the diagnostic requirement. Close-up images provide detailed views of specific conditions such as corneal cloudiness or excessive tearing, while whole-head images are more common in clinical settings and may capture multiple symptoms, including redness of the sclera or colored masses in the corner of the eye, from a broader perspective. Each image was resized to 320×320 pixels for consistency during model training, and the dataset is annotated with one-channel masks in PNG format.

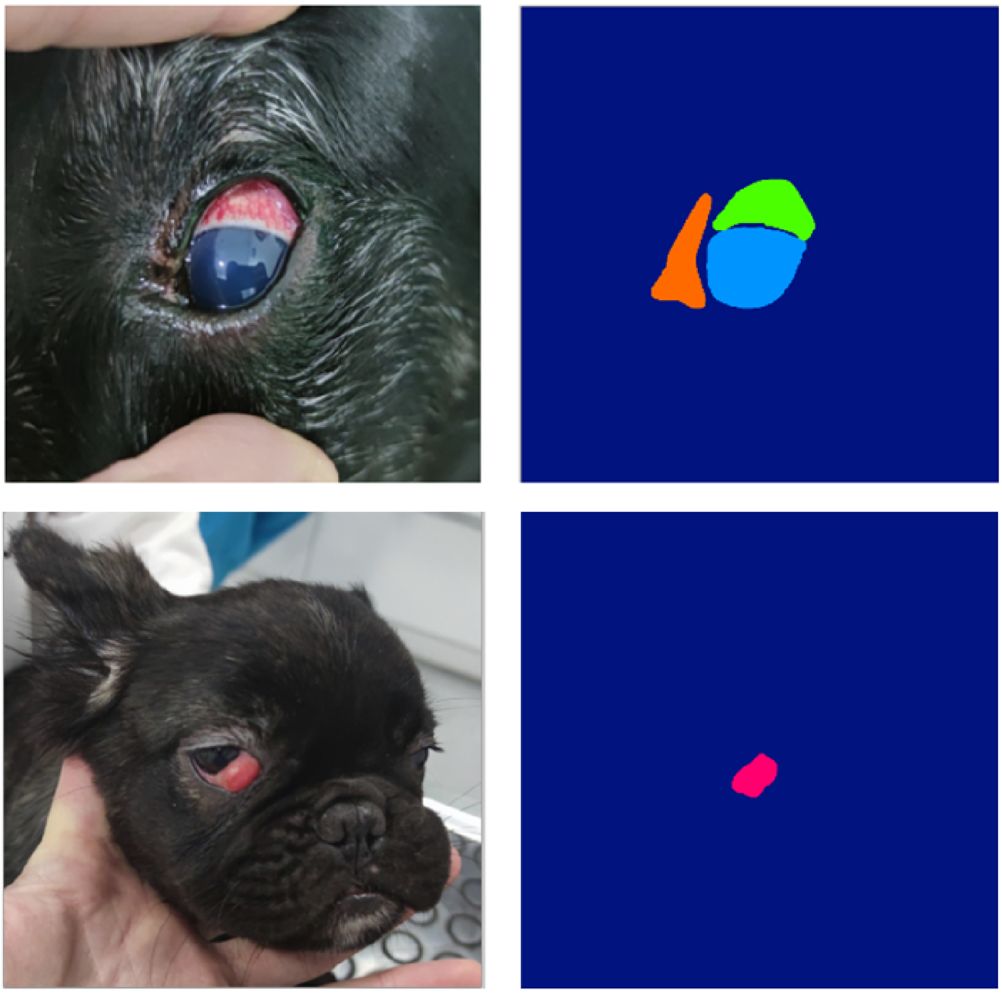


Figure 3.1: Example of images in DogEyeSeg4 dataset with corresponding masks showing closeup of an eye (upper row) and whole head (lower row) [62].

The four annotated classes, described in Figure 3.2 [62], correspond to the following symptoms:

- **S1** Cloudiness or haziness of the cornea,
- **S2** Redness of the sclera,
- **S3** Excessive tearing,
- **S4** A colored mass in the corner of the eye.

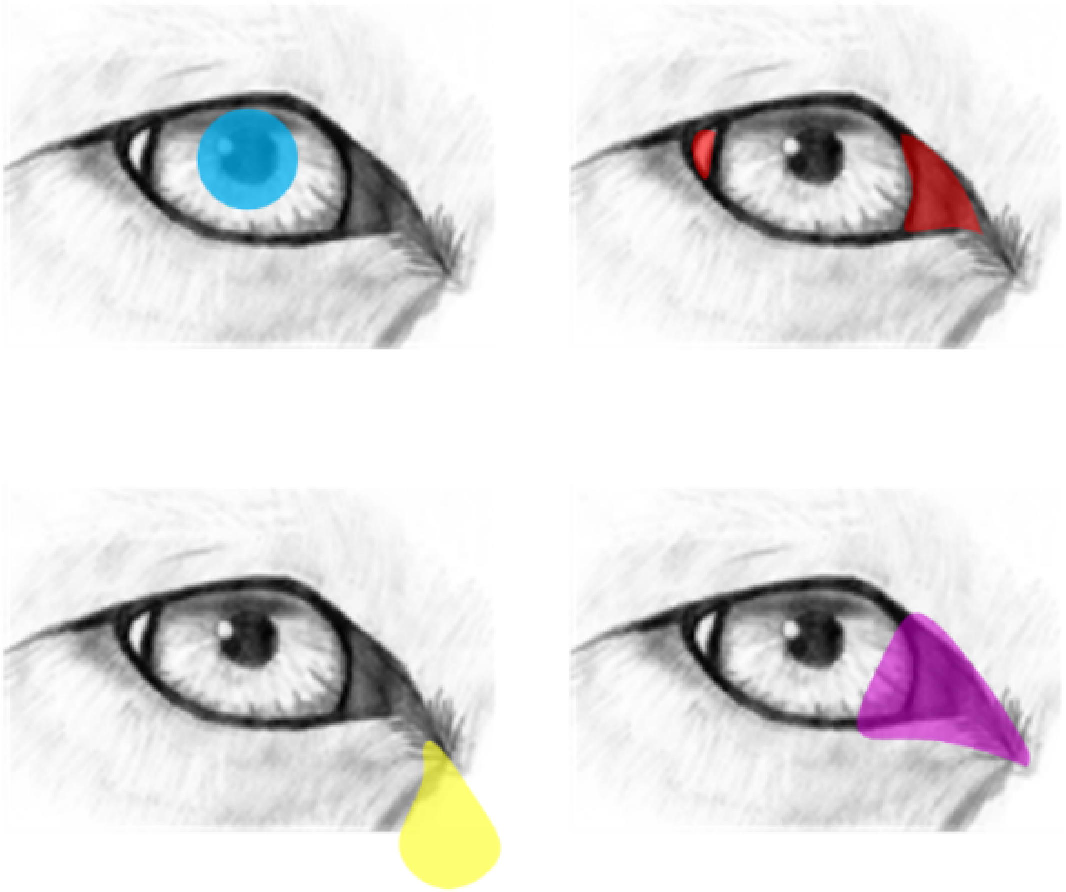


Figure 3.2: visual representation of medical symptoms in the top-left to bottom-right order: S1 - cloudiness or haziness of the cornea; S2 - sclera redness; S3 - excessive tears; and a S4 - colored mass protrusion in the corner of the eye[62].

These symptoms correspond to several common diseases, such as Cherry Eye, Glaucoma, Uveitis, Corneal Ulceration, and Bacterial Keratitis. To ensure the clinical relevance and accuracy of the dataset, each image and its annotation were reviewed by a veterinary specialist. This review process was critical in maintaining the diagnostic precision of the dataset, especially given the variability in patient behavior and the non-standardized clinical conditions under which the images were captured.

Ensuring the dataset complied with data protection regulations, particularly the General Data Protection Regulation (GDPR) [63], was a critical part of the dataset creation process. All images included in the dataset were anonymized to safeguard the privacy of clients and their animals. Anonymization involved the removal of any identifiable information, such as examination dates, client names, and animal identifiers. By adhering to GDPR guidelines, the dataset was rendered compliant with strict privacy regulations. Additionally, since the images were gathered during standard veterinary care and no animals were harmed for the purpose of image acquisition, ethical approval was not required for this study. This was made explicit in the ethics and consent statement accompanying the dataset: *"Images were collected as part of routine clinical evaluations, and no ethical approval was necessary as no harm was caused to the animals."*

One of the key challenges in assembling this dataset was balancing the representation of each class. For example, some symptoms, such as excessive tearing (S3), appeared more frequently than others, such as the colored mass (S4). To avoid bias in model training due to class imbalance, similar images were manually removed, resulting in an even distribution of ocular conditions across the dataset (see Figure 3.3). This process not only balanced class representation but also increased the dataset's diversity, supporting more robust model training across all four classes.

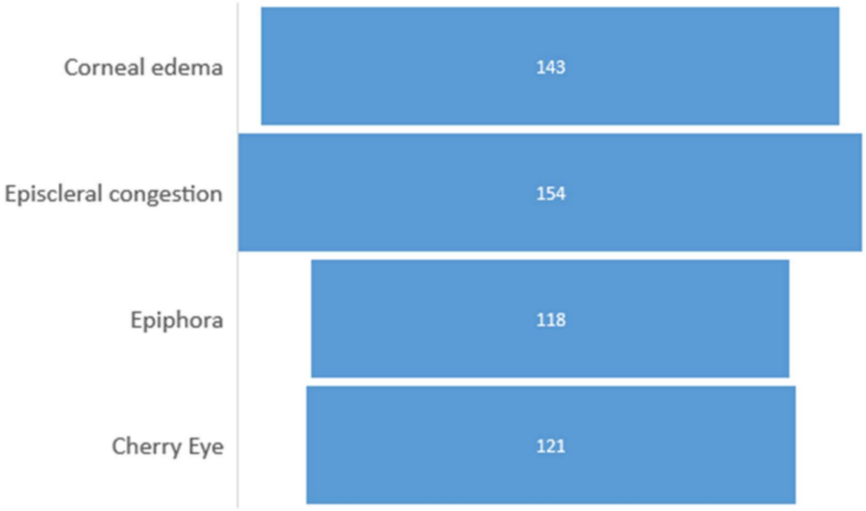


Figure 3.3: Distribution of classes in the DogEyeSeg4 Dataset. The number represents the recurrence of certain ocular diseases in images. Ocular symptoms which accompany Corneal edema, Episcleral congestion, Epiphora and Cherry Eye are Cloudiness or haziness of the cornea (S1), Redness of the sclera (S2), Excessive tearing (S3) and A colored mass in the corner of the eye (S4) respectively [62].

To further enhance the dataset's utility, data augmentation techniques were applied. Building on methodologies from medical image segmentation, particularly with respect to U-Net architecture, augmentation methods such as horizontal flipping, rotation by up to 15 degrees, and translation by 50 pixels were used, as seen in Figure 3.4 [62]. These augmentations introduce variation in camera angles and positions, simulating different real-world scenarios where veterinary professionals might capture images from slightly different perspectives. Importantly, zoom augmentation was avoided to prevent interpolation, which could introduce noise into the masks and degrade the accuracy of the annotations. The augmentation process expanded the dataset to 200 images, increasing its robustness and making it more suitable for training machine learning models that need to generalize across diverse imaging conditions.

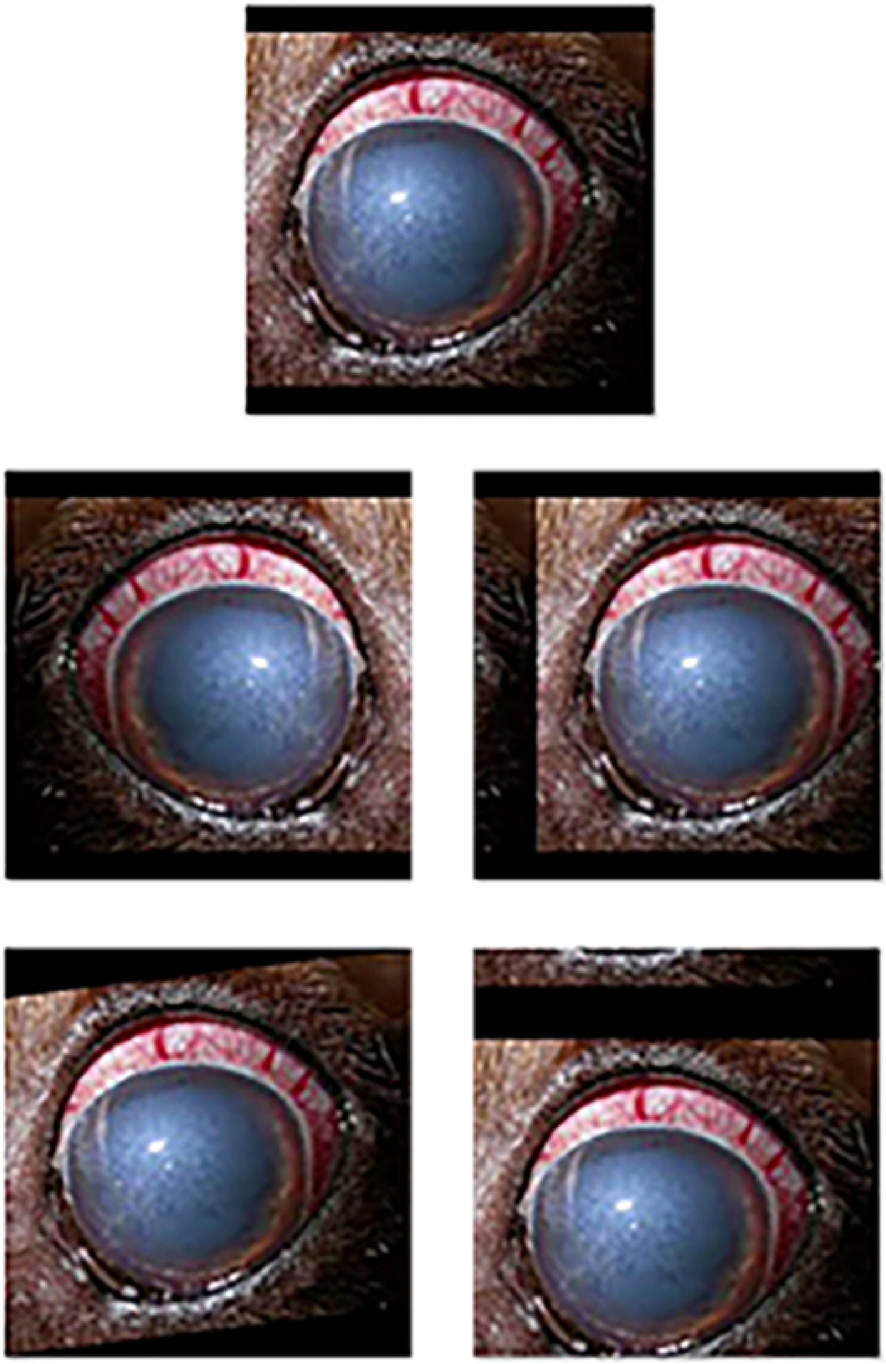


Figure 3.4: Example of applied augmentations on original image in top row: horizontal flip (middle row left), horizontal shift (middle row right), rotation (bottom row left), and vertical shift (bottom row right) [62].

3.1.2 Synthetic Datasets

Due to the limited availability of real-world data and the need for large datasets to train deep learning models, synthetic image generation has become a valuable resource in medical imaging, including veterinary ophthalmology. Several techniques exist for generating synthetic images, each offering strengths and weaknesses in tackling this issue. These methods include Generative Adversarial Networks (GANs) [64], Variational Autoencoders (VAEs) [65], and Diffusion Models [66]. Additionally, procedural methods like rule-based systems and 3D rendering techniques are also applied for synthetic image generation, offering high control over image features, albeit with increased manual intervention and resource demand [67].

GANs utilize a two-part system, where the generator creates images, and the discriminator attempts to distinguish them from real images. While GANs are capable of producing high-resolution, realistic images, they are computationally expensive and often unstable during training, with issues like "mode collapse," where limited variations of images are generated. VAEs, by contrast, are probabilistic models that encode input data into a latent space and then decode it back, generating new data by sampling from this latent space. VAEs are more stable to train than GANs but tend to produce lower-resolution images that may lack the detailed features required in tasks such as ophthalmic disease diagnosis.

Diffusion models, particularly the Stable Diffusion variant used in this study, offer a more balanced approach, combining computational efficiency with high-quality image generation. A significant feature of diffusion models is their reliance on the U-Net architecture. In diffusion models, U-Net serves as the backbone for the denoising process, transforming random noise into coherent images. This structure makes diffusion models particularly well-suited for generating structured medical images [68].

Beyond learning-based techniques, rule-based image generation and 3D rendering techniques are also used in certain fields for procedural image creation [67]. Rule-based systems apply predefined algorithms to generate images, offering high customization but requiring extensive manual intervention. Rendering techniques simulate 3D environments and lighting to create highly detailed images, but these methods demand considerable setup and computational resources [69].

For this study, Stable Diffusion[70] was selected due to its balance between image quality and computational efficiency. The models WildlifeX Animals[71] and Realistic Vision V6.0 B1 HyperVAE[72] were used, available on Civitai Generative AI[73]. A large number of synthetic images of various dog breeds were generated, and through manual selection, only images that closely matched the characteristics of images in the DogEyeSeg4 dataset and maintained realistic quality were retained. Images with unrealistic characteristics were excluded.

Example images generated using Stable Diffusion can be observed in Figure 3.5[74]. Example prompt for the bottom center image in Figure 3.5 is:

- **Positive prompt**: "Dalmatian dog's head, portrait of a dog in a park, realistic image, realistic fur, award-winning wildlife (medium long shot:1.4), 35mm film movie still."
- **Negative prompt**: "abnormal, ugly, abnormal skin"
- **Other parameters**: Steps: 20, Sampler: DPM++ 2M, Schedule type: Karras, CFG scale:

7, Size: 512x512, Model: wildlifexAnimals.

Diffusion models, particularly those built with the U-Net architecture, offer significant advantages in terms of structural detail and precision, which are crucial in medical applications like ophthalmology. The U-Net backbone, with its skip connections, helps preserve the fine-grained information necessary to capture subtle disease symptoms in canine eye images [26].

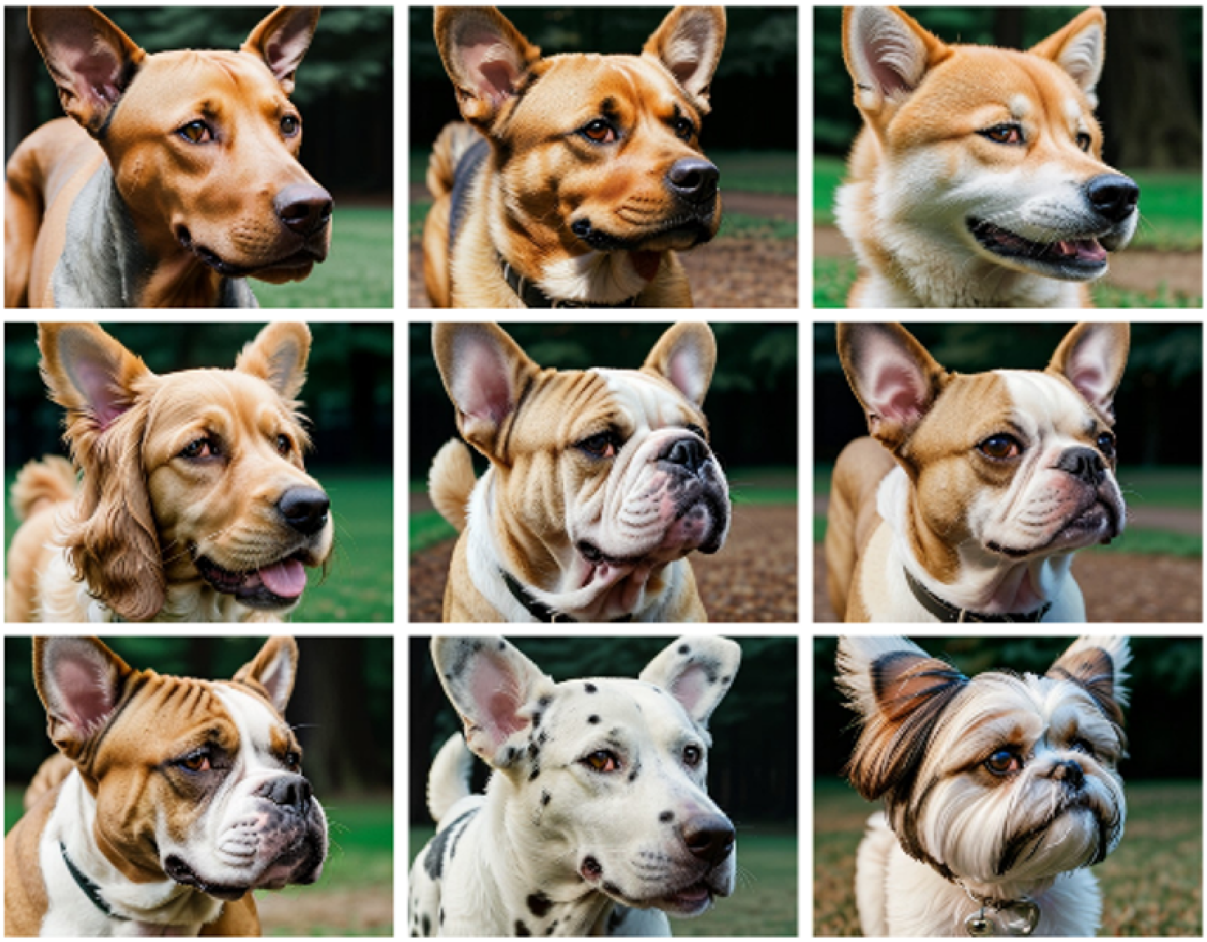


Figure 3.5: Stable diffusion generated images with fixed posture of different dog breeds. Images shows high degree of reality with certain difficulty like double set of ears [74].

Stable Diffusion was further enhanced using Low-Rank Adaptation (LoRA), which allowed for parameter-efficient fine-tuning of the model, reducing both time and computational resources [75]. LoRA training was conducted using 25 randomly selected images per class from the *DogEyeSeg4* dataset, accompanied by descriptive text prompts, such as "Cherry Eye in both eyes, a dog with a pink tongue and a black and brown face." The LoRA model was trained over 10 epochs with a training batch size of 2, and U-Net learning rate of 5e-4. By using a small subset of real-world images from the *DogEyeSeg4* dataset, Stable Diffusion was fine-tuned to specialize in generating images that accurately depict canine ophthalmic diseases like Glaucoma, Cherry Eye, and Uveitis.

Examples of such images using LoRA are presented in Figure 3.6 [74]. An example prompt for generating the bottom left image in Figure 3.6 includes:

- **Positive prompt**: "lora:CherryEye_Lora:1, CherryEye, prolapse in the corner of an eye, realistic image, realistic fur, award-winning wildlife (medium long shot:1.4), 35mm film movie still."
- **Negative prompt**: "abnormal, ugly, abnormal skin."
- **Other parameters**: Steps: 20, Sampler: DPM++ 2M, Schedule type: Karras, CFG scale: 7, Size: 512x512, Model: wildlifexAnimals_wildlifexAnimals.

For each symptom class, 100 images were generated, resulting in 400 additional synthetic images for the DogEyeSeg4 dataset. While rule-based systems and 3D rendering methods provide high control over specific image features, their high manual effort and resource demands made them less practical for large-scale image generation compared to Stable Diffusion [69].

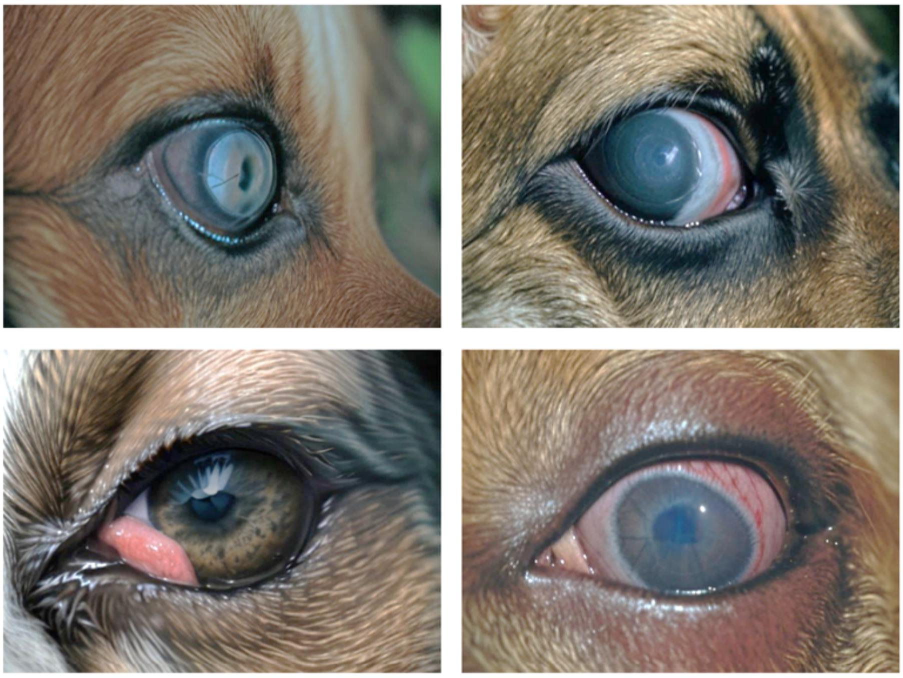


Figure 3.6: Close-up Stable Diffusion images using custom LoRA describing various medical conditions based on DogEyeSeg4 dataset [74].

In diffusion models, the process of image generation involves gradually transforming random noise into coherent images. This denoising process is managed by a U-Net architecture, which effectively captures both high-level structures and fine details. The model is trained to learn how to reverse the noise-adding process, allowing it to generate clean, high-quality images from noisy input during inference [66, 68]. In this study, the fine-tuning of Stable Diffusion was performed using custom LoRA, which reduced memory requirements by updating only a subset of the model's parameters [76]. The use of a U-Net backbone allowed the model to capture the intricate details of various eye diseases, making it particularly effective for generating clinically relevant images of canine ophthalmic conditions.

One of the key features of Stable Diffusion in this study was its ability to add disease symptoms to otherwise healthy eyes using inpainting [77, 78]. Inpainting allows for localized changes to specific regions of an image while leaving the rest of the image unchanged. This feature was valuable for introducing symptoms like scleral redness or corneal cloudiness into healthy eye images, which can be examined in Figure 3.7 [74]. Using positive prompts such as "cloudy cornea with red sclera," the model could generate localized disease manifestations, allowing for the creation of synthetic images that accurately represented various stages of disease progression. This capability enhanced the diversity of the dataset, ensuring that a wide range of symptoms and severities were represented.

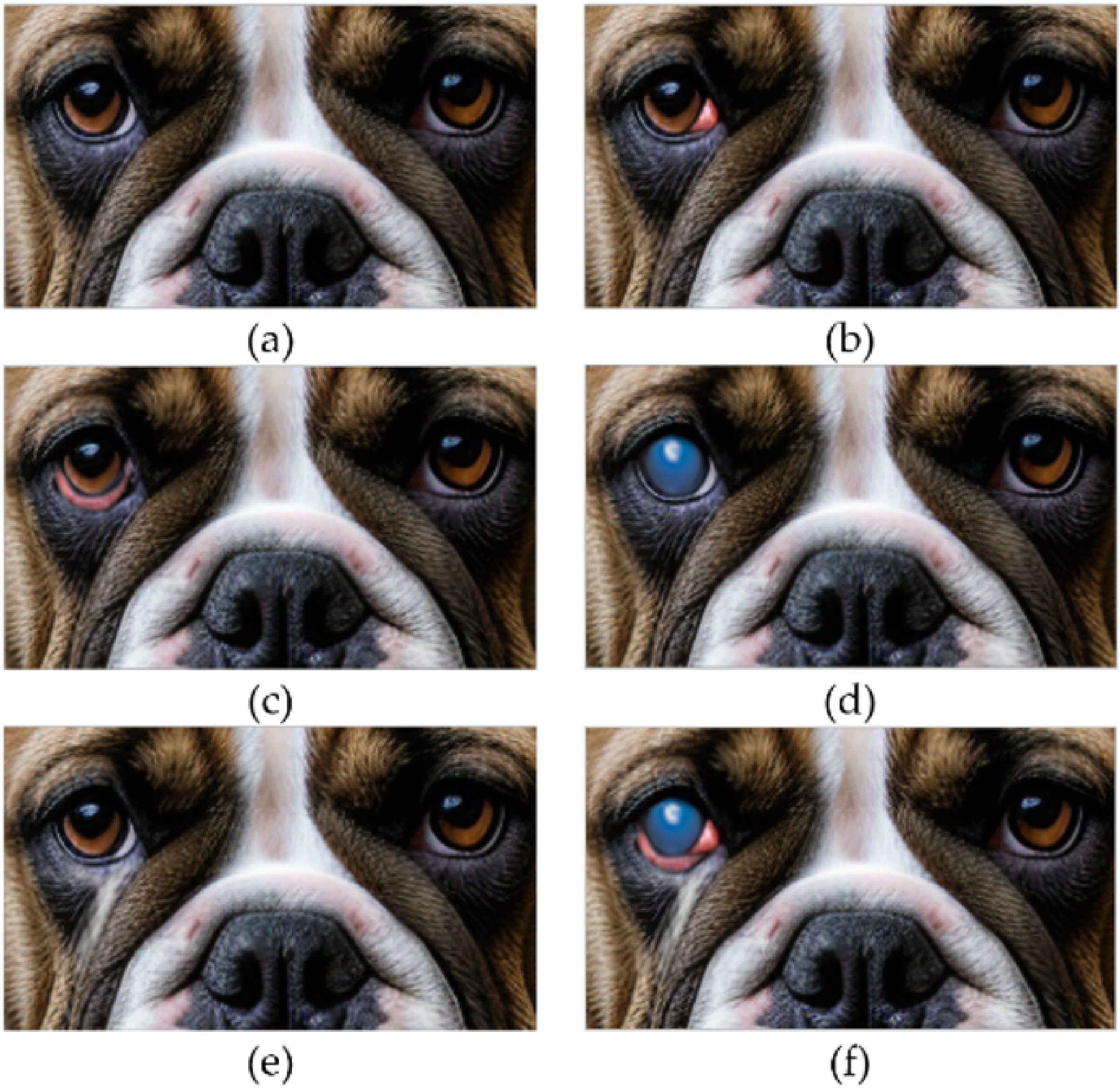


Figure 3.7: Stable Diffusion generated image containing symptoms using Inpainting showing (a) healthy eyes, left eye with: (b) prolapsed eyelid gland, (c) red sclera, (d) cloudy cornea, (e) epiphora and (f) all previously mentioned symptoms [74].

In addition to generating new images, diffusion models offer unique opportunities for augmentation:

- **Symptom Severity Modification**: Adjusting the text prompts allowed for the generation of images with varying degrees of disease severity, from mild to severe symptoms [46].

- **Symptom Combination**: Diffusion models can create images with multiple symptoms, such as excessive tearing and a colored mass, replicating complex real-world cases.
- **Stochastic Variability**: The inherent randomness in diffusion models ensures that even with the same prompt, slight variations occur in the generated images, further diversifying the dataset without requiring additional real-world data collection [45].

Advantages of Using Synthetic Images

- **Scalability of Data**: Synthetic image generation facilitates the creation of extensive datasets, which is particularly valuable for rare conditions or underrepresented canine breeds. This approach addresses the scarcity of real-world data, allowing researchers to simulate a wide range of clinical scenarios that might otherwise be difficult to capture [79].
- **Efficiency in Cost and Time**: Compared to the collection and annotation of real-world images, synthetic data can be generated rapidly and at a significantly lower cost. This enables the efficient scaling of datasets required for training deep learning models, reducing the time and resources associated with manual data acquisition [80].
- **Controlled Variability**: Synthetic generation allows researchers to precisely control the inclusion of specific symptoms, conditions, and their severity. This level of control ensures that the dataset remains balanced, mitigating issues related to class imbalance, and comprehensively covering the spectrum of disease presentations [80].
- **Ethical Benefits**: The use of synthetic images circumvents the need for invasive clinical procedures or additional veterinary visits. As a result, it provides an ethically sound method for expanding datasets without subjecting animals to unnecessary tests or discomfort [48].

Limitations When Synthetic Images Are Used

- **Realism Constraints**: Although synthetic image generation has made significant advancements, the resulting images may still exhibit subtle artifacts or unrealistic features. These imperfections could lead to a degradation in model performance when applied to real-world scenarios, as the models may struggle to generalize from synthetic to actual clinical data [47].
- **Bias Propagation**: Synthetic datasets, while generated artificially, can inadvertently carry over biases from the real-world data used in the fine-tuning process. This issue may limit the generalizability of models trained exclusively on synthetic data, as the diversity and complexity of real-world cases might not be fully captured [44].
- **Necessity for Clinical Validation**: Despite their utility, synthetic images require thorough validation to ensure their clinical relevance. Without rigorous validation processes,

models trained on synthetic data might underperform when deployed in real-world clinical settings, especially when tasked with recognizing nuanced or rare conditions [48].

3.2 Model Architecture and Training

3.2.1 U-Net

The U-Net architecture is widely regarded as a robust model for image segmentation, particularly in the field of biomedical image analysis. U-Net excels at pixel-wise classification tasks. Its architecture consists of two symmetric parts: an encoder (contracting path) and a decoder (expanding path), forming a U-shaped structure that allows the network to both extract features and reconstruct spatial details at the pixel level.

The encoder in U-Net serves to downsample the input image through a series of convolutional and max-pooling layers, extracting increasingly abstract features. Each convolutional block consists of two 3×3 convolutional layers followed by a ReLU activation function. The downsampling occurs through max-pooling layers, which reduce the spatial resolution by a factor of two at each step. This progression allows U-Net to capture the local features in early layers and more complex global patterns in deeper layers.

A visual representation of the U-Net architecture is shown in Figure 3.8 [62]. At the deepest point in the architecture, the bridge links the encoder and decoder by combining features learned from the encoder's deepest layer with features from the decoder's shallowest layer. This design ensures that global and local features are propagated through the network, facilitating accurate segmentation even in cases of complex images with small or subtle regions of interest [81]. The combination of these feature maps is essential for generating detailed and contextually accurate segmentations, particularly in medical applications where high precision is critical.

The decoder mirrors the encoder, progressively upsampling the feature maps and restoring the spatial resolution of the input image. The decoder combines the upsampled feature maps with the feature maps from the corresponding encoder layers via skip connections. These skip connections help retain the spatial context and fine details from the encoder, ensuring that the segmentation is accurate at a pixel level [26, 82]. The decoder uses transpose convolutions (deconvolutions) to increase the image resolution and further refine the output segmentation mask [83].

The final output is a pixel-wise segmentation mask, where each pixel is assigned a class label based on the features extracted by the encoder and decoder. U-Net has proven to be effective in tasks that require accurate segmentation with limited training data, making it particularly well-suited for medical imaging, where annotating large datasets is often challenging [26, 82]. Its symmetrical structure and use of skip connections make it highly capable of handling both large and small objects in an image, and it performs well even when there is significant variation in

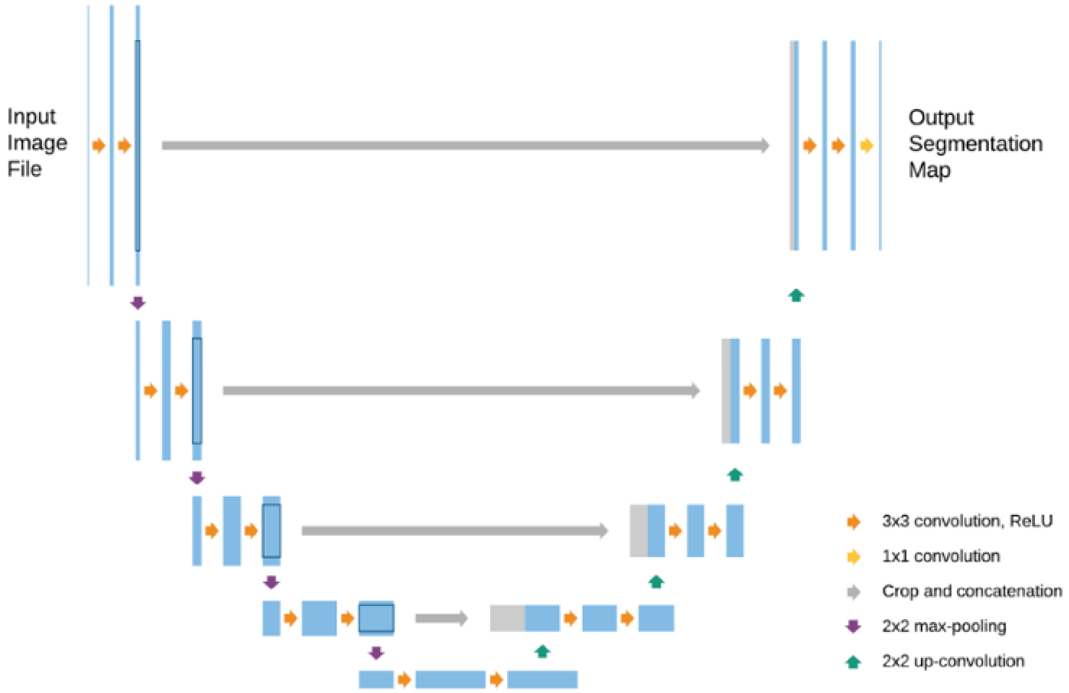


Figure 3.8: The standard U-Net architecture where left encoder part and right decoder part form the letter U. The encoder effectively captures detailed features from the input images, which are then upsampled by the decoder part of U-Net. The decoder uses transposed convolutions to restore the image to its original size, using skip connections from the corresponding encoder layers to refine the segmentation output with high-resolution details retained [62].

object size and shape.

U-Net with modified Backbone

Transfer learning significantly enhances the capability of deep learning models, particularly in specialized tasks such as medical image segmentation, by leveraging pre-trained neural networks. The U-Net architecture benefits from incorporating robust feature-extracting networks, known as backbones, which are pre-trained on large and diverse datasets like ImageNet [84]. These backbones capture a wide array of features, from simple textures to complex structures, which are crucial for accurate segmentation.

One popular choice for a backbone in U-Net is the VGG network [39], known for its simplicity and effectiveness in feature extraction due to its deep architecture of 16 convolutional layers. The typical VGG-16 architecture, as shown in Figure 3.9 [62], consists of multiple convolutional layers followed by max-pooling layers, which progressively reduce the spatial dimensions while increasing the depth, capturing finer details at each layer.

Another powerful backbone is the ResNet18 [85], which incorporates residual connections to allow training of much deeper networks by addressing the vanishing gradient problem. These residual connections act as shortcuts that enable the gradient to flow through the network without diminishing, preserving the strength of the signal (Figure 3.10 [62]). This architecture

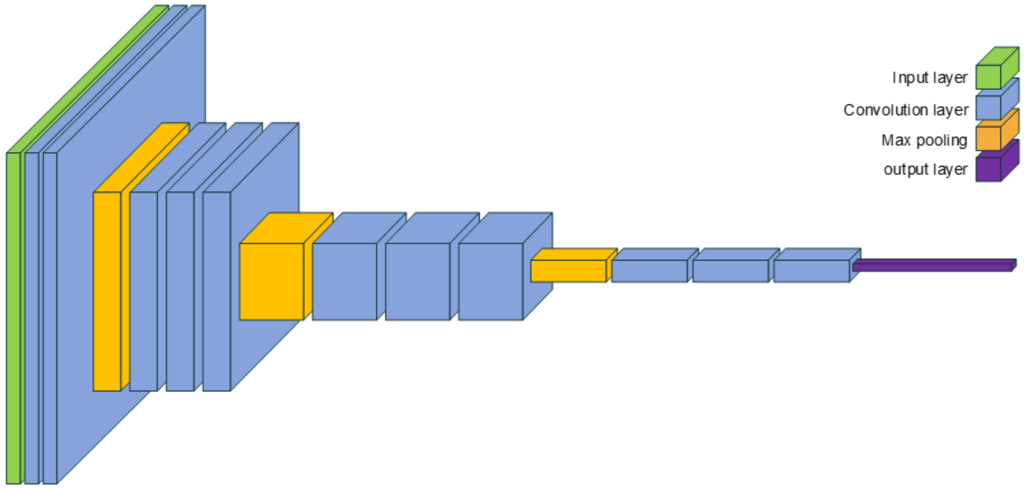


Figure 3.9: The VGG-16 architecture [62].

is particularly beneficial for U-Net, enhancing its ability to learn from medical images where preservation of spatial hierarchies is critical.

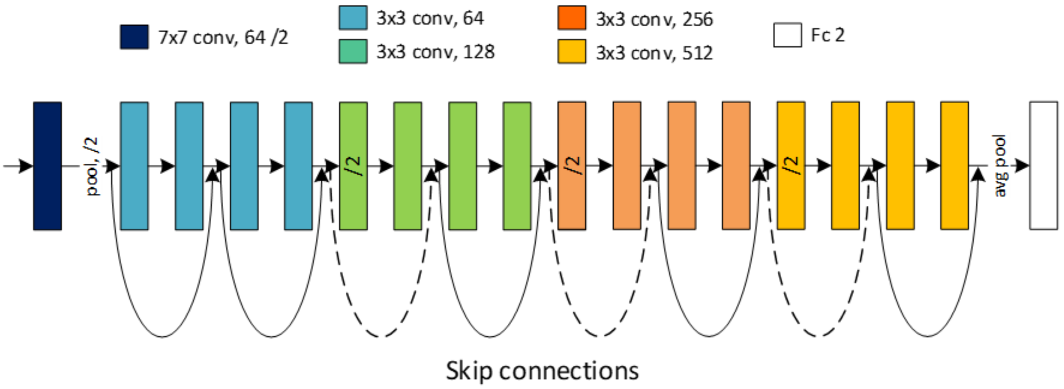


Figure 3.10: The ResNet18 architecture [62].

The Inception V3 network [86], known for its efficiency in handling multi-scale information, uses modules composed of parallel convolutional layers with varying kernel sizes. This design allows the network to capture features at various scales within the same layer, making it highly effective for tasks that require the detection of objects of different sizes, such as different types of tissues or cells in medical images. The Inception V3 network is shown in Figure 3.11 [62].

Improved U-Net with Backbone Integration: The improved U-Net model incorporates these backbone networks into its architecture, where each encoder block is an instance of either VGG-16, ResNet18, or Inception V3. This integration allows the U-Net to utilize advanced feature extraction capabilities of these networks, enhancing its performance significantly[87]. The decoder of the improved U-Net model up-samples the feature maps and combines them with the extracted features at corresponding levels from the encoder, optimizing the segmentation accuracy as seen in Figure 3.12 [62].

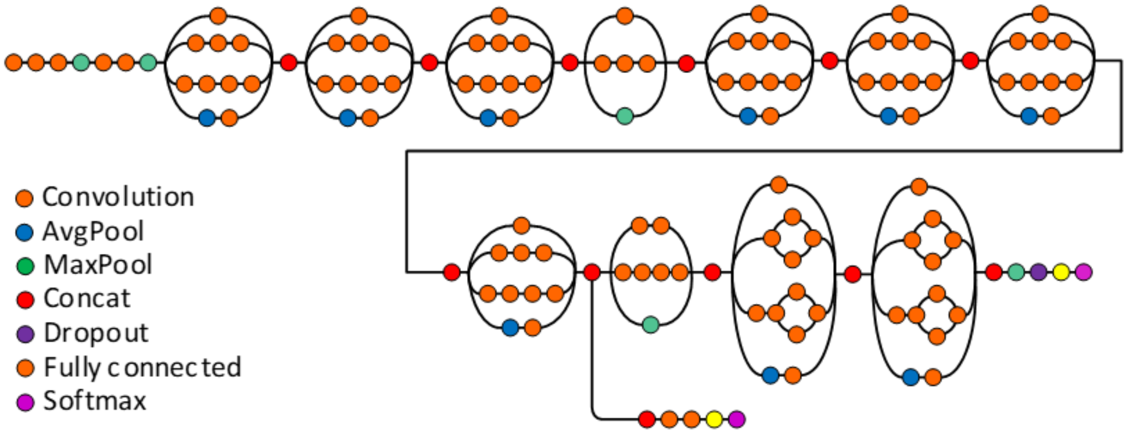


Figure 3.11: The Inception V3 architecture [62].

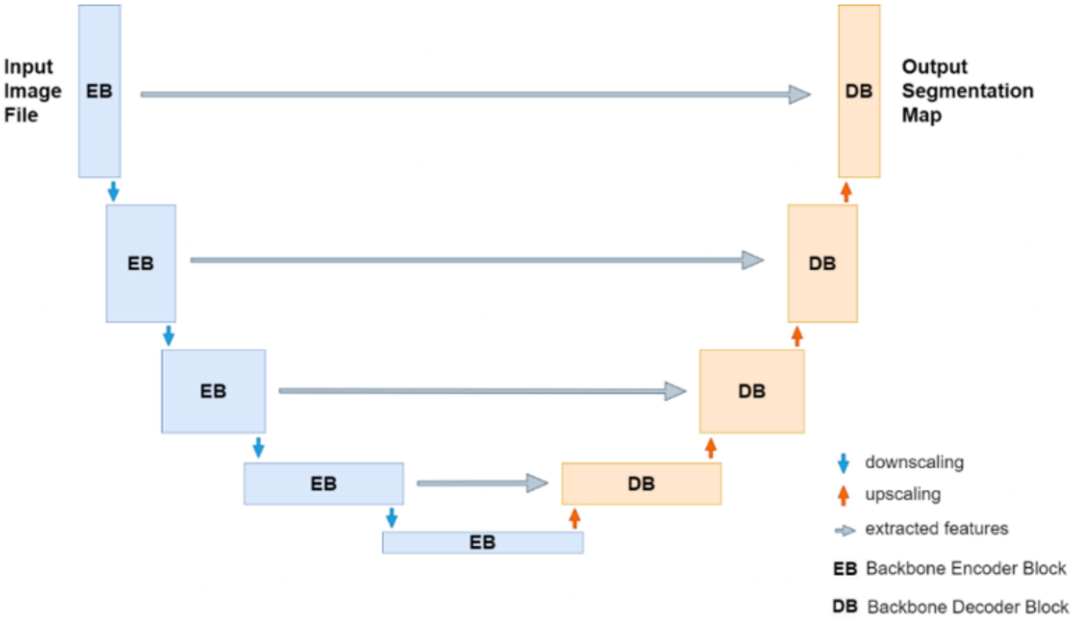


Figure 3.12: Schematic of the enhanced U-Net architecture, integrating a backbone (e.g., VGG, ResNet) within both encoder and decoder blocks. The Encoder Blocks (EB) incorporate convolutional layers from the backbone, downscaling the input while extracting progressively complex features. Only the feature-extracting layers of the backbone are used, omitting fully connected layers. The Decoder Blocks (DB) then upsample these features, using skip connections from corresponding EBs to preserve high-resolution details, ultimately reconstructing the output image [62].

U-Net Algorithm

The training of the U-Net model involves an iterative process of adjusting network weights to minimize the loss between the predicted segmentation masks and the ground truth. The algorithm begins by initializing the weights of both the encoder and decoder networks randomly. The model is then trained using a dataset consisting of training images and corresponding ground truth segmentation masks.

For each training image, the following steps are performed in the training loop:

1. The encoder network encodes the input image, extracting a set of feature maps at different levels of abstraction.
2. The feature maps from the encoder are concatenated with the corresponding feature maps from the bridge, which combines contextual information from both high- and low-level layers of the network.
3. The concatenated feature maps are then passed through the decoder network, which progressively upsamples the feature maps and reconstructs the spatial resolution to generate a predicted segmentation mask.
4. The loss between the predicted segmentation mask and the ground truth segmentation mask is calculated using the selected loss function (such as Dice loss or cross-entropy loss).
5. The encoder and decoder network weights are updated using an optimization algorithm (e.g., stochastic gradient descent or Adam) to minimize the loss.

The algorithm continues to iterate through the training images, updating the network weights after each image and its corresponding mask are processed. After each training loop, a check is performed to evaluate the convergence of the model. If the loss converges or reaches a satisfactory level, the training process is terminated. Otherwise, the loop continues until the model reaches optimal performance, as described in Algorithm 1.

Algorithm 1 U-Net Algorithm for Segmentation [62]

```
Require: Training images and corresponding ground truth segmentation masks
Ensure: Trained U-Net model
Initialize encoder and decoder network weights randomly
while not converged do
    for each training image and ground truth segmentation mask do
        Encode image using encoder network
        Concatenate encoder feature maps with bridge feature maps
        Decode concatenated feature maps using decoder network to generate predicted segmentation mask
        Calculate loss between predicted and ground truth segmentation masks
        Update encoder and decoder network weights using an optimization algorithm
    end for
end while
```

Once the model is trained, it can be used to predict segmentation masks for new images. The prediction involves passing the input image through the encoder and decoder, generating a probability map for each pixel in the image. The class with the highest probability is assigned to each pixel, resulting in the final segmentation output. This process ensures that the U-Net model learns to accurately segment images based on the features extracted during the training phase, providing robust pixel-wise predictions for medical imaging tasks.

Loss Function

In deep learning, loss functions are essential components that assess the disparity between a model's predicted output and the actual ground truth values. A higher loss indicates greater error in the model's predictions, while a lower loss suggests improved accuracy. The principal objective of loss functions is to guide the model toward minimizing prediction errors by adjusting its trainable parameters, such as weights and biases.

There are numerous loss functions available, each designed to reward or penalize a model differently, depending on the nature of the task. The choice of the most appropriate loss function is critical for achieving optimal performance. In this research, three primary loss functions were used: Dice Loss (DL), Cross Entropy (CE) loss, and Focal Loss (FL), all of which are highly relevant for segmentation tasks in medical imaging.

Dice Similarity Coefficient (DSC) is a statistical measure that evaluates the similarity between two samples. In the context of image segmentation, it quantifies the overlap between the predicted segmentation $\hat{y}$ and the ground truth mask y. For binary classification tasks, the DSC is defined as:

$$DSC = \frac{2\sum_{i=1}^{N} y_i \cdot \hat{y}_i}{\sum_{i=1}^{N} y_i + \sum_{i=1}^{N} \hat{y}_i}, \tag{3.1}$$

where:

- N is the total number of pixels in the image.
- y_i represents the ground truth label at pixel i (1 for foreground, 0 for background).
- $\hat{y}_i$ represents the predicted probability at pixel i (ranging from 0 to 1).

The DSC ranges between 0 and 1, where 1 indicates perfect overlap and 0 indicates no overlap.

Since maximizing the DSC is equivalent to minimizing $1 - DSC$, we define the **Dice Loss (DL)** for binary classification as:

$$DL(y, \hat{y}) = 1 - DSC \tag{3.2}$$

Dice Loss is particularly effective when working with small, imbalanced regions of interest, as it directly optimizes the overlap between the predicted and actual segmentation masks.

To further understand DSC, it's important to relate it to the concepts of precision and recall, which are fundamental in evaluating classification performance. Precision and recall are defined as:

$$\text{Precision} = \frac{TP}{TP + FP}, \quad \text{Recall} = \frac{TP}{TP + FN}, \tag{3.3}$$

where:

- TP (True Positives) is the number of correctly predicted positive pixels.

- FP (False Positives) is the number of incorrectly predicted positive pixels.
- FN (False Negatives) is the number of positive pixels that were incorrectly predicted as negative.

Using these definitions, the Dice Similarity Coefficient is mathematically equivalent to the **F1-score**, which is the harmonic mean of precision and recall:

$$DSC = F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{3.4}$$

This is true for binary classification tasks where the DSC and F1-score yield the same numerical value, measuring the balance between precision and recall.

In this research, Dice Loss was used to transform this metric into a continuous, differentiable function that can be applied during the training of the model. The differentiability of Dice Loss makes it suitable for optimization via gradient-based methods.

For **multi-class** segmentation problems, which is also used in this research, Dice Loss can be extended by computing the loss for each class separately and then averaging:

$$DL(y, \hat{y}) = 1 - \frac{1}{C} \sum_{c=1}^{C} \frac{2 \sum_{i=1}^{N} y_{i,c} \cdot \hat{y}_{i,c}}{\sum_{i=1}^{N} y_{i,c} + \sum_{i=1}^{N} \hat{y}_{i,c}}, \tag{3.5}$$

where:

- C is the number of classes.
- $y_{i,c}$ is the ground truth label for class c at pixel i (1 if pixel i belongs to class c, 0 otherwise).
- $\hat{y}_{i,c}$ is the predicted probability for class c at pixel i.

Cross Entropy (CE) Loss is another widely used loss function in classification tasks. It measures the divergence between the predicted probability distribution and the actual labels. For **binary classification**, the **Binary Cross Entropy (BCE)** loss is defined as:

$$BCE(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^{N} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)], \tag{3.6}$$

where:

- y_i is the ground truth label at pixel i (1 or 0).
- $\hat{y}_i$ is the predicted probability at pixel i (ranging from 0 to 1).

Binary Cross Entropy Loss penalizes the model when the predicted probabilities deviate from the actual labels. It is particularly effective for binary classification tasks and is commonly used due to its smooth and convex nature, which facilitates optimization.

For **multi-class** classification problems, which involve more than two classes, the **Categorical Cross Entropy (CCE)** loss is used. The formula for CCE is:

$$CCE(y,\hat{y}) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\log(\hat{y}_{i,c}), \tag{3.7}$$

where:

- $y_{i,c}$ is the ground truth label for class c at pixel i.
- $\hat{y}_{i,c}$ is the predicted probability for class c at pixel i.

CCE measures the divergence between the predicted probability distribution over classes and the actual one-hot encoded labels.

However, both BCE and CCE can struggle in imbalanced class distributions where larger classes dominate the loss, affecting the segmentation performance of smaller classes. To address this imbalance, **Focal Loss (FL)** was introduced. Focal Loss builds upon Cross Entropy by focusing more on hard-to-classify examples, particularly those incorrectly classified or with low confidence. This is achieved by down-weighting well-classified examples and focusing the learning process on challenging ones. For multi-class classification, Focal Loss modifies the Categorical Cross Entropy Loss as:

$$FL(y,\hat{y}) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} \alpha_c(1-\hat{y}_{i,c})^{\gamma} y_{i,c}\log(\hat{y}_{i,c}), \tag{3.8}$$

where:

- α or α_c is a weighting factor for balancing the importance of different classes.
- γ (gamma) is the focusing parameter that adjusts the rate at which easy examples are down-weighted.

By incorporating the modulating factor $(1-\hat{y}_{i,c})^{\gamma}$, Focal Loss places more focus on misclassified examples (where $\hat{y}_{i,c}$ is low) and less on well-classified ones (where $\hat{y}_{i,c}$ is high). When $\gamma = 0$, Focal Loss reduces to the standard Cross Entropy Loss (CCE).

In this research, the impact of individual loss functions and in combination provides valuable insight in finding most successful training model.

Activation Functions

Activation functions are an essential component of neural networks, determining the output of each neuron and introducing non-linearity into the network. This non-linearity enables the network to learn complex patterns and make accurate predictions. In U-Net and similar convolutional neural networks, common activation functions include ReLU, Sigmoid, and Softmax [88].

The **ReLU (Rectified Linear Unit)** activation function is defined as:

$$ReLU(x) = \begin{cases} x, & \text{if } x > 0 \\ 0, & \text{if } x \leq 0 \end{cases} \tag{3.9}$$

ReLU is widely used due to its simplicity and effectiveness, as it avoids the vanishing gradient problem that can occur with other activation functions like Sigmoid [89].

The **Sigmoid** activation function, defined as:

$$Sigmoid(x) = \frac{1}{1+e^{-x}}, \tag{3.10}$$

is often used in the final layer of binary classification models. It maps the output to a value between 0 and 1, making it suitable for probability estimation. However, Sigmoid suffers from the vanishing gradient problem, which can slow down training [90].

For multi-class segmentation tasks, **Softmax** is typically used in the final layer. Softmax normalizes the output probabilities for each class, ensuring they sum to 1. It is defined as:

$$Softmax(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}}, \tag{3.11}$$

where x_i represents the input to the i-th class. Softmax is particularly useful for multi-class classification, as it allows the network to assign probabilities to multiple classes and select the most likely class for each pixel [91].

3.2.2 GPT-4 and GPT-4o

Large Language Models (LLMs) and their various iterations, such as different versions of GPT (e.g., GPT-4, GPT-4o), Mistral, Gemini, Claude, and Llama, have demonstrated significant potential in medical diagnostics by aiding in symptom analysis and preliminary diagnosis. Each of these models brings unique strengths to processing medical data. For instance, ChatGPT excels at handling complex medical data and producing fluent, context-rich responses, while Mistral is specifically designed for efficient query resolution. Similarly, Gemini adapts well to dynamic conversational settings, and Claude offers a well-rounded approach to general data analysis. Despite these advantages, most of these models are challenged when tasked with complex, multi-symptom medical scenarios, as their architecture primarily focuses on textual understanding rather than deep multimodal analysis [80].

The introduction of **GPT-4** marks a notable advancement over these earlier models. While the previously mentioned LLMs excel at interpreting symptom descriptions and providing initial diagnoses, GPT-4 surpasses them with its ability to incorporate context-driven responses and integrate detailed multimodal inputs [92]. More importantly, its evolved version, **GPT-4o**, extends the capabilities of the original model by incorporating object detection and segmentation tasks, which are critical in image-based diagnostics. This enhancement makes GPT-4o particularly valuable for tasks requiring a combination of natural language processing and detailed image analysis, as seen in veterinary ophthalmology diagnostics.

The GPT-4 model, recognized for its robust performance in language-related tasks, has been innovatively adapted for image analysis, deviating from its primary design as a text-processing engine. This unconventional application involves using images as input, where the model is expected to generate outputs that align with the visual content observed, rather than textual data. This method leverages the model's latent ability to handle multimodal tasks despite its foundational design as a language model [93].

While GPT-4 exhibits capabilities in image classification, it inherently lacks direct functionalities for object detection and segmentation—key components in detailed image analysis. Addressing these limitations, the enhanced version, GPT-4o, introduces mechanisms to perform these tasks by generating segmentation masks akin to those in ground truth data of test images, marking a significant enhancement over its predecessor [74]. The core architecture of GPT-4 is built upon the Transformer model, specifically tailored for language generation but applicable to a broader range of sequence processing tasks. This architecture is illustrated in Figure 3.13 [74] and encompasses several critical components:

- **Embedding Layer**: Converts input tokens, for example, words in a sentence like "The dog chased the", into rich, informative embeddings, supplemented with positional encodings to preserve the sequential nature of the input data. This embedding process preserves the order of the tokens, essential for accurate understanding of the input sequence.
- **Multi-Head Attention**: This mechanism allows the model to dynamically focus and refocus on different segments of the input data, facilitating a deeper understanding of the contextual interdependencies. For instance, when processing a sentence, the attention mechanism might focus on "dog" and "chased" simultaneously to understand their relationship.
- **Feed-Forward Networks**: Positioned to process the data sequentially from the attention mechanisms, these networks are pivotal in refining the data handling.
- **Residual Connections and Layer Normalization**: These components are integral for maintaining the flow of gradients during training, thus enhancing the stability and performance of the model.
- **Output Layers**: Comprising a linear layer followed by a softmax activation, these layers produce the output, which is a probability distribution over possible next tokens. For example, given the input "The dog chased the," the model might predict "cat" with the highest probability, completing the sentence as "The dog chased the cat."

To effectively deploy GPT-4 and GPT-4o in veterinary ophthalmology, specific contextual adaptations are necessary. These models are configured to act as theoretical consultants, analyzing visual data from images and predicting possible veterinary conditions. The setup demands a tai-

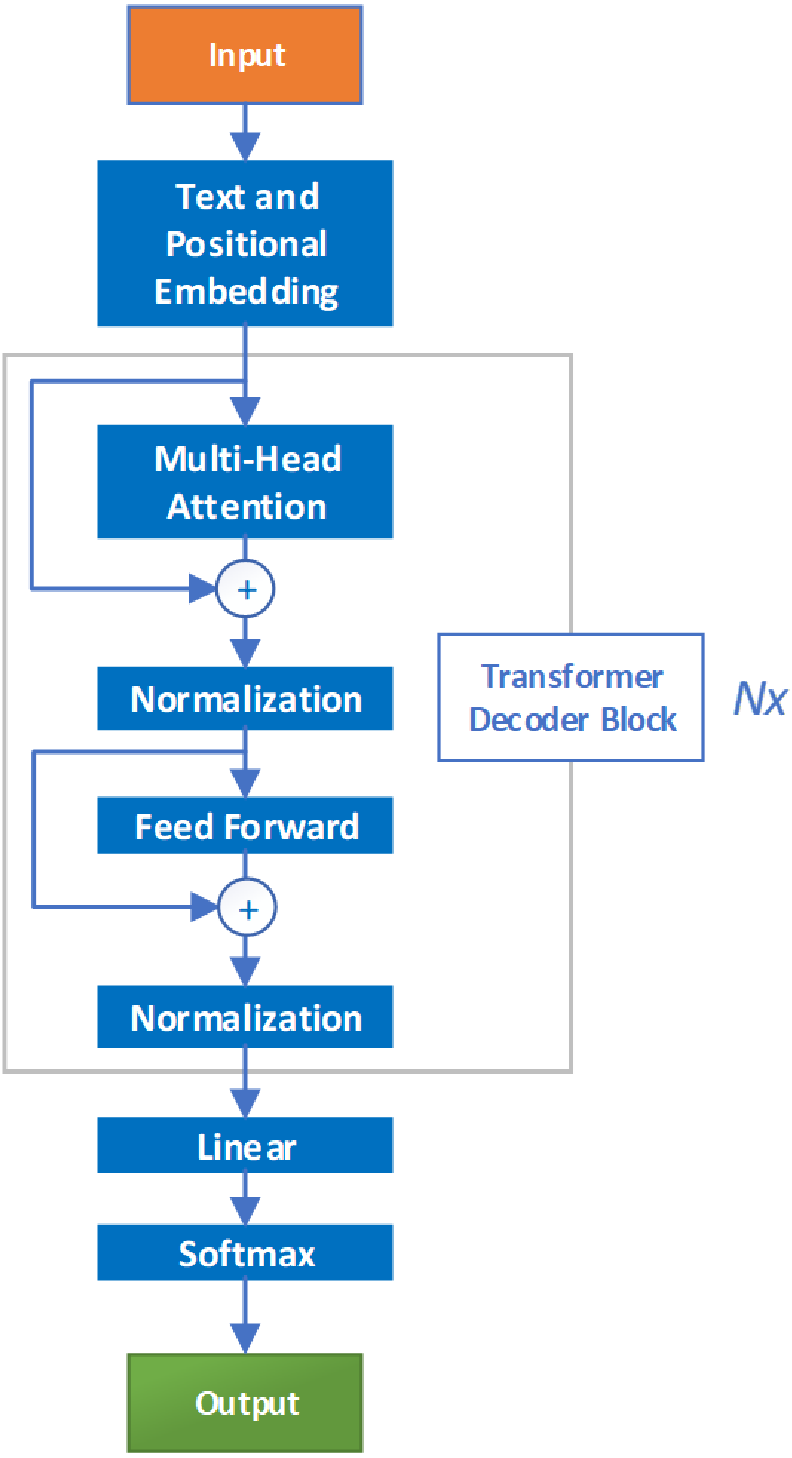
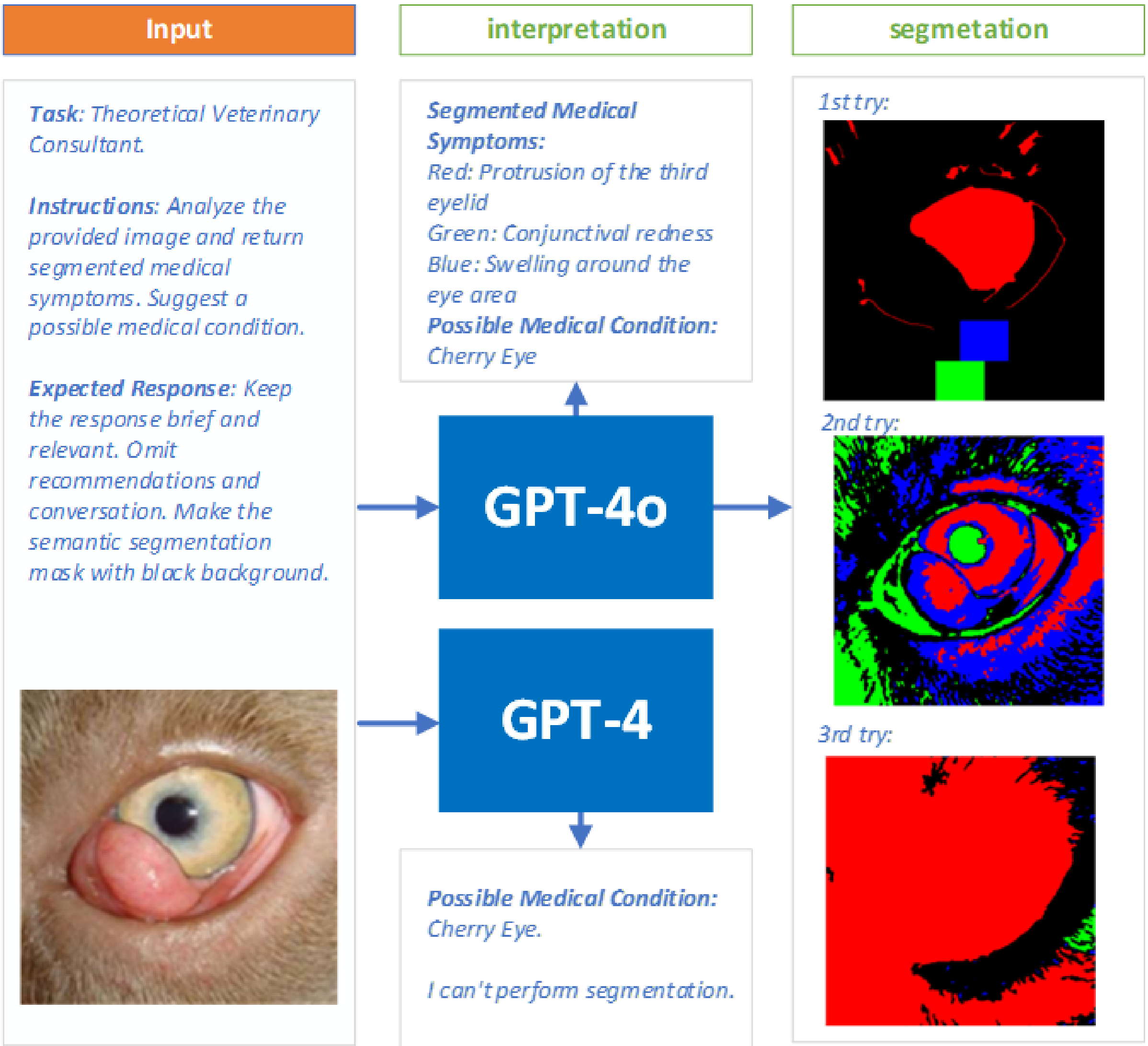


Figure 3.13: Architecture of a Transformer Decoder block in the GPT model. The process begins with the input—text tokens with positional information, transformed into vector embeddings encoding both meaning and position. These embeddings pass through layers of multi-head attention and feed-forward networks, with normalization applied for stability. Multiple Transformer Decoder Blocks (denoted as Nx) capture contextual relationships. The model's output is a probability distribution over possible next tokens, generated by a linear layer and softmax function, enabling text generation based on the input sequence. [74].

lored prompt that provides the models with a context encapsulating the task of diagnosing based on visual cues alone, without supplementary background information. An exemplary prompt used in this scenario is demonstrated in Figure 3.14 [74], where the models' output includes diagnostic suggestions and segmentation masks, crucial for practical applications [94].



Task: Theoretical Veterinary Consultant.

Instructions: Analyze the provided image and return segmented medical symptoms. Suggest a possible medical condition.

Expected Response: Keep the response brief and relevant. Omit recommendations and conversation. Make the semantic segmentation mask with black background.

Segmented Medical Symptoms:
Red: Protrusion of the third eyelid
Green: Conjunctival redness
Blue: Swelling around the eye area
Possible Medical Condition: Cherry Eye

Possible Medical Condition: Cherry Eye.

I can't perform segmentation.

Figure 3.14: An example of input prompt to GPT-4 and GPT-4o requesting segmentation of medical conditions and most probable diagnosis [74].

The outcomes derived from this innovative application of GPT-4 and GPT-4o are directly compared against traditional methods, providing a benchmark for assessing the efficacy of integrating language models in image-based diagnostic tasks. This comparative analysis is essential for validating the models' utility and effectiveness in real-world scenarios [74].

3.2.3 Grounding SAM

Grounding SAM, a combination of Grounding DINO [6] and SAM [7], provides a versatile approach for object detection and segmentation driven entirely by text prompts rather than domain-specific pre-training. This prompt-based approach removes the need for extensive, customized datasets and enables applications where labeled data are limited or hard to obtain. Grounding SAM leverages zero-shot learning capabilities, enabling it to adapt across tasks and images without retraining, setting it apart from traditional models like YOLO [95] and Mask R-CNN [96], which rely on task-specific data and fine-tuning for high accuracy.

Grounding DINO stands for "Grounding DEtection with TRansformers and Open-vocabulary using Denoising." It is a transformer-based object detection model that extends the original DETR (Detection Transformer) architecture by incorporating improved denoising techniques. Grounding DINO can detect objects in images based on textual descriptions, effectively grounding language in visual data. The model is capable of associating specific parts of an image with words in a text prompt, allowing for open-vocabulary object detection without the need for extensive labeled datasets.

Grounding DINO is the first step in the Grounding SAM framework, responsible for object detection. It operates by taking both images and text prompts as input. The transformer-based architecture of Grounding DINO enables it to process relationships between various objects in the image through its self-attention mechanism. This process allows the model to focus on relevant parts of the image, enhancing its ability to detect objects even in visually complex or noisy environments. The text prompts provided to Grounding DINO are essential for guiding the object detection process. For instance, in the case of diagnosing canine eye diseases, text prompts describing symptoms in layman's terms (e.g., "excessive tearing") are used, as medical terminology might not yield effective results due to the model's limitations in domain-specific vocabulary.

Once the objects have been detected by Grounding DINO, the bounding boxes are passed to SAM, which stands for "Segment Anything Model." SAM is designed to generate accurate segmentation masks for any object in an image based on prompts. It utilizes a promptable segmentation approach, where prompts can be in the form of points, bounding boxes, or text descriptions. SAM's architecture includes a transformer-based image encoder that produces segmentation masks with high precision, even in complex scenes.

SAM's task is to convert these bounding boxes into precise segmentation masks, which provide pixel-level accuracy for each detected object. SAM utilizes a transformer-based image encoder that can handle a wide range of segmentation tasks with minimal human intervention, making it highly versatile for medical applications, where segmentation accuracy is paramount.

In veterinary ophthalmology, SAM's segmentation masks are particularly useful for delineating fine details of ocular symptoms, such as corneal cloudiness or scleral redness. This level of precision in image segmentation ensures that even subtle abnormalities can be detected and analyzed, providing critical data for diagnostic purposes. SAM's ability to generate accurate segmentation masks in a zero-shot manner—without the need for task-specific training—positions it as a powerful tool in the medical domain.

The combined operation of Grounding DINO and SAM is depicted in Figure 3.15 [74], which illustrates the process of diagnosing canine ocular conditions. The input image, typically a clinical photograph of a dog's eye, is first processed by Grounding DINO, which identifies areas of interest based on the provided text prompt. Grounding DINO generates bounding boxes around the detected areas, such as the eye or tear ducts, and these are passed to SAM for detailed segmentation. SAM then produces segmentation masks that outline the detected regions with pixel-level precision. The segmented regions, such as areas of corneal opacity or excessive tearing, are used as input for further analysis.

A significant challenge in applying Grounding SAM to medical image analysis is prompt engineering, particularly when dealing with complex medical terminology. Grounding DINO and SAM require prompts that can be interpreted accurately by the model. Medical terms such as "epiphora" or "keratitis" may not yield effective segmentation results due to their specialized nature. Instead, prompts need to be rephrased in simpler terms, such as "watery eyes" or "eye inflammation," to ensure the model can process and detect the relevant symptoms. Figure 3.16 [74] demonstrates how the structure and content of the text prompt significantly influence the model's output, highlighting the importance of careful prompt formulation.

3.2.4 Single Shot Multibox Detector (SSD) for Eye Detection

The Single Shot Multibox Detector (SSD) [97] was utilized to pinpoint the eye region in canine images, serving as an assisting tool for the improved segmentation task. SSD stands out for its real-time performance, accomplishing object localization and classification simultaneously in a single forward pass of the network. This efficiency is achieved by omitting a separate region proposal generation stage, a common feature in other detection systems like Mask R-CNN, which simplifies the detection process and reduces computational overhead.

The architecture of SSD is built upon a robust base convolutional neural network, which is employed for feature extraction from the input image. Following this feature extraction phase, SSD utilizes additional convolutional layers to produce feature maps at multiple scales. This multi-scale feature extraction allows SSD to detect objects of various sizes effectively. Key components of the SSD workflow include:

- **Feature Extraction**: Utilizes a pretrained network like VGG16 or ResNet to derive rich feature maps from the input image. These features form the foundation for detecting objects at various scales.
- **Detection Heads**: Each feature map level has associated detection heads that output scores and bounding boxes for potential objects at that scale.
- **Non-Maximum Suppression (NMS)**: This step refines the detection by removing overlapping boxes, ensuring that each detected object is represented by the single best bounding box.

The process flow of the SSD model is illustrated in Figure 3.17, which clarifies the stages from

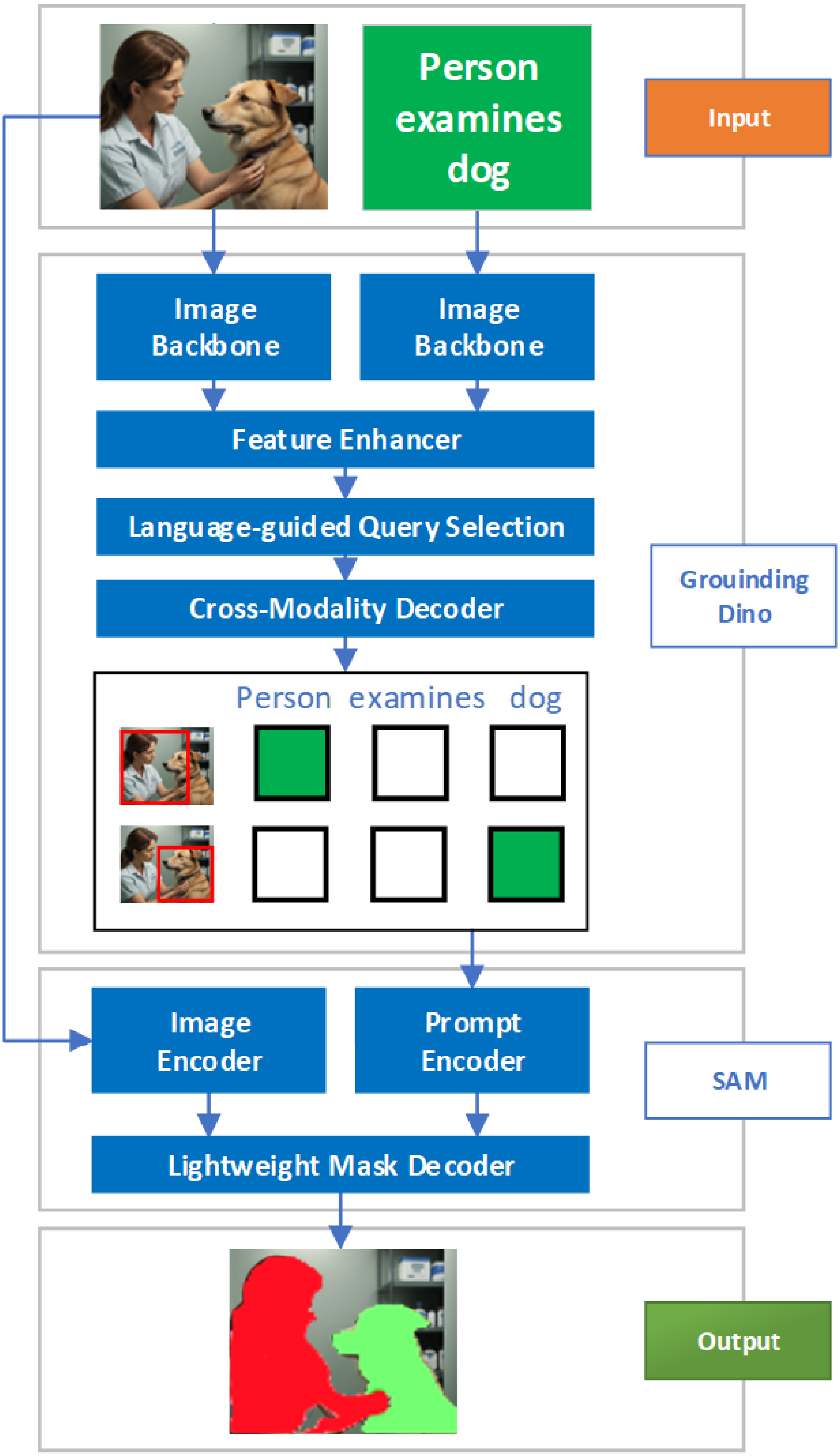
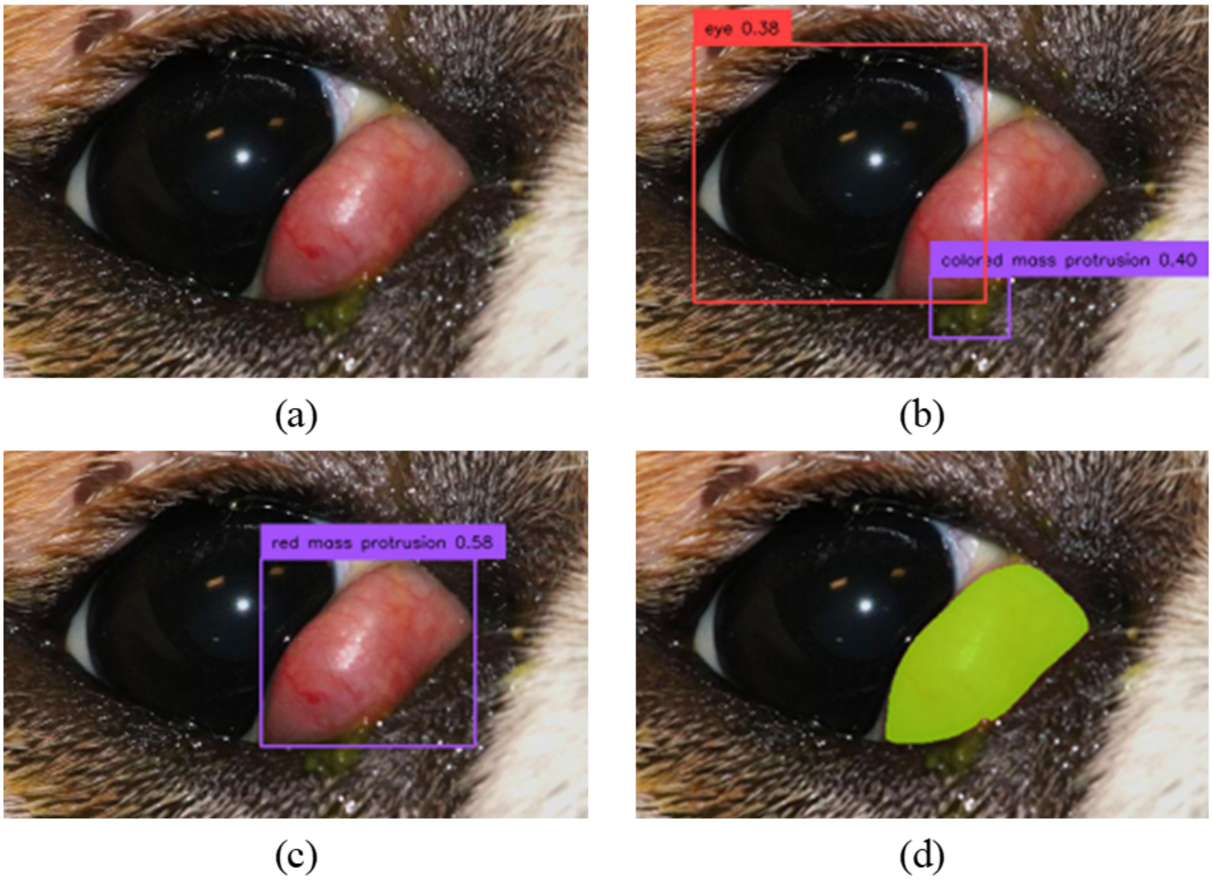


Figure 3.15: Grounding Dino and SAM integration flowchart [74].



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

Figure 3.16: Grounding Dino prompting examples showing diversity with resulting detection and its prediction score. Prompts used for generating detection from top to bottom are: (a) "colored mass protrusion in the corner of the eye", (b) "colored mass protrusion in the eye" and (c) "red mass protrusion". Third prompt (c) is used for segmentation by SAM implementation (d) [74].

image input through to the final predictions. The diagram shows how the image is processed through the VGG16/ResNet layers for feature extraction, followed by multi-scale detection heads, and culminating in the NMS stage to produce the final object predictions.

Given the absence of publicly available models for detecting canine eyes, a custom SSD model was trained specifically for this task. The dataset consisted of 99 images of dog eyes from various breeds and medical conditions, including healthy eyes, each annotated with ground truth bounding boxes in XML format. From this dataset, 20 images were set aside for testing, while the remaining 79 images were used for training.This approach was designed to ensure that the model could accurately identify eye regions across diverse scenarios.

The performance of the custom-trained SSD model was evaluated with next outcomes. The model achieved an average Intersection over Union (IoU) of 0.92 with a standard deviation of 0.08, and an average Dice Score of 0.96 with a standard deviation of 0.05. An F1 score of 0.90, calculated using Precision (0.97) and Recall (0.84), defines a well-suited detection model. These results highlight the SSD model's high precision in localizing the eye region within the canine images, confirming its effectiveness for this specific application in veterinary ophthal-

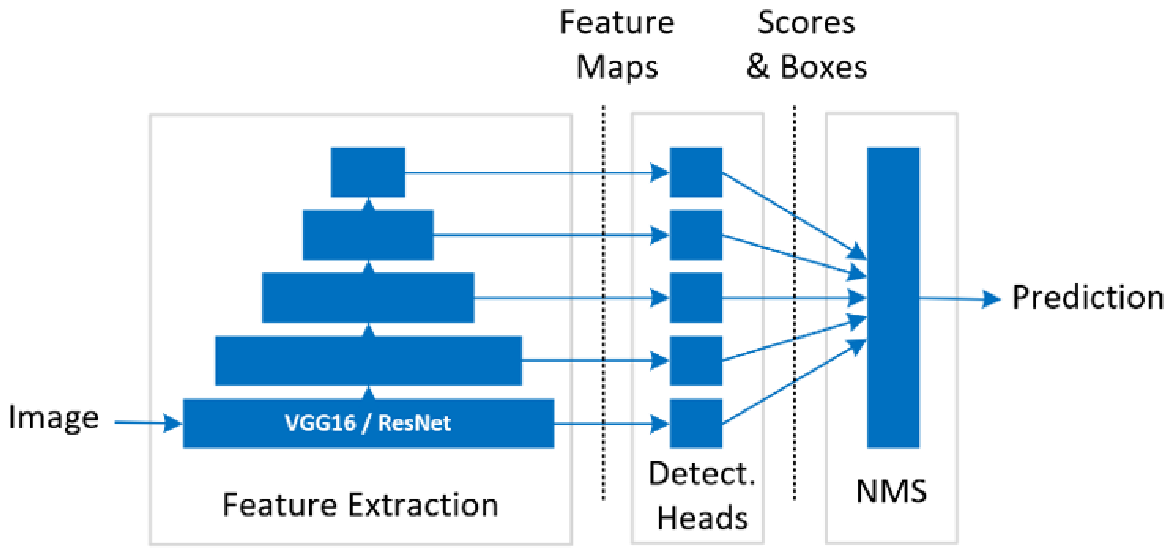


Figure 3.17: Single Shot Multibox Detector diagram for eye detection.

mology. While the model demonstrates strong performance metrics, occasional detection inaccuracies underscore the need for ongoing adjustments and improvements in model training.

3.3 Web Application Development

The development of the web application for eye symptoms segmentation and medical condition prediction focuses on providing a robust, user-friendly interface to assist veterinarians and pet owners in diagnosing canine ocular diseases using AI-based image analysis. The web application is built to integrate the suitable segmentation model, ensuring seamless deployment, accessibility, and scalability.

The development process is initiated by designing an intuitive user interface that would guide users through the image acquisition and diagnostic process. As seen in Figure 3.18, the system starts with a user interface that allows users to either capture images of their dog's eyes or upload them from local storage.

This flexibility accommodates both veterinary professionals and pet owners who may not have prior experience with such diagnostic tools. The quality and framing of these images are crucial for accurate segmentation [62], so the interface should include an overlay mask, as shown in Figure 3.19 [42], guiding users on how to correctly align the dog's eye for optimal image capture [42].

The backend architecture of the application was designed to support the high computational demands of the segmentation model while ensuring portability. The system is containerized using Docker [98], which simplifies the deployment process and ensures that the application runs consistently across different platforms. This containerized approach also allows the application to scale efficiently, accommodating an increasing number of users without sacrificing

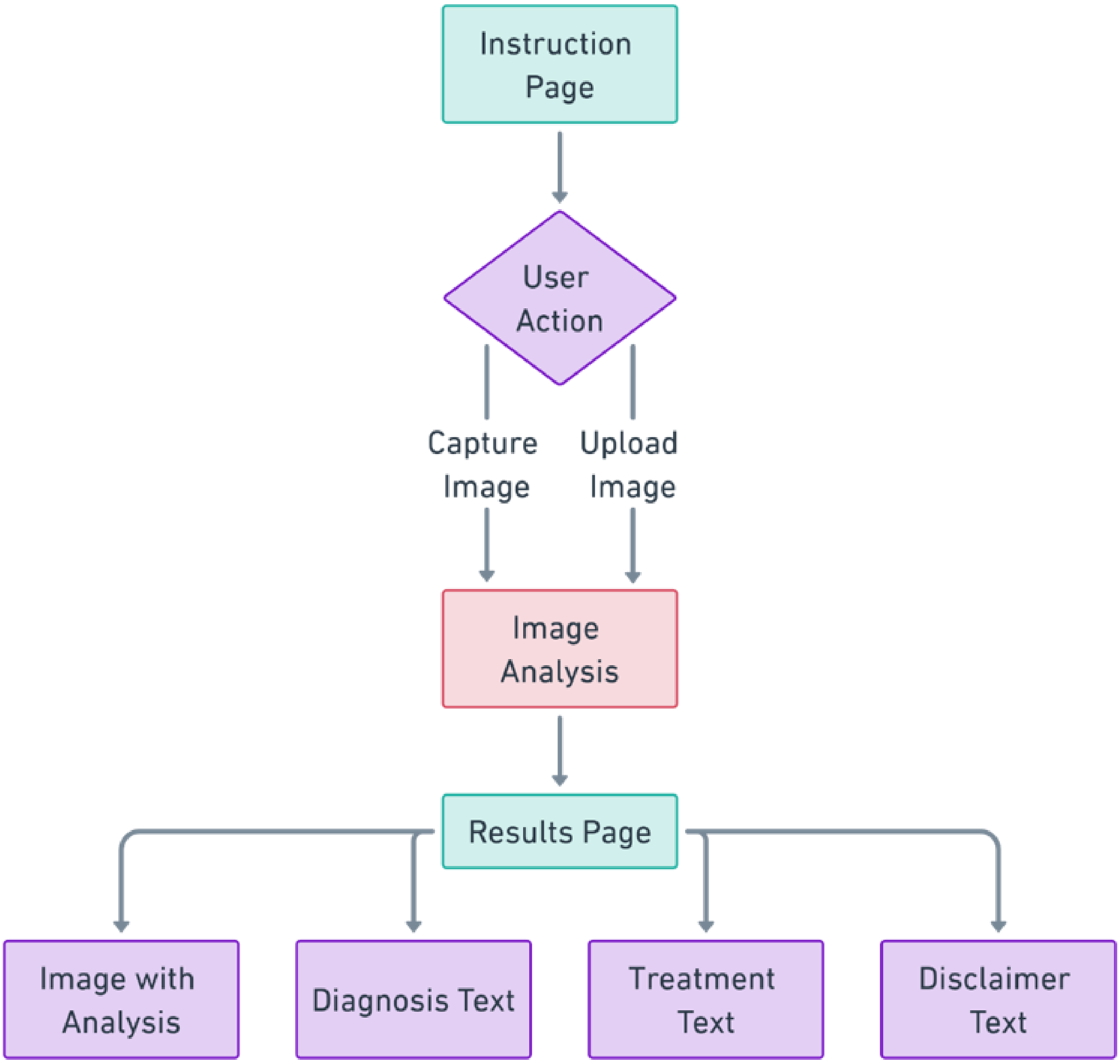


Figure 3.18: A web-based application flowchart for eye symptoms segmentation and medical condition prediction.

performance. The core image processing tasks are handled by a suite of Python [99] libraries, including TensorFlow [100] for model execution, OpenCV [101] for image manipulation, and Pillow [102] for image formatting. The backend server, built using Flask [103], ensures stable communication between the web interface and the desired segmentation model, processing images and delivering results in real-time [42]. The entire setup ensures that the application is portable, scalable, and easy to deploy on both local machines and cloud-based servers.

To ensure efficient handling of requests and model processing, the deployment was optimized for server environments equipped with advanced graphics processing units (GPUs), ensuring that the heavy computational tasks, such as segmenting and analyzing canine eye images, are handled with minimal latency. This server-based approach also allows for future scalability, where additional images can be collected and processed, contributing to the improvement of

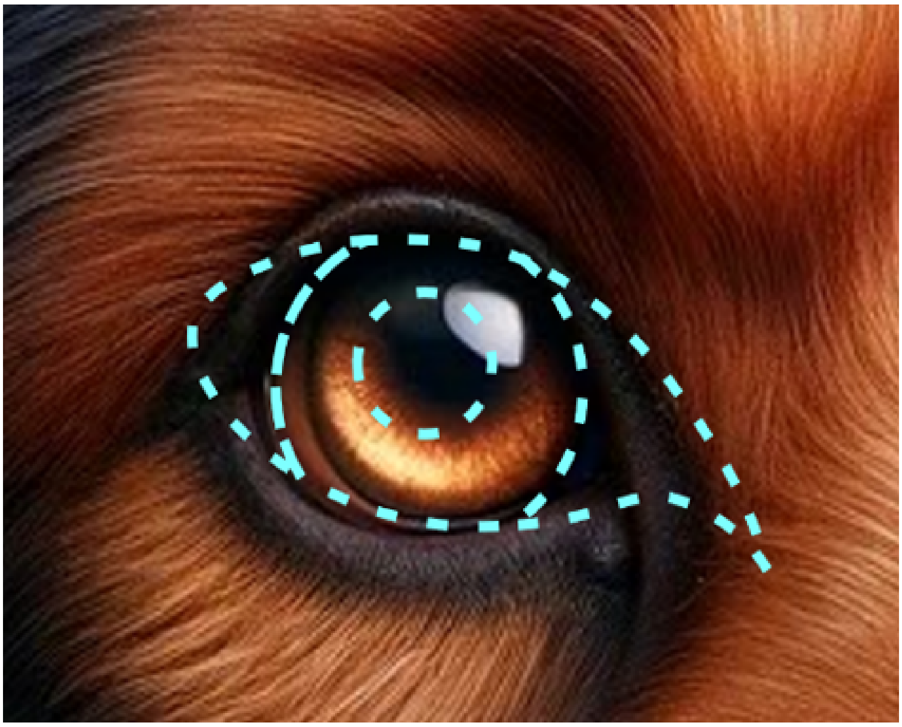


Figure 3.19: An overlay mask acting as capturing images guide for accurate segmentation [42].

the model through continuous retraining and dataset expansion. Additionally, the application is built with the capacity to expand into mobile platforms, further enhancing its accessibility.

The web application not only provides diagnostic insights but also aims to enrich the model by collecting user data with their consent. Each user interaction and image submission can potentially contribute to the growth of the dataset, allowing for further model refinement. This continuous loop of data collection, analysis, and improvement represents the application's potential for evolving alongside the veterinary community's needs. The user experience is further enhanced with feedback features, guiding users with next steps based on the results, and providing valuable insights into their pet's ocular health.

The combination of these technical and practical design choices has led to a comprehensive tool for diagnosing canine eye diseases, setting the foundation for further developments and improvements.

Chapter 4

Experiments and Results

This section presents experimental procedures and outcomes, focusing on the performance of various models in diagnosing canine ocular diseases. It is structured to provide insights into the evaluation process through several stages. First, we describe the test dataset used for model evaluation, including its structure and content. Next, we elaborate on the metrics applied to assess both segmentation and text generation tasks.

The core of the experimentation focuses on the training and evaluation of the U-Net model, both with and without backbone networks. A comparative study is also performed between the U-Net model trained on a real dataset versus one trained on a combined real and synthetic dataset to evaluate the impact of synthetic data augmentation. In a separate experiment, the benefits of binary vs. multiclass models were assessed along with their limitations.

Additionally, this section compares the performance of U-Net against GPT-4 and Grounding DINO in both segmentation and diagnostic interpretation tasks. These comparisons highlight the strengths and weaknesses of different approaches, allowing for a comprehensive assessment of model efficacy in real-world applications.

4.1 Test Dataset

The test dataset for evaluating the U-Net, GPT-4, and Grounding SAM models was assembled to ensure diversity and clinical relevance, drawing from a wide range of veterinary resources. The images in the dataset were sourced from veterinary medical encyclopedias, peer-reviewed medical articles, and specialized veterinary clinics, all of which provided documented diagnoses and necessary permissions [61]. Each image in the dataset was resized to a standardized 320×320 pixel format to facilitate consistent processing and analysis across models. This standardization is crucial to maintain the uniformity of input data and ensure the comparability of model outputs during testing.

A representative example from the dataset, as seen in Figure 4.1, includes images of an American Bulldog [104] and a crossbred dog [105], both diagnosed with Cherry Eye. These images depict a pink-colored ocular protrusion from the medial canthus—a classic sign of Cherry Eye—accompanied by excessive tearing. Although both dogs present similar symptoms, their differing breeds introduce subtle anatomical variations that challenge automatic segmentation models. The models must learn to handle such variations and correctly interpret symptoms in diverse breeds. This illustrates some of the inherent difficulties that computer vision must address when attempting to accurately segment and diagnose eye conditions in canines.

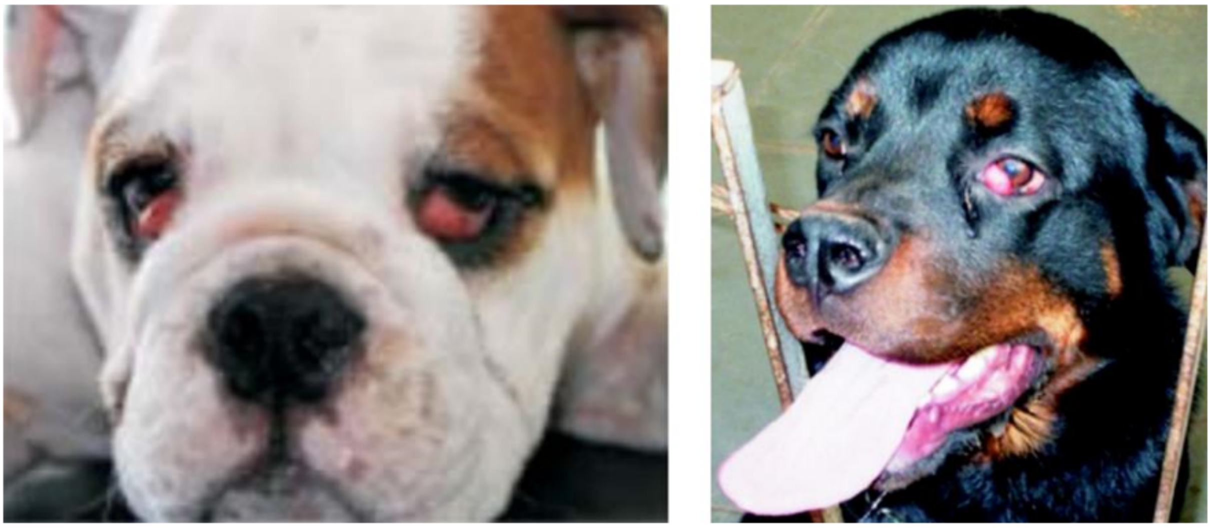


Figure 4.1: Images of an American bulldog (left) and a crossbred dog (right) show a pink-colored mass protruding from the medial canthus of both eyes, accompanied by signs of inflammation, including pain and excessive tearing, indicative of Cherry Eye [104, 105].

The test dataset was annotated in detail to reflect four key ocular symptoms that are commonly observed in veterinary ophthalmology:

- **S1** - Cloudiness or haziness of the cornea,
- **S2** - Redness of the sclera,
- **S3** - Excessive tearing,
- **S4** - A colored mass in the corner of the eye.

In this context, "symptoms" refer to visible signs, such as cloudiness or redness, that indicate underlying issues but are not diagnoses themselves. Conversely, "diseases" are specific diagnosed conditions that may manifest one or more of these symptoms. For consistency, both symptoms and diseases are collectively referred to as "ocular conditions" in this study. The goal of this study is to recognize diseases through symptom interpretation, as each disease is characterized by a unique combination of symptoms.

Each image was paired with a one-channel mask, stored in PNG format, annotating the four distinct classes of symptoms. These annotations are validating the models, as they provide ground truth data that allows the evaluation. The pixel intensity in the mask varies depending on the symptom being represented, allowing for precise identification of each condition. This approach mirrors the methodology used for the *DogEyeSeg4* custom dataset, ensuring consistency between training and test datasets.

The dataset includes images representing five significant ocular diseases in canines: Cherry Eye, Glaucoma, Uveitis, Corneal Ulceration, and Bacterial Keratitis. Each of these diseases is featured through five distinct cases, providing a total of 25 images for comprehensive model testing.

- **Cherry Eye**: This condition is characterized by the prolapse of the gland of the third eyelid, visible as a pink-colored mass in the corner of the eye. Cherry Eye can cause discomfort and, if left untreated, may lead to more severe complications such as dry eye or infection. The test images show various stages of Cherry Eye, providing the models with different levels of protrusion and tear production to segment and interpret.
- **Glaucoma**: Caused by elevated intraocular pressure, Glaucoma often results in pain, corneal cloudiness, and scleral redness. The dataset includes images showing a range of Glaucoma severity, from mild to advanced cases where the eyeball is visibly enlarged. These images challenge the models to recognize subtle differences in redness intensity and cloudiness levels.
- **Uveitis**: This inflammatory condition affects the uveal tract and often presents with sclera redness, pain, and sensitivity to light. In some cases, corneal cloudiness is also observed. The Uveitis cases in the dataset emphasize the need for precise segmentation of redness and cloudiness, ensuring that the models can distinguish between inflammatory conditions and other ocular issues.
- **Corneal Ulceration**: This condition is typically the result of trauma or infection and is marked by corneal erosion or scratches. The test images highlight the symptoms of excessive tearing and sclera redness, along with visible cloudiness in some cases. These images require the models to focus on small, localized areas of damage, a challenge in real-world clinical scenarios where the ulcers may be difficult to detect without specialized equipment.
- **Bacterial Keratitis**: Bacterial infection of the cornea leads to symptoms such as corneal cloudiness and tearing and can progress rapidly if untreated. The dataset provides examples of varying severity, allowing the models to learn how to segment the infected areas accurately and distinguish them from less severe conditions like Corneal Ulceration.

Each disease class in the test set is represented with diverse images, ensuring that the models are exposed to a broad spectrum of cases. This enables the models to generalize better and handle real-world variations in symptom presentation, breed anatomy, and disease progression. These images, along with their corresponding annotated masks, form the foundation for the final model evaluation. A sample of the test dataset can be observed in Figure 4.2.

This test dataset is not only a tool for evaluating the U-Net segmentation model but also serves as a benchmark for comparing the performance of the GPT-4 and Grounding SAM models. By

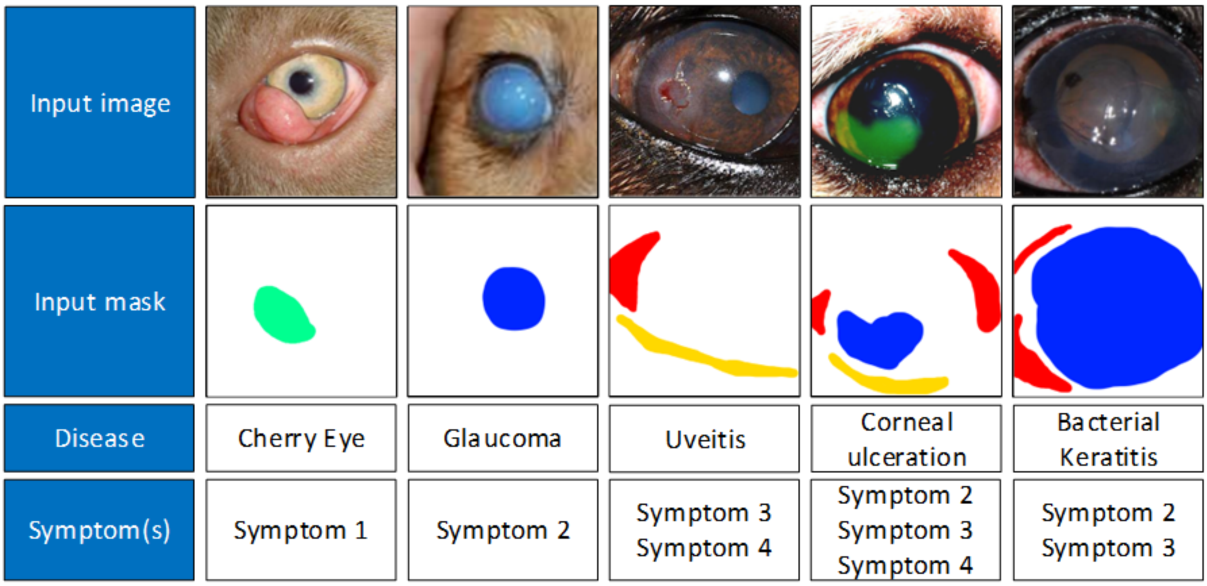


Figure 4.2: The test dataset contains 25 samples, evenly distributed across five different diseases. Symptom 1 represents a colored mass protruding in the corner of the eye, Symptom 2 refers to cloudiness or haziness in the cornea, Symptom 3 signifies redness of the sclera, and Symptom 4 denotes excessive tearing.

utilizing the same dataset across all models, the results can be compared directly. The models will be assessed based on their ability to accurately segment and interpret the symptoms associated with each disease, with particular focus on their precision in identifying small, nuanced differences between similar ocular diseases.

4.2 Metrics

The evaluation of the experiment is centered around two primary aspects: the **segmentation of symptoms** and the **interpretation of input images**. The results from both stages of the experiment will be quantitatively assessed using standard metrics for segmentation accuracy and text similarity, comparing the model outputs to ground truth data. The following subsections provide detailed explanations of the metrics and procedures employed for evaluating model performance.

4.2.1 Segmentation Metrics

To assess the effectiveness of the segmentation models, several well-established metrics are applied. These metrics offer a robust evaluation of how accurately the models segment the key symptoms in canine ocular images.

1. **Jaccard Index (Intersection-over-Union, IoU)**: The Jaccard Index measures the overlap between the predicted segmentation and the ground truth, reflecting how well the model can identify the same regions, which are the connected component of pixels, as the ground

truth segmentation. It is calculated as:

$$\mathrm{IoU} = \frac{TP}{TP+FP+FN}, \tag{4.1}$$

where TP represents true positives (correctly identified regions), FP represents false positives (incorrectly identified regions), and FN represents false negatives (missed regions), as described in Figure 4.3.

2. **Dice Index**: A measure similar to the Jaccard Index but with greater sensitivity to overlap, the Dice Index evaluates the degree of similarity between the predicted and actual segmentations. It is calculated using the following formula:

   $$\text{Dice Index} = \frac{2\times TP}{2\times TP+FP+FN}. \tag{4.2}$$

   The Dice Index is particularly useful in medical imaging as it penalizes small discrepancies between the predicted and actual segmentation more strongly than IoU, thus providing a more stringent assessment of segmentation performance.

3. **Pixel Accuracy**: This metric calculates the proportion of correctly classified pixels across the entire image, both for foreground and background classes. The formula is:

   $$\text{Pixel Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}, \tag{4.3}$$

   where TN refers to true negatives. While Pixel Accuracy offers a broad overview of the model's ability to classify pixels, it may be less sensitive to rare conditions when most of the image consists of healthy tissue.

4. **Sensitivity (Recall)**: Sensitivity, also known as the true positive rate, measures the proportion of actual positives (symptomatic regions) that the model correctly identifies. It is a critical metric in medical diagnostics because it indicates the model's ability to detect all possible symptomatic areas. The formula is:

   $$\text{Sensitivity} = \frac{TP}{TP+FN}. \tag{4.4}$$

   A high sensitivity score is crucial for ensuring that no symptomatic regions are missed.

5. **Specificity**: Specificity, or the true negative rate, evaluates how well the model identifies negative regions, which are areas of the image that are free from symptoms. The formula is:

   $$\text{Specificity} = \frac{TN}{TN+FP}. \tag{4.5}$$

   Specificity is important for minimizing false positives, ensuring that the model does not incorrectly classify healthy regions as symptomatic.

Both Sensitivity and Specificity work together to provide a comprehensive evaluation, ensuring

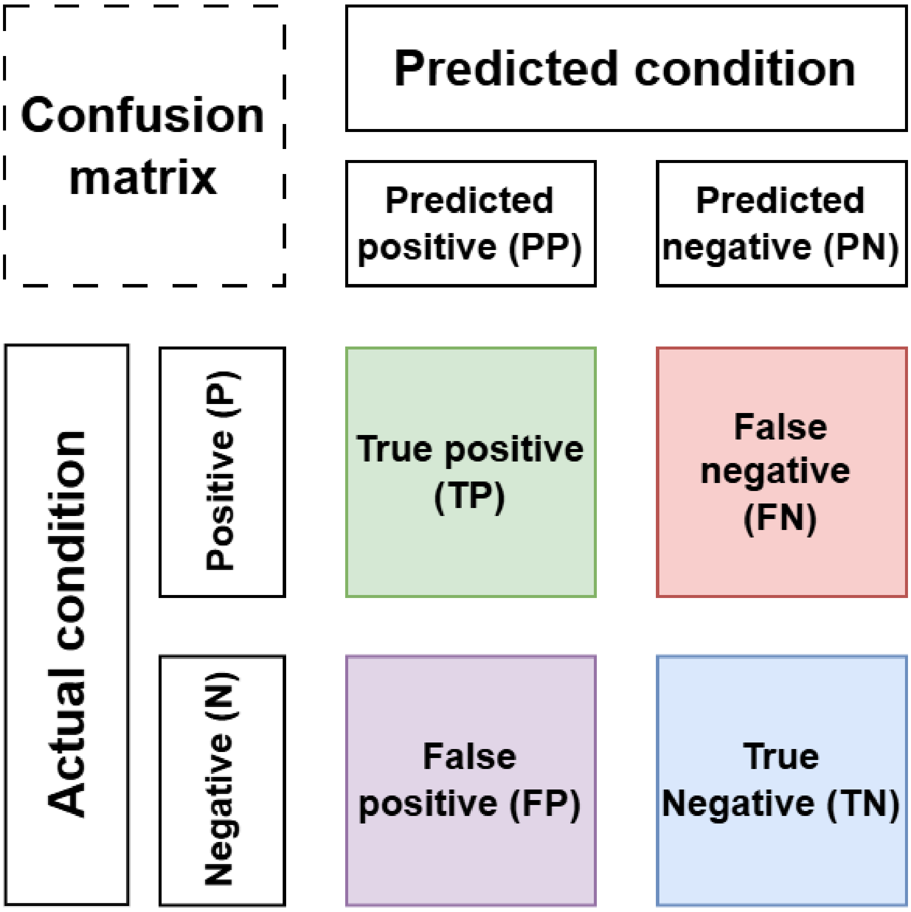


Figure 4.3: Representation of confusion matrix with horizontal argument as Ground Truth (GT) and vertical as Prediction showing correctly identified regions as TP, incorrectly identified regions as FP, missed regions as FN and correctly identified negative regions as TN.

that the model not only identifies symptomatic regions but also avoids misclassifying healthy areas.

For all these metrics, a score closer to 1 indicates better performance, signifying near-perfect alignment between the predicted segmentation and the ground truth mask. These metrics will be used to compare the performance of U-Net with other models, providing a detailed analysis of how well the models generalize across different cases.

4.2.2 Text Similarity Metrics

In the second part of the experiment, the models' ability to interpret images and generate accurate textual diagnoses is evaluated. This is particularly important for models like GPT-4 and GPT-4o, which generate text-based interpretations of the symptoms seen in the images. A suite of metrics commonly used in natural language processing (NLP) is employed to compare the generated text with the ocular diseases (Cherry Eye, Glaucoma, Uveitis, Corneal Ulceration, and Bacterial Keratitis) described in the Test dataset.

1. **BERTScore**:[54] BERTScore is a sophisticated metric that uses BERT embeddings to

evaluate the semantic similarity between the generated text and the ground truth. Embeddings are vector representations of words or tokens, capturing their meanings based on context. In BERTScore, embeddings represent the meaning of each token within its specific sentence, allowing the model to understand nuances in meaning. Each embedding, denoted as $\mathrm{emb}(x)$ for a token x, is obtained by passing x through the BERT model's layers. This process produces a contextualized vector for each token, where each dimension of the vector captures aspects of its meaning in relation to surrounding words.
The similarity is measured by calculating the cosine similarity between the tokens of the predicted and reference sentences using Precision, Recall and F1 score:
The formulas for Precision (P_B) and Recall (R_B) are:

$$P_B = \frac{1}{|C|}\sum_{t\in C}\max_{t'\in R}\cos(\mathrm{emb}(t),\mathrm{emb}(t')), \tag{4.6}$$

$$R_B = \frac{1}{|R|}\sum_{t'\in R}\max_{t\in C}\cos(\mathrm{emb}(t'),\mathrm{emb}(t)), \tag{4.7}$$

where C represents the set of tokens in the candidate sentence and R the set of tokens in the reference sentence, and $\cos(\mathrm{emb}(t),\mathrm{emb}(t'))$ denotes the cosine similarity between the embeddings of tokens t and t'. The F1 Score (F_B) is computed as:

$$F_B = \frac{2P_BR_B}{P_B+R_B}. \tag{4.8}$$

2. **CLIPScore** [55]: Leveraging the multimodal CLIP model [106], CLIPScore measures the similarity between the vector representations of the generated text and the reference text. It is calculated using the cosine similarity between the two vectors:

$$\mathrm{CLIPScore} = \frac{V_{\mathrm{candidate}}\cdot V_{\mathrm{reference}}}{||V_{\mathrm{candidate}}||\,||V_{\mathrm{reference}}||}, \tag{4.9}$$

   where $V_{\mathrm{candidate}}$ and $V_{\mathrm{reference}}$ represent the vector embeddings of the candidate and reference texts.
3. **BLEU** [56]: This metric evaluates the overlap of n-grams (sequences of words) between the candidate and reference texts, penalizing shorter sentences to ensure that full descriptions are considered. The formula is:

$$\mathrm{BLEU} = BP\cdot\exp\left(\sum_{n=1}^{N} w_n \log p_n\right), \tag{4.10}$$

   where BP is the brevity penalty, w_n is the weight for n-gram precision at n-th order, and p_n is the modified n-gram precision.
4. **METEOR** [57]: This metric computes the harmonic mean of unigram precision and recall, leveraging linguistic resources such as stemmers and synonyms. The formula is:

$$\text{METEOR} = F_{\text{METEOR}} = \frac{10PR}{9P+R}, \tag{4.11}$$

where P represents unigram precision, and R represents unigram recall. In this context:

- **Unigram Precision (P)** is the ratio of unigrams (single words) in the candidate text that match unigrams in the reference text to the total number of unigrams in the candidate. This precision measure reflects how many of the candidate's words are relevant compared to the reference.
- **Unigram Recall (R)** is the ratio of matched unigrams in the candidate text to the total number of unigrams in the reference text, capturing how many of the reference's words are covered by the candidate.

5. **ROUGE**[58]: ROUGE measures the overlap between the longest common subsequences in the candidate and reference texts, without requiring consecutive matching words. ROUGE-L is a common variant used:

$$\text{ROUGE}_L = \frac{(1+\beta^2)RP}{R+\beta^2 P}, \tag{4.12}$$

   where:

   - R is **recall**, calculated as the length of the longest common subsequence divided by the total length of the reference text. It reflects how much of the reference content is covered by the candidate.
   - P is **precision**, calculated as the length of the longest common subsequence divided by the total length of the candidate text, indicating how accurately the candidate matches the reference.
   - β is a **weighting factor** that controls the balance between precision and recall. Setting $\beta = 1$ treats precision and recall equally, while higher values of β place more emphasis on recall, which may be desirable when capturing the completeness of the candidate text relative to the reference.

6. **SPICE**[59]: SPICE compares the scene graphs of the candidate and reference sentences, focusing on the semantic content. Scene graphs represent the semantic structure of a sentence as a graph of objects (nodes), attributes, and relationships (edges). It evaluates the propositional meaning of the sentences rather than just surface-level n-gram matches:

$$\text{SPICE} = F_{\text{SPICE}} = \frac{2PR}{P+R}. \tag{4.13}$$

where:

- P (**Precision**) is calculated as the ratio of correctly matched semantic propositions in the candidate scene graph to the total propositions in the candidate graph, reflecting how accurately the candidate captures the reference content.
- R (**Recall**) is calculated as the ratio of correctly matched semantic propositions in the

candidate scene graph to the total propositions in the reference graph, showing the completeness of the candidate in covering the reference's key meanings.

4.3 Performance Evaluation

In this section, we explore a series of experiments designed to determine the most effective approach to the workflow of canine ocular disease diagnosis, as depicted in Figure 4.4. Through these experiments, we aim to identify the optimal solution that integrates both advanced image segmentation and precise diagnostic interpretation.

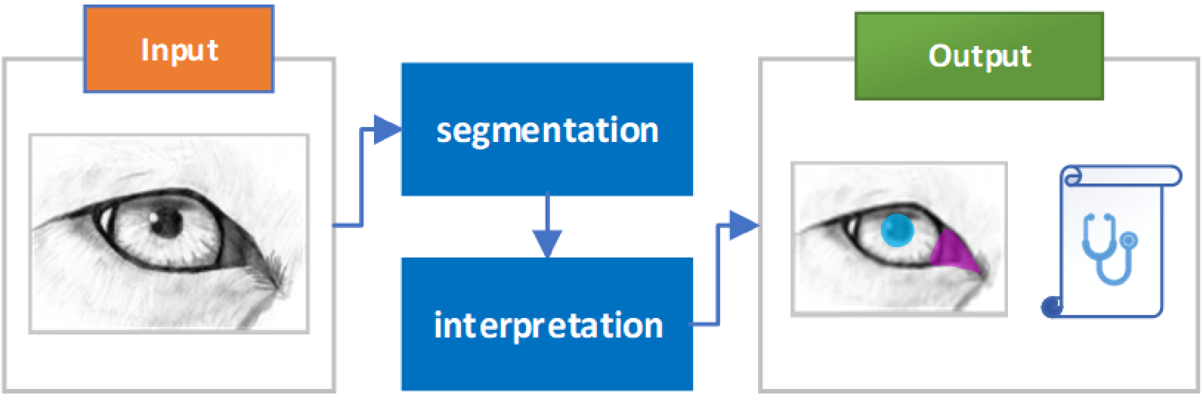


Figure 4.4: The workflow diagram for analyzing canine ophthalmic conditions demonstrates the process of evaluating an image of a dog's eye. This involves utilizing model for segmentation, followed by the use of LLM (Large Language Model) for interpreting the medical symptoms. The outcome is a segmented image alongside a comprehensive diagnostic report.

4.3.1 Experiment 1: Ocular symptom segmentation with U-Net

The first experiment focuses on the training and initial evaluation of models based on the U-Net architecture for segmenting ocular symptoms from canine eye images. This foundational stage assesses U-Net's ability to accurately delineate areas of interest within the images, which are crucial for subsequent diagnostic processes.

To improve the segmentation performance, several backbone networks were tested in use with the U-Net model was tested with several backbone networks. Four different backbones were chosen based on their popularity and proven efficiency in medical imaging:

- **ResNet34**,
- **Inception V3**,
- **VGG16**, and
- **EfficientNet B3**.

The integration of these backbones into the U-Net model helps capture both low-level and high-level features from the images, allowing for more accurate segmentation of ocular symptoms [26]. These specific backbones were selected due to their availability with pre-trained weights, which allowed for faster and more accurate convergence during training. Each backbone was evaluated through a series of preliminary experiments, taking into consideration the balance between processing speed and segmentation accuracy.

The training of the U-Net model, including its variations with different backbones, was conducted with a learning rate of 0.0001 and a fixed batch size of 16 images. The Adam optimizer was employed, as it efficiently handles sparse gradients, which is essential for accurate segmentation in medical imaging. Three loss functions were tested across 100 epochs:

- **Categorical Cross-Entropy (CCE)**,
- **Dice Loss (DL)**, and
- **Focal Loss (FL)**.

These loss functions were applied individually and also summed together to assess their influence on segmentation accuracy. The models' performance was monitored through metrics like loss values and Intersection-over-Union (IoU) during training, and the best-performing models were preserved for further evaluation.

For this initial experiment, the *DogEyeSeg4* dataset was divided into a 90:10 split, where 90% of the images were used for training, and 10% for testing, ensuring that the models had enough diverse data for robust learning. Images were preprocessed to standardize their resolution and quality, ensuring consistency across all training and evaluation procedures. Additionally, data augmentation techniques, such as flipping, rotation, and brightness adjustment, were employed to increase the diversity of the training data.

The experiment was conducted using a NVIDIA GeForce RTX 2060 GPU with 12GB of memory for the training phase, while the testing phase was run on an Intel Xeon E5-1603 v4 CPU. The use of this hardware enabled efficient training of the models, ensuring timely completion of the experiments. The models were sequentially trained on the same hardware using identical datasets to ensure the fairness of the evaluation process. The performance metrics observed during training were the loss and IoU values, with progress detailed in Figure 4.5 [62]. This figure illustrates the average training loss and IoU across epochs, showcasing the development and refinement of each model's segmentation ability over time.

The evaluation of the models focused on their segmentation capabilities, utilizing a balanced metric approach that incorporated Intersection over Union (IoU). Models that exhibited the highest scores during the 100 training epochs were selected for further evaluation.

As illustrated in Table 4.1, the experiment categorized the models into the base U-Net model and four groups differentiated by the backbone utilized. Each group was further divided based on the loss function employed during training, including Categorical Cross Entropy (CCE), Dice Loss (DL), Focal Loss (FL), and a combined Dice and Focal Loss (DL+FL).

The results clearly demonstrate that incorporating U-Net with various backbone models significantly improved segmentation and classification performance compared to the original U-Net. This improvement is seen across all symptom classes, with the exception of **S4** (A colored mass

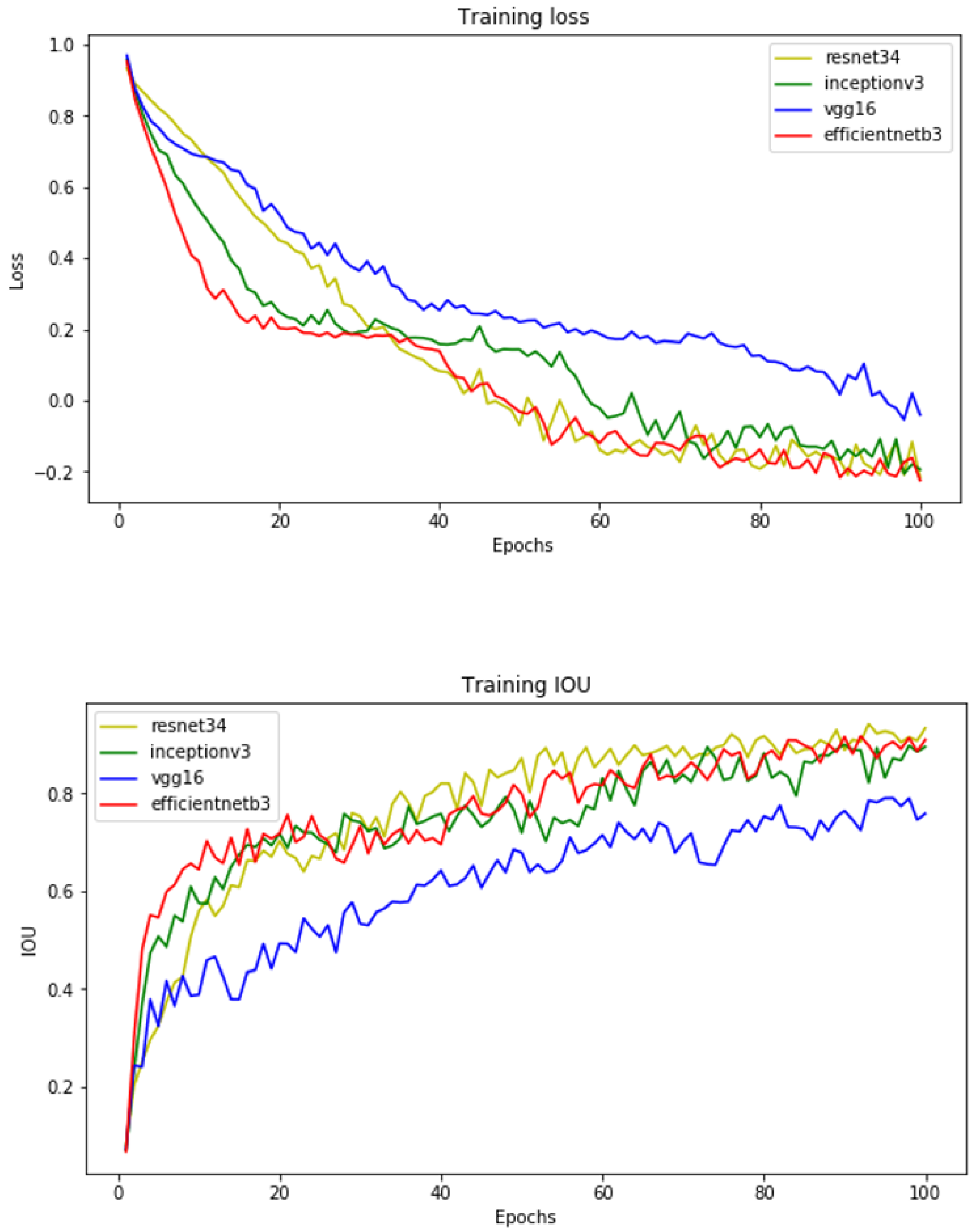


Figure 4.5: Average training loss (upper) and IOU (lower) values for 100 epochs showing expecting curve of well performing model training [62].

in the corner of the eye). The challenges in this class likely stem from an insufficient amount of data, which hindered EfficientNet's parameter tuning, and the relatively simplistic VGG backbone compared to other backbones, which was unable to extract the necessary discriminative features for this condition.

Table 4.1: Segmentation models evaluation results in terms of Jaccard coefficient (IoU) percentages. Training time (t^*) needed for training on full dataset using NVIDIA GeForce RTX 2060 GPU with 12GB RAM and average test time per image (t^{**}) on Intel(R) Xeon(R) CPU E5-1603 v4 @ 2.80GHz.

Architecture	Loss Func.	S1	S2	S3	S4	Mean IoU	t^*/t^{**}
U-Net	CCE	38.7	37.5	0	47.9	31	4.5h / 0.5s
	DL	36.2	35.1	0.1	58.9	32.6	
	FL	41.1	37.7	0	56.8	33.9	
	DL+FL	38.5	44	0.3	55.8	34.7	
U-Net+ResNet34	CCE	62.3	63.9	20	37.2	45.9	6h / 1s
	DL	70.8	62.1	**64.6**	32.8	57.6	
	FL	68.5	78.2	59.9	53.2	65	
	DL+FL	73.9	**80.6**	38	**73.9**	**66.6**	
U-Net+InceptionV3	CCE	64.1	62.9	0.9	36.2	41	8h / 2s
	DL	71.2	68	31.6	61.1	58	
	FL	62.1	65.6	18.2	21.4	41.8	
	DL+FL	**78.3**	78.3	37.9	54.2	62.2	
U-Net+VGG16	CCE	63.1	66.6	1.3	33.8	41.2	6h / 0.5s
	DL	67.1	60.4	31.4	27.4	46.6	
	FL	66.1	63.8	29.3	41	50.1	
	DL+FL	75.1	75.7	27.2	54.1	58	
U-Net+EfficientNetB3	CCE	66.4	64.5	2.9	37	42.7	7h / 3s
	DL	67.6	63.7	31.9	31.9	48.8	
	FL	67.3	70.1	28.7	38	51	
	DL+FL	69.7	79.2	36.1	67.5	63.1	

S1- Cloudiness or haziness of the cornea, **S2** - Redness of the sclera, **S3** - Excessive tearing and **S4** - A colored mass in the corner of the eye.

The U-Net+ResNet34 model trained with the DL+FL loss function showed the highest overall performance, achieving the top IoU scores for two out of the four symptom classes. In particular, this model displayed significant accuracy in segmenting **S2** (Redness of the sclera) and **S3** (Excessive tearing), indicating its effectiveness in detecting subtle symptoms involving redness and tear production.

For **S3** (Excessive tearing), using Dice Loss alone within the U-Net+ResNet34 model resulted in a marked improvement in segmentation accuracy, outperforming other models for this class. This highlights the advantage of tailored loss functions for improving performance in detecting less prominent or infrequent symptoms.

The U-Net+InceptionV3 model achieved the highest Jaccard score for **S1** (Cloudiness or haziness of the cornea), with a Jaccard index of 78.3%. This indicates that Inception V3's architectural design is particularly adept at handling more diffuse symptoms, where cloudiness and haziness affect larger areas of the image and require precise boundary detection.

When the background class is excluded, the difference between the best and worst-performing models reaches 27%, both of which utilize the same backbone architecture. This underscores the significant impact of loss function selection on segmentation performance. In particular, the combined DL+FL loss function led to the greatest improvement, especially in cases where symptom boundaries are less distinct.

Excluding the least detectable symptom class, **S3** (Excessive tearing), from the evaluation resulted in an average improvement of over 13% across all models, further emphasizing the difficulty in segmenting this class. Nonetheless, U-Net+VGG16 models demonstrated the quickest segmentation times, making them highly efficient for real-time applications, despite not being the top performers in terms of accuracy.

These findings support the proposition that incorporating advanced backbones into the U-Net architecture, coupled with appropriately chosen loss functions, can significantly enhance segmentation accuracy, particularly for more challenging symptoms in canine ocular disease diagnostics.

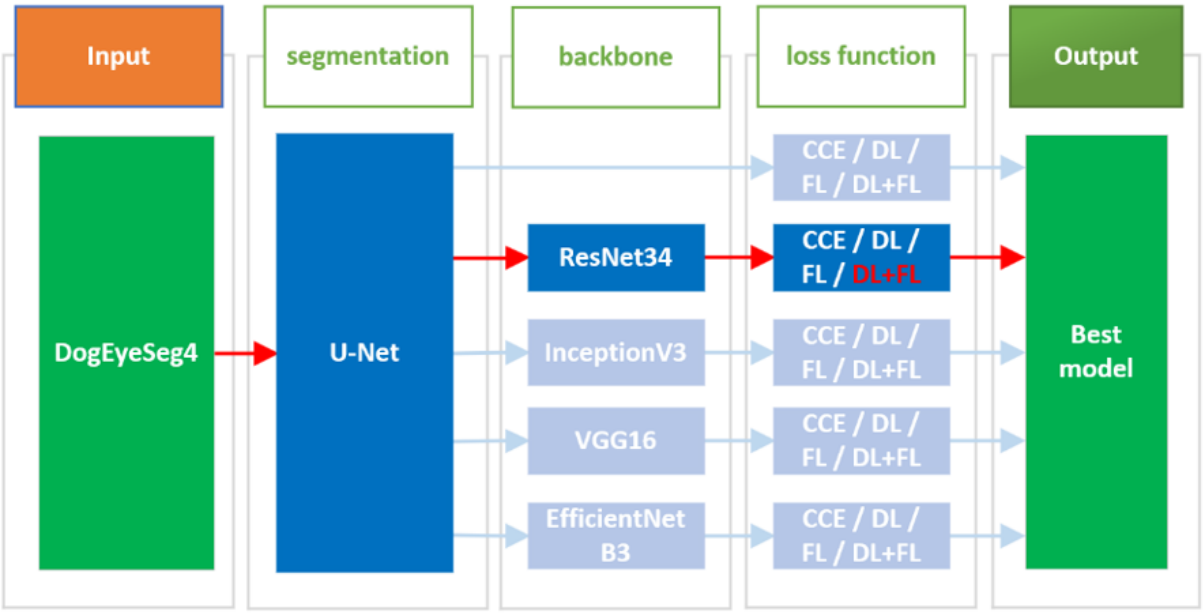


Figure 4.6: Flowchart describing the results of the first experiment where U-Net with ResNet34 backbone and combination of Dice and Focal loss provide best performance model when accuracy and time is taken into consideration for further research.

The in-depth evaluation of our models employed 5-fold cross-validation, focusing on those using the combined Dice Loss (DL) and Focal Loss (FL) due to their superior initial performance, as seen in Figure 4.6. Cross-validation helps demonstrate the robustness of the models and the reliability of the dataset across various training and validation scenarios. This method ensures that every data point is used both for training and validation, thus providing a comprehensive assessment of model stability and performance consistency.

In this detailed phase of evaluation, the *DogEyeSeg4* dataset was divided into five folds, following an 80:20 ratio for training to validation, which allowed for the rigorous testing of model reliability across different subsets of data. As depicted in Figure 4.7 [62], each fold was sequentially used as the validation set, with the remaining parts used for training. This rotational approach not only maximizes the use of available data but also tests the model's ability to adapt to new, unseen data, thereby reinforcing the robustness of both the model architecture and the training process.

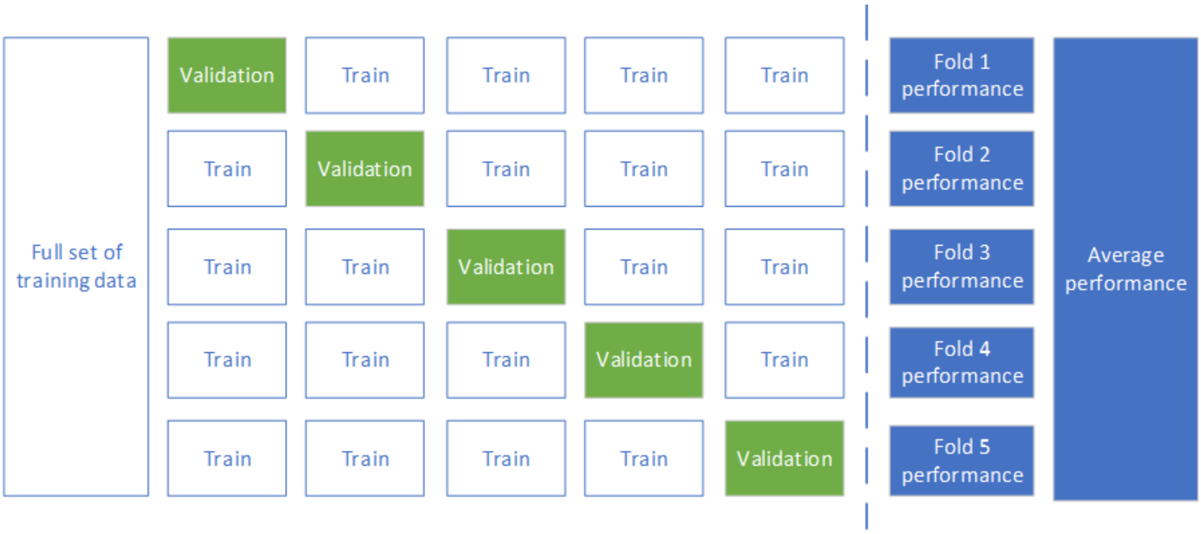


Figure 4.7: Visual representation of 5-fold cross-validation where the dataset folds in ratio 80:20 providing 5 different models which are validated on unseen part of the dataset [62].

The primary goal of employing cross-validation in this experiment was to validate the quality of the dataset and the consistency of the model performance. By rotating the validation set across different subsets of the data, the process mitigates any potential biases or anomalies that could affect the model's efficacy, providing a more accurate and generalizable assessment of its capabilities. This is particularly important in medical imaging applications , where high precision and reliability are crucial.

The results of the cross-validation are summarized in Table 4.2, which uses the IoU metric to quantify the performance across different folds. The inclusion of the background as a class during this phase of the experiment substantially improved the segmentation accuracy, as evidenced by the higher average IoU scores.

U-Net Base Model and U-Net with Inception V3, VGG16, and EfficientNet B3 Backbones pre-

Table 4.2: 5-fold cross-validation results using mean IoU in percentage with background class included.

Architecture	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
U-Net	53.0	54.2	50.1	52.0	52.2
U-Net+ResNet34	**73.2**	**72.6**	**72.3**	**71.2**	**76.0**
U-Net+InceptionV3	69.5	68.6	69.7	67.5	69.7
U-Net+VGG16	65.3	69.5	65.2	64.3	65.5
U-Net+EfficientNetB3	67.2	69.3	73.4	70.6	68.0

sented strengths in various aspects of ocular disease segmentation, showing promise for different symptom characteristics and image features. However, the U-Net with ResNet34 backbone consistently delivered the best performance across all folds, including those with higher variability in symptom presentation and image conditions. This performance stability across diverse folds suggests that the ResNet34 backbone generalizes effectively, making it particularly suited for complex segmentation tasks where image features can vary significantly.

In the absence of reference results for the models trained on our image database for segmentation and classification of eye diseases, we turned to statistical tests—ANOVA [107] and the Tukey HSD [108]—to validate the effectiveness of our models. These tests used to assess whether the differences in performance across various model configurations are statistically significant and not due to random variation.

First, we assessed the normality of the model performance metrics (IoU scores across different architectures) distribution through a QQ-Plot. This plot is used for validating the assumption of normality required by ANOVA and Tukey HSD tests. The QQ-Plot in our study (Figure 4.8 [62]) indicated that the standardized residuals of our models closely align with a theoretical normal distribution, suggesting that the assumptions of normality are satisfactorily met. This alignment supports the subsequent application of ANOVA and Tukey HSD tests to our dataset.

To assess the statistical significance of the differences observed between the various U-Net configurations, a one-way Analysis of Variance (ANOVA) was conducted [109, 110]. This statistical method helps determine whether there are any statistically significant differences between the means of three or more independent groups. In our study, these groups were defined by different network architectures.

Step-by-step ANOVA Procedure:

1. **Calculation of Group Means**: The first step involved calculating the mean Jaccard index coefficient for each group (U-Net, U-Net+ResNet34, U-Net+InceptionV3, U-Net+VGG16 , and U-Net+EfficientNetB3) based on their performance across all folds. This provides a baseline from which to measure deviations among the groups.

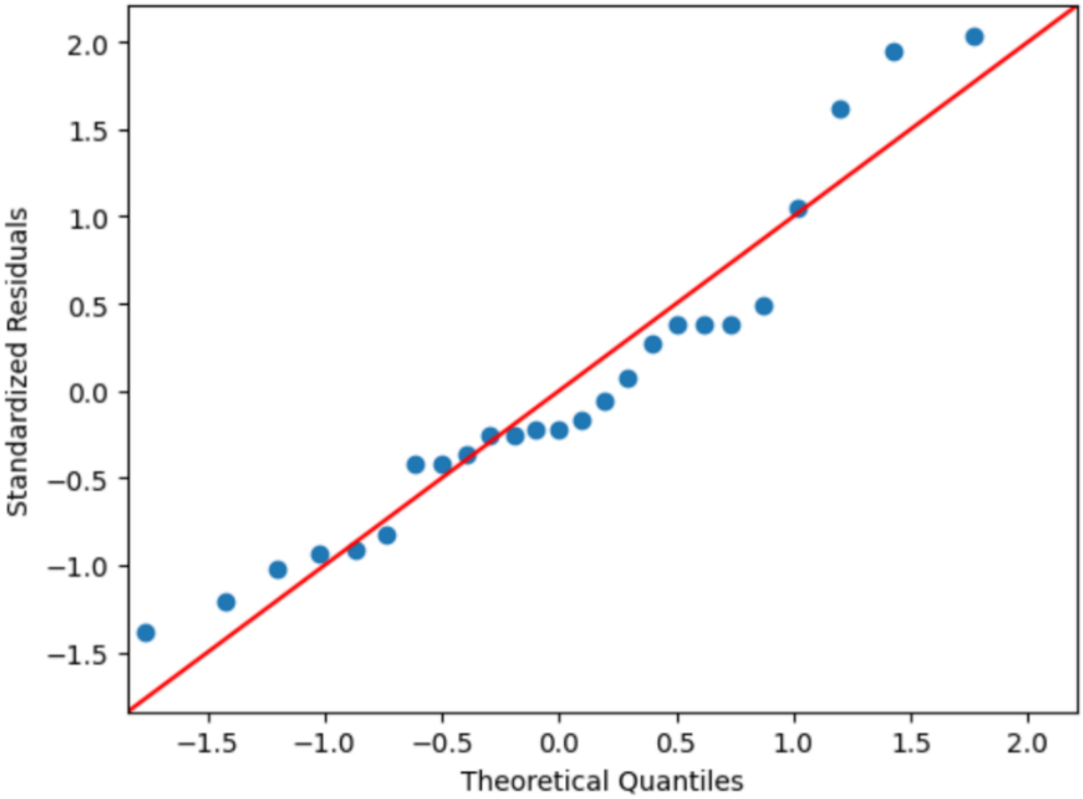


Figure 4.8: Display of a QQ-plot which evaluates the standardized residuals of a model against a theoretical normal distribution. The data points closely follow the expected normal line, exhibiting only slight deviations at the ends, suggesting that the residuals generally adhere to a normal distribution [62].

2. **Between-Group Variance**: This metric was computed to quantify how much the group means deviate from the overall mean of all combined samples. This variance is indicative of the effect that changing the network architecture has on segmentation performance.
3. **Within-Group Variance**: This step measured the variability of segmentation performance within each group around their respective mean. This variance is critical as it shows how consistent a model's performance is across different folds, reflecting the inherent variability in model predictions.
4. **Calculation of the F-value**: The F-value was calculated by dividing the between-group variance by the within-group variance. An F-value higher than the critical value from F-distribution tables, considering the degrees of freedom associated with the between- and within-group variances, indicates that the group means are not all equal.
5. **Deriving the P-value**: The p-value is derived from the calculated F-value and the degrees of freedom for the test. It represents the probability of observing the calculated level of variation among group means under the null hypothesis, which posits that all group means are equal.

In our analysis, a p-value of 5.294×10^{-13} was derived, which is significantly lower than the typical threshold of 0.05. This extraordinarily low p-value suggests that there is a statistically significant difference between the group means being compared. This result leads us to reject the null hypothesis and confirms that the modifications in network architecture significantly affect the models' segmentation performance.

To further investigate these differences, the Tukey Honest Significant Difference (HSD) test was employed [108]. This test facilitates multiple pairwise comparisons between the model architectures while controlling the family-wise error rate. It adjusts p-values to account for multiple testing, thus reducing the likelihood of type I errors (false positives).

The results of the Tukey HSD test are detailed in Table 4.3. These results show significant differences in performance between the base U-Net model and its enhanced versions incorporating the ResNet34, Inception V3, VGG16, and EfficientNet B3 backbones, with all p-values being notably low (≤ 0.001 for most comparisons). The enhancements clearly outperformed the base model, affirming the effectiveness of integrating advanced architectures into the U-Net framework.

Table 4.3: Comparison of models based on the Jaccard Index using the Tukey HSD test. P-values are provided for each pairwise comparison.

Architecture 1	Architecture 2	p-value
U-Net	U-Net+ResNet34	0.001
U-Net	U-Net+InceptionV3	0.001
U-Net	U-Net+VGG16	0.001
U-Net	U-Net+EfficientNetB3	0.001
U-Net+ResNet34	U-Net+InceptionV3	0.013
U-Net+ResNet34	U-Net+VGG16	0.001
U-Net+ResNet34	U-Net+EfficientNetB3	0.046
U-Net+InceptionV3	U-Net+VGG16	0.09
U-Net+InceptionV3	U-Net+EfficientNetB3	0.9
U-Net+VGG16	U-Net+EfficientNetB3	0.028

The pairwise comparisons within the enhanced models revealed varying levels of impact on performance. Notably:

- **Base U-Net vs. U-Net + different backbone**: A p-value well below 0.001 indicates a highly significant difference between the base U-Net model and the backbone-enhanced versions.
- **U-Net+ResNet34 vs. U-Net+InceptionV3**: A p-value of 0.013 indicates a statistically significant difference in performance between these two models, showing that they do not perform equivalently.
- **U-Net+ResNet34 vs. U-Net+EfficientNetB3**: With a p-value of 0.046, suggesting a

modest but still significant difference.

- **U-Net+ResNet34 vs. U-Net+VGG16** : The p-value of 0.001 shows a statistically significant difference between these two models, confirming that they do not yield equivalent results.
- **U-Net+InceptionV3 vs. U-Net+VGG16** : A p-value of 0.09 suggests that the difference in performance between these models is not statistically significant, indicating comparable results.
- **U-Net+EfficientNetB3 vs. U-Net+VGG16** : The p-value of 0.028 shows a statistically significant difference between these two models, confirming that they do not yield equivalent results.
- **U-Net+InceptionV3 vs. U-Net+EfficientNetB3**: A p-value of 0.9 suggests no statistically significant difference between these models, implying similar performance levels.

These findings indicate that certain architectural enhancements significantly improve model performance, with U-Net+ResNet34 consistently showing superior results and also significantly differing from other models. The cross-validation results for the U-Net+ResNet34 models, as illustrated in Table 4.4, show a detailed performance breakdown across IoU, DSC, Sensitivity, and Pixel Accuracy metrics.

Table 4.4: 5-fold cross-validation results of U-Net+ResNet34 models based on IoU, Dice Index (DSC), Sensitivity, and Pixel Accuracy. The best results are marked in bold, and average in *italic*.

Model	IoU (mean ± std)	DSC (mean ± std)	Sensitivity (mean ± std)	Pixel Accuracy
1	0.732 ± 0.142	0.838 ± 0.092	0.868 ± 0.101	0.966
2	0.726 ± 0.172	0.830 ± 0.117	0.814 ± 0.172	0.981
3	0.723 ± 0.159	0.829 ± 0.110	0.876 ± 0.103	0.959
4	0.712 ± 0.188	0.817 ± 0.135	0.849 ± 0.212	**0.982**
5	$\mathbf{0.760 \pm 0.108}$	$\mathbf{0.860 \pm 0.066}$	$\mathbf{0.913 \pm 0.086}$	0.963
Average	$\mathit{0.740 \pm 0.158}$	$\mathit{0.834 \pm 0.097}$	$\mathit{0.863 \pm 0.117}$	*0.972*

The IoU values for these models range from 71.2% to 76.0%, with a mean of 74.0% and a standard deviation of 0.158. The Dice Similarity Coefficient (DSC) ranges from 81.7% to 86.0%, with an average of 83.4% and a standard deviation of 0.097. These metrics reflect a relatively tight performance band, indicating consistent model effectiveness across different folds with minimal fluctuation, demonstrating a maximum performance deviation of approximately 4%. Sensitivity values, which measure the true positive rate, range from 81.4% to 91.3%, with an average of 86.3%, suggesting that the model reliably identifies relevant features across different test scenarios. Pixel accuracy across the models remains high, averaging 97.2%, highlighting the model's overall accuracy in classifying each pixel correctly, whether as a part of the lesion or background.

The confusion matrix for U-Net+ResNet34 Cross-Validation Model 2, shown in Table 4.5, provides a granular view of the model's classification accuracy across different classes against the ground truth. It helps in understanding how well the model discriminates between different ocular conditions.

Table 4.5: Confusion matrix for U-Net+ResNet34 Cross-Validation Model 2 (pixels presented in %).

		Prediction				
		BCK	**S1**	**S2**	**S3**	**S4**
Ground Truth	**BCK**	**99**	1	0	0	0
	S1	15	**85**	0	0	0
	S2	16	0	**84**	0	0
	S3	3	0	0	**97**	0
	S4	30	0	0	0	**70**

- **Background (BCK)**: Shows an exceptionally high identification rate of 99% pixels, with a very low false positive rate, misclassifying only 1% as Cloudiness or haziness of the cornea (**S1**).
- **S1**: Correctly identified 85% pixels, demonstrating strong model performance in recognizing this condition, with 15% misclassified as background but with still some space for improvement.
- **S2**: Exhibits an accuracy of 84% pixels, indicating effective recognition, though 16% of cases are mistaken as background, suggesting room for sensitivity improvement.
- **S3**: Achieves the highest-class accuracy among disease conditions at 97% pixels, with minimal misclassification.
- **S4**: Shows lower correct classification at 70% pixels, with a significant 30% pixels of instances misclassified as background, indicating specific challenges in distinguishing this condition due to its visual similarity to background features.

It is worth mentioning that in this particular model no misclassification was made between symptoms but only between symptoms and background. These observations collectively highlight the U-Net+ResNet34 model's strengths in accurately segmenting and classifying ocular conditions with high reliability and minimal variation across different validation folds. The detailed metrics and confusion matrix analysis provide a comprehensive view of where the model excels and where improvements could be targeted.

In Figures 4.9 and 4.10, the segmentation results are displayed alongside the corresponding ground truth and predictions made by the U-Net+ResNet34 model utilizing the combined Dice and Focal Loss approach. Different colors are employed to distinguish various classes: dark blue indicates the background, light blue represents cloudiness or haziness of the cornea (**S1**), green denotes redness of the sclera (**S2**), orange signifies excessive tearing (**S3**), and red indicates a colored mass in the corner of the eye (**S4**).

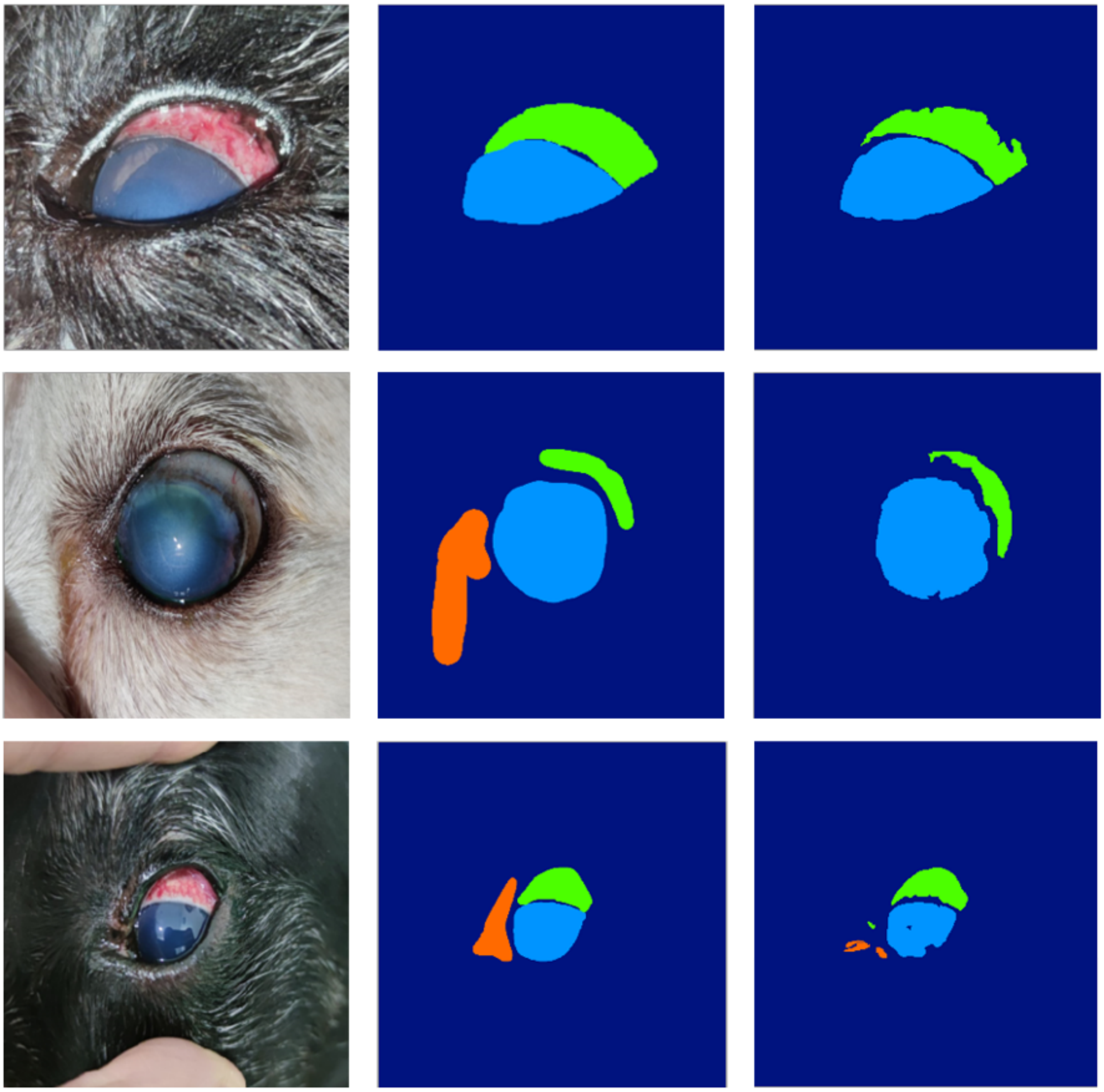


Figure 4.9: Examples of the segmentation analysis of close-up eye images with diagnostic labels. The original images are shown in the left column, the middle column represents the ground truth (GT), and the predictions are displayed in the right column. In the segmentation results, the top row correctly identifies two classes according to the ground truth, the middle row misses the detection of the excessive tearing class, and the bottom row incorrectly identifies an additional instance of redness of the sclera.

Now that the in-depth evaluation of the preferred U-Net based model is performed, we can conclude that the U-Net+ResNet34 model trained with combined Dice and Focal loss is the optimal segmentation solution in correlation to the *DogEyeSeg4* dataset for further research.

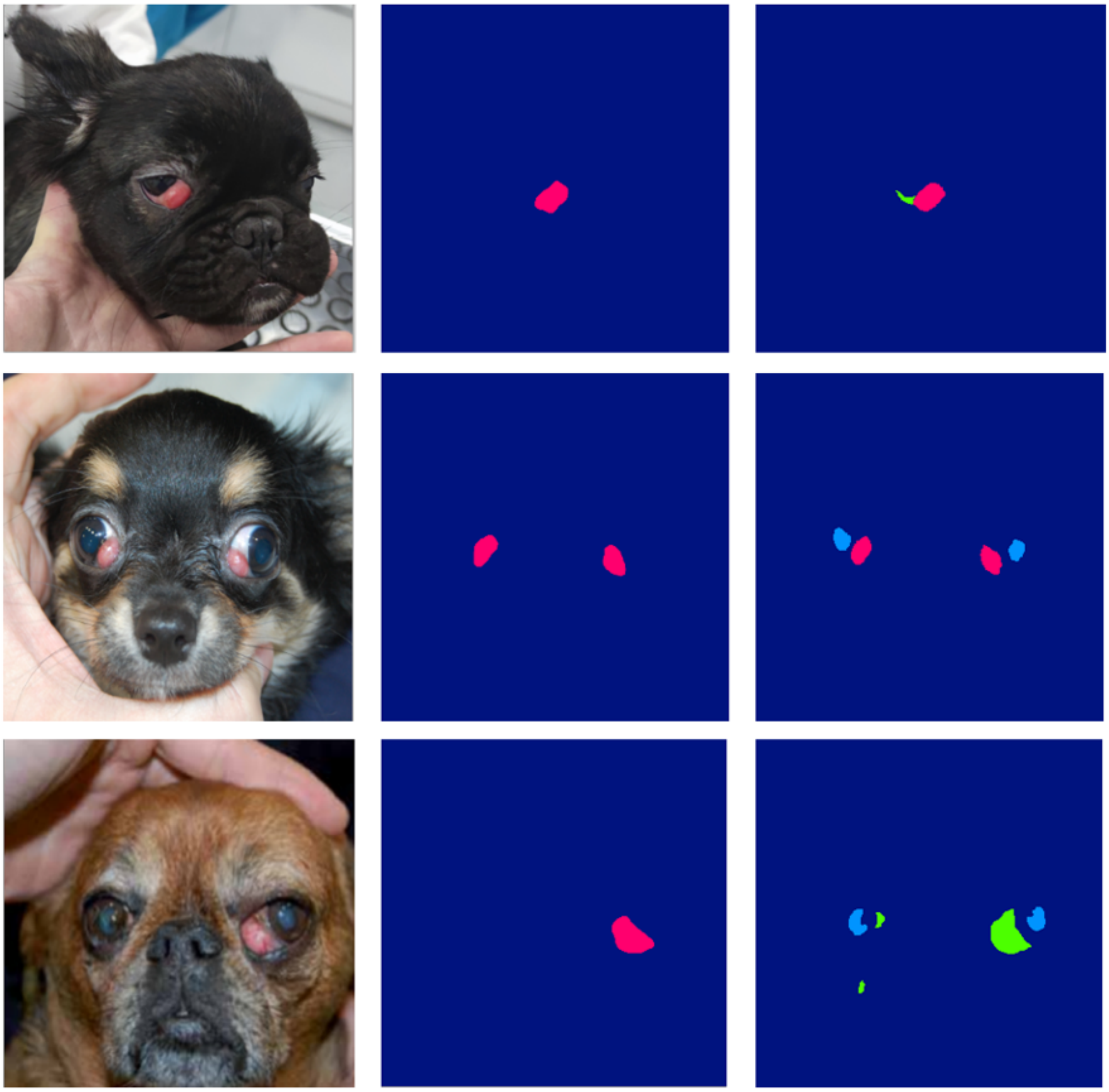


Figure 4.10: Segmentation of test images featuring whole head figures with diagnoses is depicted. The left column shows the original images, the center column displays the ground truth (GT), and the right column illustrates the predictions. In the top row, a colored mass in the corner of the eye is correctly predicted while redness of the sclera is mistakenly identified. The middle row incorrectly detects cloudiness or haziness of the cornea but successfully segments a colored mass in the corner of the eye. The bottom row falsely identifies a combination of redness of the sclera and cloudiness or haziness of the cornea where only a colored mass in the corner of the eye is actually present.

4.3.2 Experiment 2: Segmentation Comparative Study

In the second experiment, we conduct a comparative analysis of the segmentation capabilities of the U-Net models against Grounding SAM and GPT 4 segmentation models based on Large Language Models (LLMs). We use the best performing U-Net model (U-Net+ResNet34) described in Experiment 1.

Considering the limited number of images in the *DogEyeSeg4* dataset used for training the U-

Net+ResNet34 model, an additional model based on it is trained using DogEyeSeg4 dataset and synthetized images by Stable Diffusion [68]. We refer to the model trained solely on DogEyeSeg4 as U-Net+ResNet34(R), and the model trained on DogEyeSeg4 and synthesized images as U-Net+ResNet34(RSD) with ResNet34 backbone and Focal and Dice loss. For training the enhanced U-Net+ResNet34(RSD) model, the original DogEyeSeg4 dataset was expanded with an additional 400 synthetized images, 100 for each class, as seen in Figure 4.11.

For this and succeeding experiments the test dataset described in Section 4.1. will be used, that was not used in the k-fold cross validation training of the U-Net+ResNet34(R) model. By doing so we are avoiding bias towards trained U-Net models compared to others.

The test set encompasses a total of 25 images, categorically divided into five distinct disease groups. This study highlights the relative strengths and weaknesses of each model and evaluates their performance on a standardized test dataset comprising various stages of ocular disease progression.

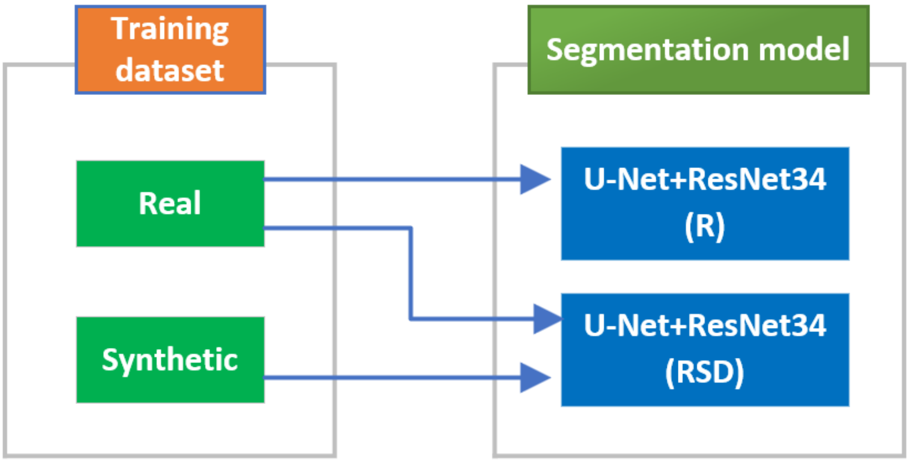


Figure 4.11: The training dataset used to develop the U-Net+ResNet34(R) model consists exclusively of real images, while the U-Net+ResNet34(RSD) model incorporates a combination of both real and synthetic images along with their corresponding masks.

The results of the U-Net models are compared with results from Grounding SAM [7] and GPT-4o [111]. Grounding SAM, which initiates segmentation by processing predefined bounding boxes determined by most favorable custom prompt for each case, where, for example, term "red mass protrusion" is used for S4 symptom instead of "A colored mass in the corner of the eye" as described in 3.16, allowing for a direct comparison with the U-Net outputs. The prompt used in GPT-4o is the one described in 3.14. For a better response prompt is divided into:

- Task (act as veterinary consultant),
- Instruction (Analyze image and return segmented medical symptoms) and
- Expected response (color the symptoms and make the background black).

Results from this evaluation are presented in Table 4.6, where U-Net+ResNet34(RSD) consistently shows superior performance across the metrics of Pixel Accuracy, Intersection over Union (IoU), and Dice Index (DSC). This suggests that the synthetic dataset contributed to improved segmentation accuracy on real images, enhancing the model's ability to handle complex features in real-world images. Additionally, results for GPT-4o and Grounding SAM indicate that these pre-trained, general-purpose models are inadequate for this specialized task, underscoring the need for task-specific adaptations in ocular disease segmentation.

Table 4.6: Segmentation evaluation of U-Net+ResNet34(R), U-Net+ResNet34(RSD), GPT-4o, and Grounding SAM methods using IoU, Dice Index (DSC), and Pixel Accuracy.

Method	**IoU**	**DSC**	**Pixel Accuracy**
U-Net+ResNet34(R)	0.74	0.83	0.94
U-Net+ResNet34(RSD)	**0.81**	**0.88**	**0.97**
GPT-4o	0.16	0.20	0.72
Grounding SAM	0.14	0.16	0.67

Further assessments of the U-Net (RSD) modelfocusing on specific diagnostic categories are documented in Table 4.7. Here, the symptom characterized as **S4** — "colored mass protrusion in the corner of the eye" — demonstrates the highest measures of Specificity, IoU, and Dice Index, highlighting the accuracy of U-Net+ResNet34(RSD) in segmenting this particular pathology. Conversely, the challenge is most evident with **S3** — "excessive tearing" — which proves difficult to segment accurately due to its more dispersed and irregular manifestation.

Table 4.7: Evaluation of U-Net+ResNet34(RSD) using IoU and Dice Index with standard deviation, along with Pixel Accuracy, Sensitivity, and Specificity on individual symptoms.

	IoU	**DSC**	**Pixel Accuracy**	**Sensitivity**	**Specificity**
S1	0.84 ± 0.08	0.91 ± 0.04	**0.991**	**0.944**	0.972
S2	0.74 ± 0.20	0.83 ± 0.15	0.976	0.928	0.973
S3	0.74 ± 0.26	0.82 ± 0.20	0.972	0.851	0.983
S4	$\mathbf{0.90 \pm 0.22}$	$\mathbf{0.92 \pm 0.20}$	0.984	0.932	**0.991**

These findings are visually summarized in Figure 4.12, illustrating the Dice Index for each symptom across the evaluated segmentation methods. U-Net+ResNet34(RSD) records the highest indices for three of the four symptoms tested, while U-Net+ResNet34(R) excels specifically in segmenting the "colored mass protrusion in the corner of the eye". This indicates that the advanced U-Net+ResNet34(RSD) typically provides enhanced performance. In contrast, GPT-4o, compared to Grounding SAM, performs better with symptoms that are irregularly shaped and scattered, but both GPT-4o and Grounding SAM are not suitable for segmentation purposes.

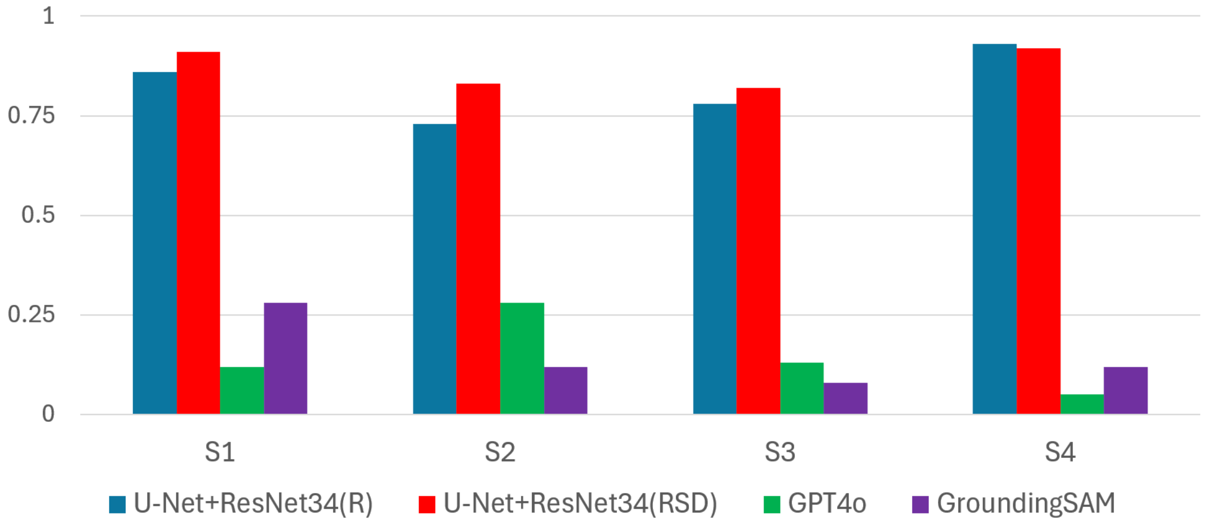


Figure 4.12: Dice Index of four different segmentation methods for individual classes. The U-Net+ResNet34(RSD) shows best results in three classes and closely follows result of U-Net+ResNet34(R) when forth class is observed.

4.3.3 Experiment 3: Binary vs. Multiclass Segmentation

The primary goal of this experiment is to assess whether training individual models for binary segmentation of each symptom class can outperform a multiclass segmentation approach in the context of canine ocular diseases. Given that a single model usually does not perform optimally across all symptoms, this experiment seeks to explore the potential benefits and limitations of specialized binary models compared to a generalized multiclass model. The experiment is divided into two phases:

1. **Binary Models Training**: Each class (**S1**, **S2**, **S3**, **S4**) was individually trained using a U-Net+ResNet34(R) architecture, designed to segment specific symptoms without consideration for overlapping conditions. The results can be observed in Table 4.8.
2. **Enhanced Binary Models with Heatmaps**: To refine the focus on the eye region and reduce false positives (FP), binary models were re-trained using U-Net+ResNet34(R) with an added heatmap channel. This channel includes three specific white spots: one centered in the eye and two at the lower corners of the bounding box around the eye. The bounding boxes were generated using a Single Shot Detector (SSD) object detection model trained on a set of 99 images that depict various canine ocular conditions. This SSD model helps localize the eye area accurately by identifying bounding boxes around the eyes, thus enhancing the precision of the segmentation by narrowing the model's focus to the regions most relevant for detecting ocular conditions. The results can be observed in Table 4.9.

Training individual binary models for each class offers the benefit of specialized focus, potentially improving model accuracy for each symptom. Binary models are computationally more efficient because each class is treated independently, meaning each model only needs to distinguish between the presence or absence of a specific symptom. This makes it easier to scale

Table 4.8: Evaluation of binary U-Net+ResNet34(R) models not considering class overlapping. Results are generally lower compared to multiclass U-Net+ResNet34(R) with significant decrease in sensitivity.

	IoU	DSC	Pixel Accuracy	Sensitivity	Specificity
S1b	**0.87 ± 0.41**	**0.90 ± 0.41**	**0.985**	**0.717**	0.989
S2b	0.59 ± 0.46	0.61 ± 0.46	0.980	0.395	**0.994**
S3b	0.38 ± 0.37	0.38 ± 0.37	0.970	0.190	0.972
S4b	0.70 ± 0.45	0.72 ± 0.47	0.980	0.563	0.981

Table 4.9: Evaluation of binary U-Net+ResNet34(R) models with Heatmaps. Heatmap includes three specific white spots: one centered in the eye and two at the lower corners of the bounding box around the eye, identified using an SSD object detector.

	IoU	DSC	Pixel Accuracy	Sensitivity	Specificity
S1b+h	0.60 ± 0.43	0.63 ± 0.46	0.987	0.457	0.990
S2b+h	0.64 ± 0.47	0.66 ± 0.46	0.980	0.440	**0.999**
S3b+h	0.64 ± 0.50	0.64 ± 0.50	**0.995**	0.440	**0.999**
S4b+h	**0.86 ± 0.39**	**0.90 ± 0.40**	0.992	**0.680**	0.993

and manage the models, especially when introducing new classes, as no dataset rebalancing is required. Each model can be trained separately, simplifying the integration of new symptoms without revisiting the previous dataset.

However, using binary models requires multiple models to be deployed to cover the full range of symptoms, which increases the overall complexity. Additionally, binary models face challenges when symptoms overlap. These models need extra mechanisms to handle such overlaps effectively, making the approach more complicated. Another drawback of binary models is their lack of contextual information. Since each model is focused on a single symptom, it may misclassify symptoms because it lacks information about the absence of other classes, which can lead to higher misclassification rates.

Figures 4.13 and 4.14 demonstrate the advantages and limitations of using heatmaps to reduce false positives outside the eye region. Figure 4.13 shows a close-up of an eye where the heatmap discards many false positives, and Figure 4.14 illustrates even more pronounced improvements in a whole-head image. However, focusing too narrowly on the eye region can result in false negatives, as seen in Figure 4.15, where excessive tearing remains unsegmented. Furthermore, despite the high IoU rate of the SSD object detector, it occasionally fails to detect eyes, as shown in Figure 4.16, which directly impacts subsequent segmentation results.

Despite the promising results from binary models, including a reduction in false positives when

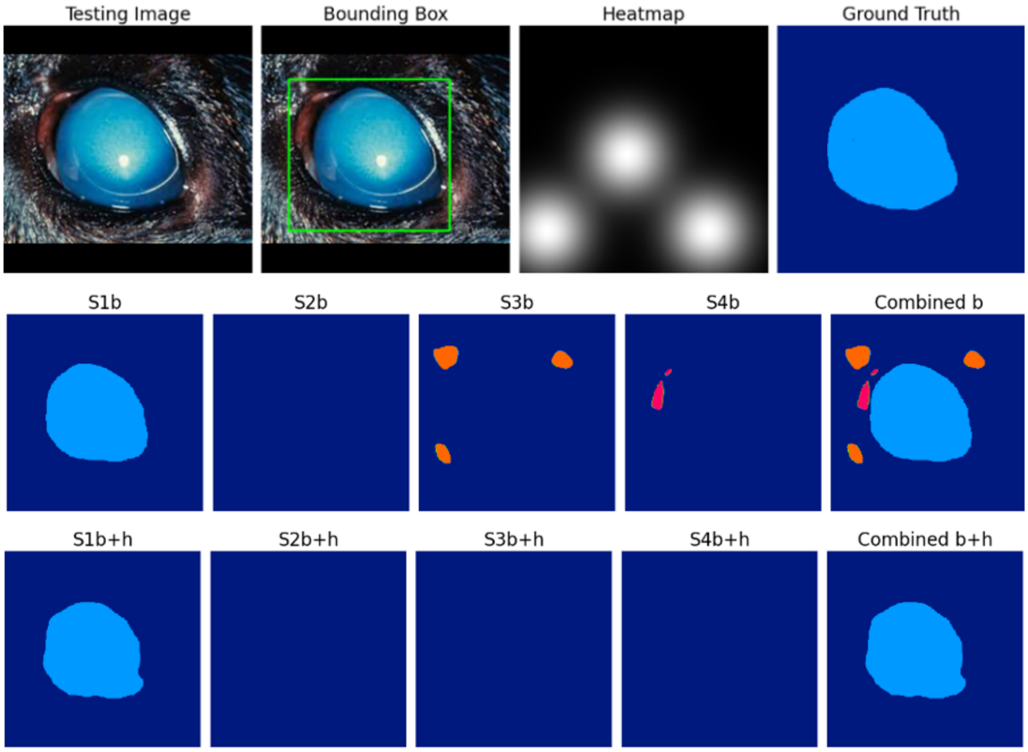


Figure 4.13: Example of binary segmentation without (2nd row) and with (3rd row) heatmap as 4 channel input.

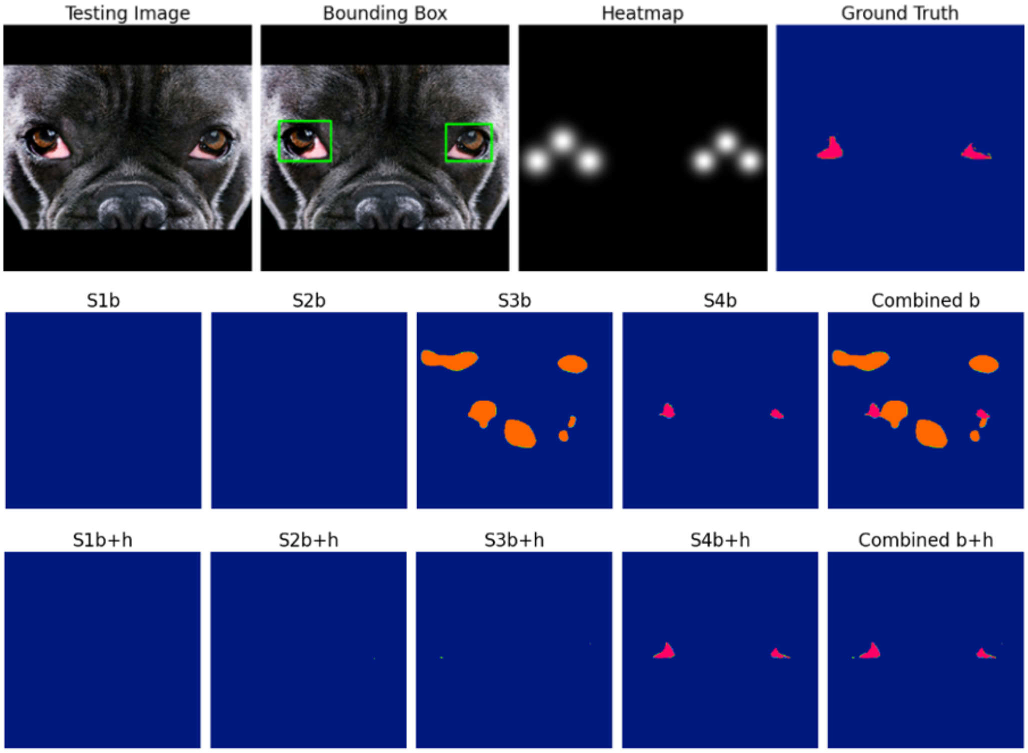


Figure 4.14: Example of binary segmentation on whole head without (2nd row) and with (3rd row) heatmap as 4 channel input

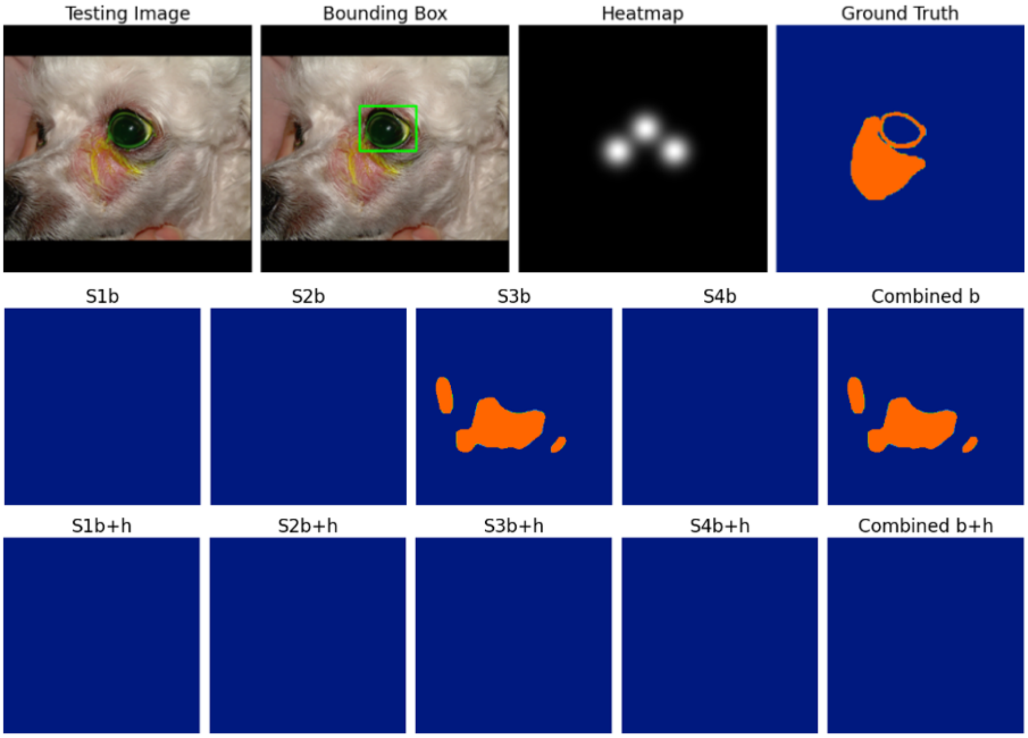


Figure 4.15: Binary segmentation example showing that focusing too narrowly on the eye region can result in false negatives (3rd row) compared to partly segmented excessive tearing (2nd row).

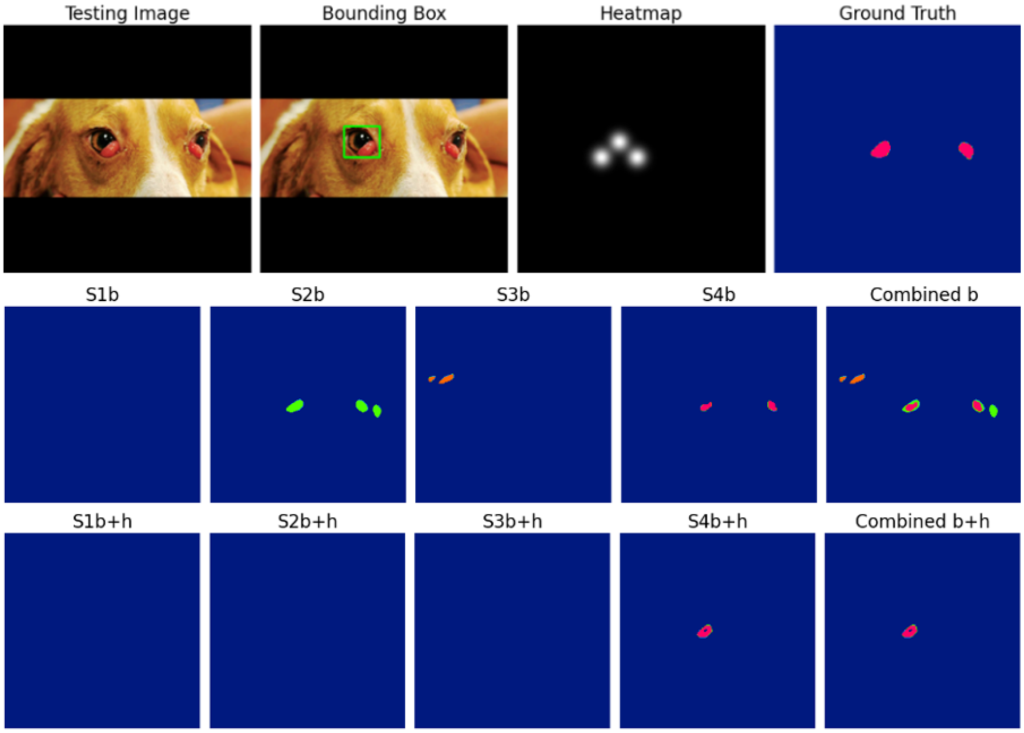


Figure 4.16: Example where lack of detection reflects in absence of segmentation of a right eye (3rd row) but increases FPs if not applied at all (2nd row).

using heatmaps, the complexity involved in managing overlapping symptoms and the increased operational demands of maintaining multiple models led to the decision to pursue multiclass segmentation. Multiclass models can handle overlapping conditions more naturally, avoiding the need for multiple models, and offer a more streamlined and effective approach for diagnosing canine ocular diseases. This makes multiclass segmentation the preferred approach for practical deployment in veterinary ophthalmology.

4.3.4 Experiment 4: Diagnostic Interpretation

In the fourth experiment, we compare the performance of various Large Language Models (LLMs)—including GPT-4, Gemini, Mistral, Claude, and Llama-3—in analyzing and interpreting combinations of four ocular symptoms that U-Net+ResNet34(RSD) successfully classifies and segments. This study aims to assess each model's ability to understand complex medical data and generate accurate diagnostic insights, thereby identifying the most effective LLM for supporting ophthalmological diagnostics.

To assess the diagnostic capabilities of large language models (LLMs) in interpreting and providing preliminary diagnoses for canine eye diseases, this experiment utilized 15 distinct medical cases selected from the *Medical Atlas of Canine and Feline Eye Diseases* [61]. Each "medical case" in this context includes:

- An image sourced from the atlas, displaying specific ocular symptoms associated with a disease.
- A validated diagnosis, with up to three words (for example: Blocked Tear Duct, Ocular Trauma Glaucoma), for each image, as specified in the atlas.

Each image underwent segmentation using a U-Net+ResNet34(RSD) model with four symptom classes (S1 to S4). If U-Net+ResNet34(RSD) successfully segmented only the symptoms present in the atlas-defined diagnosis, with no extra or missing symptoms, that image was used as a reference. This approach yielded 15 images, each corresponding to a unique symptom combination (e.g., S1, S2, S3, S4, S1+S2, etc., up to S1+S2+S3+S4).

In the LLM testing phase, only the segmented symptom combinations, not the images themselves, were provided as input prompts to the LLMs, as some models cannot process images. Each LLM output was then compared to the diagnosis in the atlas to determine diagnostic alignment. Scoring was based on the following criteria:

- A score of "1" was assigned if the LLM's diagnosis matched the atlas diagnosis.
- A score of "-1" if the LLM's response differed.
- A score of "0" if the LLM did not specify a diagnosis or could not reach a conclusion.

This scoring system allowed for an evaluation of the LLMs' diagnostic reliability based solely on the symptom prompts identified through U-Net segmentation.

The experimental design incorporated several stages to evaluate the LLMs. Initially, we generated queries mimicking questions posed by dog owners lacking medical expertise, and these were compared with symptom-based queries derived from U-Net image segmentation intended for use by veterinary specialists. An illustrative example from this stage, represented by the Mistral LLM's performance, is documented in Table 4.10.

Table 4.10: An illustrative example comparing the results generated by the Mistral LLM when prompted by a dog owner without specifying any symptoms with the outputs from U-Net image segmentation that utilized symptom information akin to that provided by a veterinarian. Glaucoma is the correct diagnosis.

Prompt by	Diagnosis generated by the Mistral LLM	Score
Owner	I'm sorry to hear that your dog is having eye issues. Blurry and pink eyes in dogs can be caused by a variety of factors, such as allergies, eye injuries, infections (bacterial, viral, or fungal). It's important to get a proper diagnosis from a veterinarian as soon as possible, as some eye conditions can progress rapidly and potentially lead to vision loss.	0
U-Net symptoms	The symptoms of ocular opacity, sclera redness, and excessive tearing in a dog could be associated with conditions such as anterior uveitis, lens luxation, or glaucoma, and a professional veterinarian's examination is necessary for an accurate diagnosis and appropriate treatment.	1

Owner Prompt1: "I've noticed something odd with my dog's eye, and I'm really worried. It has blurry and pink eye. Should I be worried?"
U-Net symptoms Prompt2: "Dog has next symptoms: ocular opacity, sclera redness, and excessive tearing. Give me possible diagnosis."

In the next stage, the evaluation focused on understanding the diagnostic effectiveness of LLMs when utilizing structured, symptom-specific prompts. Each of the 15 medical cases selected from the Medical Atlas of Canine and Feline Eye Diseases was assigned a "veterinary-style" prompt. These prompts included only the symptoms that U-Net had accurately segmented, which corresponded directly to the conditions documented in the atlas.

These symptom-based prompts, identified from U-Net segmentation, aimed to provide precise symptom descriptions that would closely align with those a veterinary specialist might use. Structured prompts consisted of a brief contextual background, a direct query, and a list of symptoms. For example, the prompt format was: "*Make a short pinpoint diagnosis based on the combination of these dog eye symptoms to assist a vet in preliminary decision-making for diagnosis: 1. <class returned by U-Net>, 2. <class returned by U-Net>.*" This approach allowed the experiment to test the accuracy of LLM diagnoses based on symptom-specific input.

In example where two symptoms were presented to LLMs as described in Table 4.11, to evaluate the responses, scoring rules were applied consistently across cases: a score of "1" indicated a diagnosis matching the Cherry Eye diagnosis, "-1" for mismatches, and "0" for responses without a specific diagnosis or inconclusive results. These scoring criteria were used to quantify the diagnostic reliability of the LLMs.

The role of U-Net segmentation in this experiment was to ensure that each prompt accurately reflected the relevant symptoms present in each case, as identified from the atlas images. U-Net was responsible for identifying specific symptoms (S1-S4) in the atlas images and only cases where U-Net accurately segmented symptoms matching the ground truth (100% alignment with atlas-documented symptoms) were used.

Table 4.11: Example of prompts given to five LLM models, along with their responses and associated score values. The correct diagnosis in this case is Cherry Eye.

LLM	**Response to Prompt***	**Score**
Llama-3	Redness of the sclera + Colored Mass Protrusion in the Eye Corner Diagnosis: Allergic Reaction	-1
Mistral	Sclera redness + Colored Swelling in the Eye Edge: Although less commonly seen together, coexisting watery eyes and colored swelling may warrant referral to a specialized veterinarian for proper evaluation and management of suspected immune-mediated disorders affecting multiple organs.	0
GPT-4	Redness of the sclera + Protrusion of a Red or Pink Mass - Diagnosis: Cherry Eye with secondary tear duct irritation.	1
Claude	Based on the symptoms you described, I do not have enough information to make a definitive diagnosis. The redness and protruding mass indicate an abnormality affecting the eye and surrounding structures. But the specific diagnosis cannot be made based on these limited symptoms alone. Further veterinary examination of the eye would be recommended.	0
Gemini	The combination of red sclera and a red/pink mass protruding from the eye corner strongly suggests glaucoma. This condition usually requires veterinary examination and potentially surgery.	-1

Prompt: "Make a short pinpoint diagnosis based on the combination of these dog eye symptoms to assist a vet in preliminary decision-making for diagnosis:
1. Redness of the sclera
2. Protrusion of a Red or Pink Mass in the Eye Corner"

The scores for all 15 cases are presented in Table 4.12.

GPT-4 demonstrates effectiveness in straightforward cases but has limitations in identifying

Table 4.12: Comparison of LLMs for Diagnostic Evaluation of a Specific Disease Using Symptom Criteria individually and in combination.

Sym.	**Disease Medical Condition**	**GPT-4**	**Gemini**	**Mistral**	**Claude**	**Llama-3**
S1	Nuclear Sclerosis	1	0	1	-1	-1
S2	Blocked Tear Duct	1	-1	-1	0	-1
S3	Conjunctivitis	1	-1	-1	-1	-1
S4	Cherry Eye	1	-1	-1	1	-1
S12	Corneal Ulcer	1	-1	0	1	-1
S13	Uveitis	1	1	1	1	-1
S14	Corneal Ulcer	-1	-1	0	0	-1
S23	Conjunctivitis	1	1	1	1	-1
S24	Cherry Eye	1	-1	0	0	-1
S34	Conjunctivitis	-1	-1	0	1	1
S123	Glaucoma	-1	-1	1	1	-1
S124	Bacterial Infection	-1	-1	0	-1	-1
S134	Ocular Trauma	-1	1	0	0	-1
S234	Blocked Tear Duct	1	-1	0	-1	-1
S1234	Ocular Trauma	-1	0	0	0	0

Sym.: Symptom combinations, where S1-S4 are individual symptoms, and combinations represent multiple symptoms.

more complex conditions, particularly when symptoms overlap, as evidenced by its failure to diagnose Corneal Ulcer. Mistral approaches diagnoses with caution, giving undefined answers more often than other LLMs. It is less prone to overconfidence and therefore has fewer hallucinations. Claude performs well but has difficulties with multisymptomatic diseases, as indicated by its variable scores in conditions like Blocked Tear Duct and Ocular Trauma. In comparison, Llama-3 and Gemini struggle across various scenarios. Notably, Llama-3 doesn't recognize excessive tearing as a symptom at all, resulting in a diagnosis that describes a healthy eye. The most successful LLMs are GPT-4 and Claude.

In order to provide a more generalized diagnosis, a dual diagnosis prompt is tested to accommodate the fact that a combination of symptoms does not necessarily point to a unique disease. This approach sacrifices precision for a broader range of possible medical conditions, which is often desired in medical settings. A Table 4.13 shows combinations of symptoms and first and second most probable diagnoses from LLMs. In case when resulting diagnosis are inconclusive word "None" is used.

Table 4.13: Comparison of LLM Responses for Dual Diagnostic Evaluation

Symptom	Diagnosis	GPT-4	Gemini	Mistral	Claude	Llama-3
S1	1st	**Nuclear Sclerosis**,	None,	**Nuclear Sclerosis**,	Cataract,	Cataract,
	2nd	Cataract	None	Cataract	Lens luxation	None
S2	1st	**Blocked Tear Duct**,	Uveitis,	Epiphora,	None,	Dry Eye Syndrome,
	2nd	Cherry Eye	None	Entropion	None,	None
S3	1st	**Conjunctivitis**,	Scleritis,	Scleritis,	Scleritis,	Scleritis,
	2nd	Uveitis	None	Uveitis	Uveitis	None
S4	1st	**Cherry Eye**,	Proptosis,	Glaucoma,	**Cherry Eye**,	Glaucoma,
	2nd	Orbital Neoplasia	Glaucoma	**Cherry Eye**	Proptosis	**Cherry Eye**
S12	1st	**Ulcer**,	Scleritis,	None,	**Corneal Ulcer**,	Cataract,
	2nd	Nuclear Sclerosis	None	None	Cataract	Dry Eye
S13	1st	**Uveitis**,	**Uveitis**,	**Uveitis**,	**Uveitis**,	Cataract,
	2nd	Nuclear Sclerosis	Scleritis	Lens Luxation	Glaucoma	Scleritis
S14	1st	Nuclear Sclerosis,	Cataract,	None,	None,	Cataract,
	2nd	Cherry Eye	Proptosis	None	None,	Cherry Eye
S23	1st	**Conjunctivitis**,	**Conjunctivitis**,	**Conjunctivitis**,	**Conjunctivitis**,	Dry Eye,
	2nd	Uveitis	Blocked Tear Duct	Uveitis	Epiphora	**Conjunctivitis**
S24	1st	**Cherry Eye**,	Proptosis,	None,	None,	Dry Eye,
	2nd	Blocked Tear Duct	**Cherry Eye**	None	None,	None
S34	1st	Uveitis,	Scleritis,	None,	**Conjunctivitis**,	**Conjunctivitis**,
	2nd	Cherry Eye	Proptosis	None	Epiphora	Dry Eye
S123	1st	Nuclear Sclerosis,	Ocular Trauma,	**Glaucoma**,	**Glaucoma**,	Cataract,
	2nd	Uveitis	None	Cataract	Cataract	Dry Eye
S124	1st	Nuclear Sclerosis,	Cataract,	None,	Cataract,	Cataract,
	2nd	Cherry Eye	Proptosis	None	Glaucoma	Dry Eye
S134	1st	Nuclear Sclerosis,	**Ocular Trauma**,	None,	None,	Cataract,
	2nd	Cherry Eye	Cataract	None	None,	Scleritis
S234	1st	**Blocked Tear Duct**,	Proptosis,	None,	Cataract,	Dry Eye,
	2nd	Conjunctivitis	Glaucoma	None	Glaucoma	Scleritis
S1234	1st	Nuclear Sclerosis,	Glaucoma,	None,	None,	None,
	2nd	**Trauma**	**Eye Trauma**	None	None	None

The performance metrics of LLMs in diagnosing diseases based on clinical symptoms are detailed in Table 4.14. This evaluation covers both single and dual-case scenarios, assessing the most probable medical condition in single-case diagnosis, and comparing the two most probable conditions to the reference standard in dual-case diagnosis.

Table 4.14: Evaluation of Large Language Models (LLMs) using various text similarity metrics.

LLM	**BERTScore**	**CLIPScore**	**BLEU**	**METEOR**	**ROUGE**	**SPICE**
Single Diagnosis						
GPT-4	**0.80**	**0.91**	**0.52**	**0.42**	**0.56**	**0.56**
Gemini	0.77	0.85	0.20	0.13	0.20	0.20
Mistral	0.75	0.86	0.27	0.19	0.27	0.27
Claude	0.76	0.87	0.40	0.26	0.40	0.40
Llama-3	0.69	0.85	0.07	0.06	0.07	0.07
Dual Diagnosis						
GPT-4	**0.69**	0.86	**0.22**	**0.25**	**0.27**	**0.43**
Gemini	0.68	0.85	0.15	0.20	0.19	0.23
Mistral	0.66	0.85	0.10	0.11	0.12	0.20
Claude	0.66	0.84	0.19	0.21	0.24	0.28
Llama-3	**0.69**	**0.87**	0.07	0.07	0.09	0.15

GPT-4 demonstrated superior performance across all assessed metrics in single diagnostic scenarios, followed by Claude, Mistral, Gemini, and Llama-3. Notably, BERTScore and CLIPScore, in contrast to other metrics, showed minimal variation among LLMs. The disparity between the highest and lowest scores was most pronounced when using ROUGE and SPICE metrics, compared to a narrower range observed in CLIPScore.

Computational metrics as shown in Table 4.14, largely correspond with human evaluations, where terms with slight variations (such as 'Ocular Trauma' being synonymous with 'Eye Trauma' and 'Ulcer' broadly matching 'Corneal Ulcer') were considered correct interpretations in context. These results, illustrated in Figure 4.17, provide further validation of the LLMs' diagnostic accuracy, even when exact terminology may differ.

The evaluation noted that GPT-4 yielded the most accurate diagnoses, outperforming other LLMs in both single and dual diagnosis scenarios. The dual diagnosis analysis across all LLMs using BERTScore illustrated in Figure 4.18 reveal that the diagnosis of Cherry Eye when accompanied by the symptom of Colored Ocular Protrusion and Redness of the sclera was most challenging while combination of Redness of the sclera and Ocular Tearing seems to be easier to diagnose. In Figure 4.19, the differences between single and dual diagnosis evaluations using CLIPScore on GPT-4 are presented.

Single diagnosis scenarios, despite being less successful in pinpointing the correct diagnosis, showed higher certainty in predictions—a reflection of the strategic inclusion of error as a result

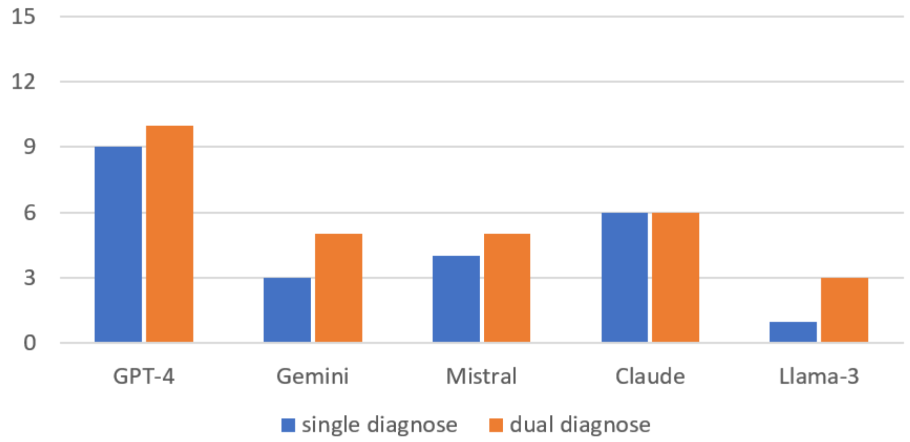


Figure 4.17: True positive diagnoses by each LLM across 15 cases, comparing single and dual diagnoses.

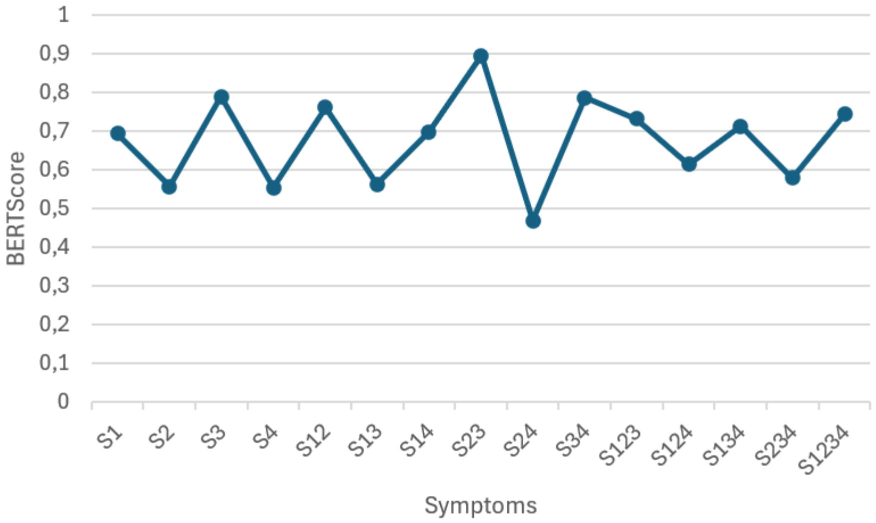


Figure 4.18: Mean dual diagnosis BERTScore across all LLMs evaluated for each symptom combination.

of an additional text in a form of second diagnosis. Both single and dual diagnosis perform better when single symptoms are present with exception of combination of Redness of the sclera and excessive tearing and colored mass in the corner of the eye. Cloudiness or haziness of the cornea symptom gets better recognized in dual disease diagnosis.

This evaluation underscores the diagnostic capabilities of LLMs in interpreting symptom data processed by U-Net, with GPT-4 demonstrating the strongest performance across multiple cases. Semantic and linguistic metrics such as BERTScore, CLIPScore, BLEU, METEOR, ROUGE, and SPICE support the reliability of GPT-4's outputs in a medical context. In contrast, Llama-3 showed limited accuracy, successfully diagnosing only a small number of cases, indicating that not all evaluated models currently meet the threshold for practical utility.

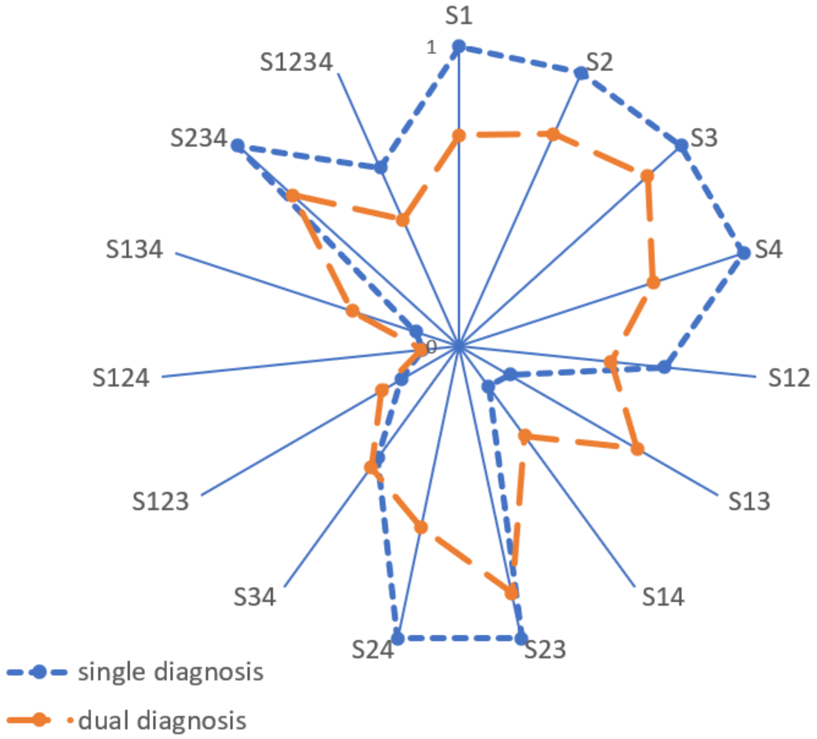


Figure 4.19: An analysis of the differences between results provided using single and dual diagnoses, using CLIPScore metric, evaluated on GPT-4 across various symptom combinations.

4.3.5 Experiment 5: Combined Approach for Complete Diagnosis

The final experiment considers the interpretative capabilities of GPT-4 and its successor GPT-4o[111], where non-segmented images and the images segmented by U-Net+ResNet34(RSD) are analyzed to generate comprehensive diagnostic reports. The goal is to test the ability of these language models to synthesize visual data into accurate, actionable medical advice.

The experiment uses the test set from the 4.1. Test Dataset section, consisting of 25 images representing five distinct canine eye diseases. These images are used as inputs for both GPT-4 and GPT-4o, with and without segmentation masks generated by U-Net+ResNet34(RSD). The mask and input image are supplied separately using two PNG files where the mask is similar to one presented in 4.2. The objective is to evaluate the models' diagnostic capabilities across single and dual diagnosis scenarios, specifically examining whether the additional segmentation masks enhance diagnostic accuracy for the large language models.

Results from Table 4.15 provide a detailed comparison of single and dual medical diagnoses generated by GPT-4 and GPT-4o, with and without the aid of segmentation masks. Across all evaluated metrics—BERTScore, CLIPScore, BLEU, METEOR, ROUGE, and SPICE—the performance generally improves when GPT-4o is supplemented with segmentation masks, highlighting the model's enhanced capability in interpreting the structured visual data.

The comparative analysis of disease diagnosis by GPT-4 and GPT-4o, as illustrated in Figure 4.20, where 20 cases of true positives corresponds to 5 disease in single and dual diagnosis

Table 4.15: Evaluation of GPT-4 and GPT-4o in single and dual medical diagnoses, with and without masks generated by U-Net+ResNet34(RSD) segmentation.

Model	BERTScore	CLIPScore	BLEU	METEOR	ROUGE	SPICE
Single Diagnosis						
GPT-4o image	0.83	0.90	0.48	0.31	0.48	0.48
GPT-4o with mask	**0.84**	**0.91**	**0.49**	**0.32**	**0.51**	**0.51**
GPT-4 image	0.63	0.84	0.08	0.06	0.08	0.08
GPT-4 with mask	0.65	0.86	0.15	0.08	0.16	0.15
Dual Diagnosis						
GPT-4o image	0.62	0.90	0.14	0.13	0.17	0.45
GPT-4o with mask	0.64	0.91	0.16	0.14	0.19	0.49
GPT-4 image	0.59	0.85	0.07	0.05	0.08	0.11
GPT-4 with mask	0.59	0.85	0.03	0.03	0.04	0.06

scenarios with and without segmentation mask, underscores the superior diagnostic accuracy of GPT-4o across multiple eye diseases. This model excels particularly in the diagnosis of Uveitis and Cherry Eye, in cases of single/dual diagnosis and with/without segmentation masks, indicating a robust capability to interpret complex visual patterns associated with these conditions. However, an divergence is noted in the diagnosis of Corneal Ulcer, where GPT-4 outperforms GPT-4o, suggesting potential limitations in GPT-4o's processing of the visual cues specific to this condition.

Further insights are gleaned from the distribution of BERT Scores across different diseases based on results of GPT-4o in Figure 4.21, which charts the results on a logarithmic scale for both single and dual diagnoses. The radial plot illustrates that while GPT-4o generally achieves higher scores in single diagnosis scenarios—particularly for Cherry Eye—the model also shows slight advantages in dual diagnosis contexts for more complex diseases like Bacterial Keratitis and Corneal Ulcer.

This analysis demonstrates that the integration of advanced language processing models with image segmentation technologies like U-Net+ResNet34(RSD) enables consistent diagnostic accuracy based on visual data. It also highlights the critical role of model selection and the configuration of input data, which consists of engineered prompt requesting two most probable diagnosis using only input image and using input image and segmented mask generated by U-Net+ResNet34(RSD) in optimizing diagnostic outcomes in medical AI applications.

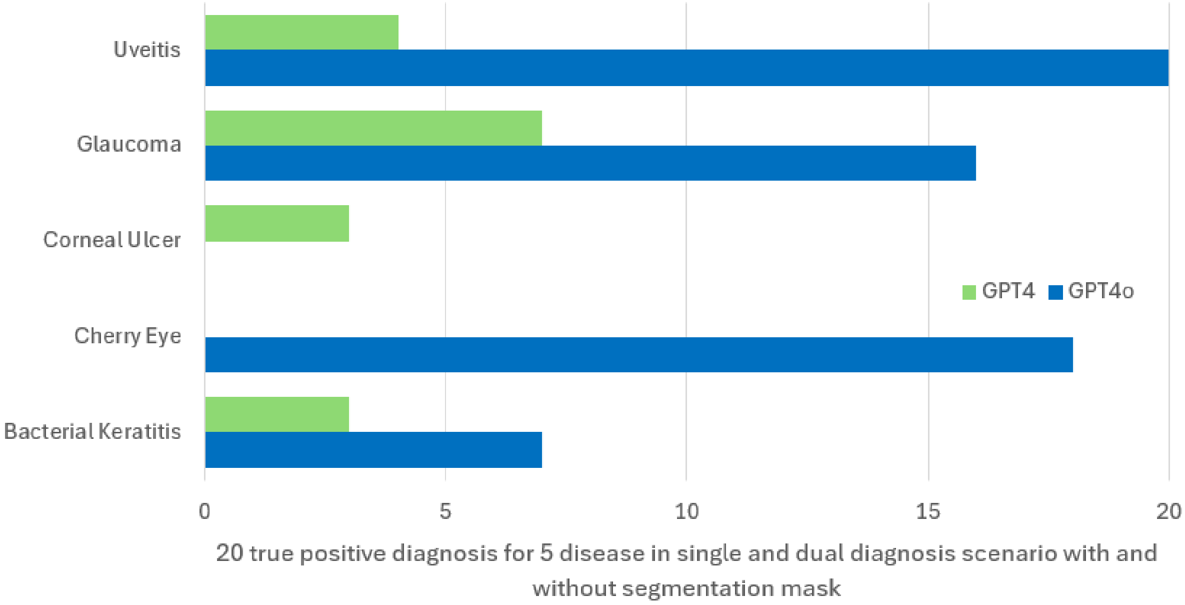


Figure 4.20: An Evaluation of the Diagnostic Differences Between GPT-4 and GPT-4o in five Canine Eye Diseases showing performance in favor of GPT-4o in four out of five diseases. Uveitis was diagnosed by GPT-4o with 100% prediction rate in single and dual diagnosis and with and without provided U-Net+ResNet34(RSD) segmentation masks.

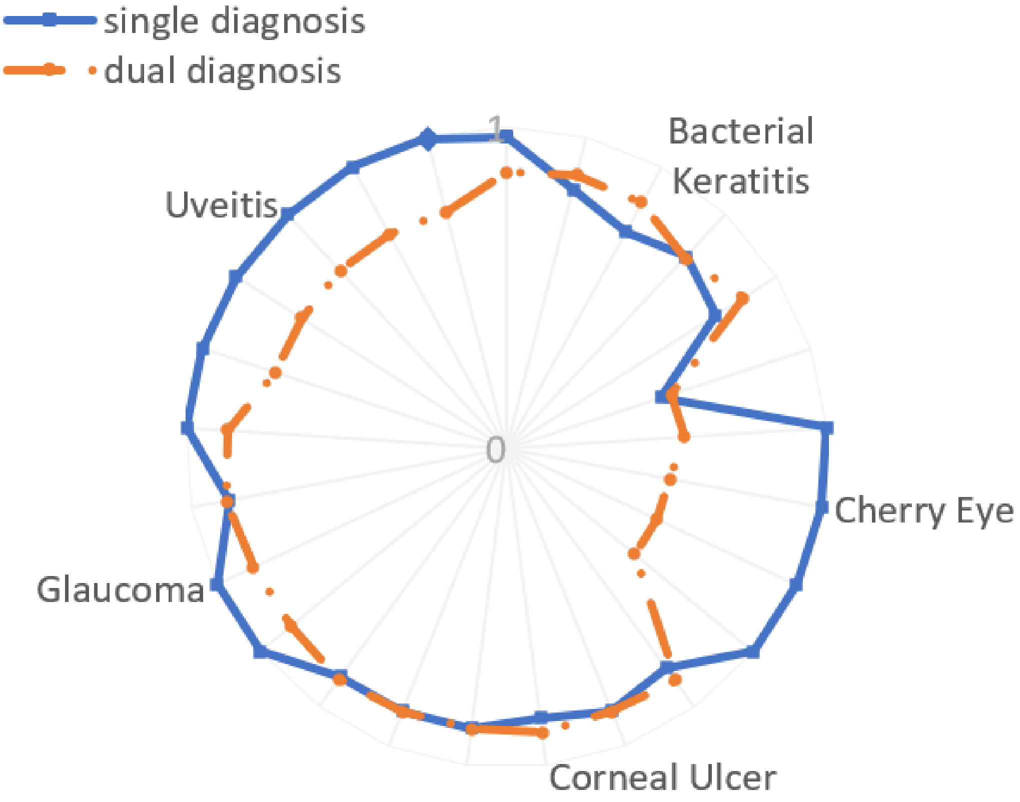


Figure 4.21: BERTScore results for each medical case, categorized into four disease groups, are presented based on GPT-4o when both single and dual diagnosis approaches are applied.

Chapter 5

Discussion

This study was guided by two primary hypotheses:

1. **Hypothesis 1**: A computer vision model can recognize certain canine ocular conditions in still images taken in an unconstrained environment.
2. **Hypothesis 2**: Modification of the input and architecture of the U-Net network contributes to a better segmentation of canine eye conditions.

In the *Experiments and Results* section, we provided strong evidence supporting both hypotheses. We developed and evaluated a series of models for detection, segmentation, and interpretation tasks to diagnose canine ocular diseases from images captured in real-world, unconstrained environments.

One of the contributions of this research was the development and publication of the *DogEyeSeg4* dataset. The published custom-made *DogEyeSeg4* dataset was important in training the models. It provided a diverse set of images reflecting real-world clinical conditions, including variations in lighting, breed, and image quality. The dataset consists of 320x320 pixel annotated images using ground truth masks in PNG file format where particular class is determined by specific pixel value. It covers four classes of canine ocular symptoms including background, with annotations for both symptom types and regions. It incorporates a variety of breeds and clinical conditions, ensuring diversity in terms of lighting, breed differences, and image quality. This diversity was enhanced further by additional 400 synthetic images, on top of the DogEyeSeg4 dataset allowing the models to learn from a broader range of symptom presentations.

Given the lack of pre-existing models for detecting canine eyes, we trained a custom Single Shot Multibox Detector (SSD) model. This SSD model was trained on 99 images encompassing various breeds and medical conditions, including healthy eyes. The model achieved the following scores: IoU 92%, Dice Score 96%, Precision 97%, Recall 84%, and F1 Score of 90%. These metrics indicate high accuracy in localizing the eye region within unconstrained images, confirming that a computer vision model can effectively detect canine eyes in real-world conditions, which is needed for the experiment with the custom attention mechanism used by the

U-Net segmentation.

For symptom segmentation, we initially implemented the standard U-Net architecture and then enhanced it with four different backbone networks: ResNet34, Inception V3, VGG16, and EfficientNet B3. These models were trained using the following loss functions: Categorical Cross-Entropy (CCE), Dice Loss (DL), Focal Loss (FL), and Combined Dice and Focal Loss (DL+FL). The best performing model is the U-Net model with a ResNet34 backbone trained using the combined Dice and Focal Loss (U-Net + ResNet34 with DL+FL).

In 5-fold cross-validation, it achieved:

- Mean IoU: 74%
- Dice Similarity Coefficient (DSC): 83.4%
- Sensitivity: 86.3%
- Pixel Accuracy: 97.2%

The U-Net + ResNet34 with DL+FL outperformed other configurations in three out of four classes and demonstrated robust performance in accurately segmenting key ocular symptoms:

S1 Cloudiness or haziness of the cornea,
S2 Redness of the sclera,
S3 Excessive tearing, and
S4 A colored mass in the corner of the eye.

Further enhancements to the segmentation model were made by training on a combined dataset of real images from *DogEyeSeg4* and 400 synthetic images generated using Stable Diffusion, referred to as U-Net+ResNet34(RSD). The synthetic dataset was designed to ensure high quality, resembling real-world symptom presentations. These quality standards enabled synthetic images to complement real images effectively, improving model performance. The U-Net+ResNet34(RSD) model achieved IoU 81%, DSC 88%, and Pixel Accuracy 97%. These improvements underscore the effectiveness of augmenting training data with synthetic images to enhance model generalization and performance.

The U-Net+ResNet34(RSD) model was compared with other segmentation approaches:

- **GPT-4o**: IoU 16%, DSC 20%, and Pixel Accuracy 72%.
- **Grounding SAM**: IoU 14%, DSC 16%, and Pixel Accuracy 67%.

The U-Net+ResNet34(RSD) model outperformed these models across all metrics, highlighting its superior capability in segmenting canine ocular symptoms in unconstrained environments.

We explored training individual binary segmentation models for each symptom versus a single multiclass model. Binary models were trained using heatmaps as a fourth channel in the RGB input image, generated using information from SSD eye detection. While binary models with heatmaps reduced false positives compared to binary models trained without the custom attention mechanism, they faced challenges with overlapping symptoms and increased complexity. The multiclass U-Net+ResNet34(RSD) model provided better overall performance and operational efficiency, supporting its use for practical applications.

For diagnostic interpretation, several Large Language Models (LLMs) were evaluated: GPT-4, Gemini, Mistral, Claude, and Llama-3. GPT-4 consistently provided the most accurate diagnoses based on the segmented symptoms. For single-diagnosis scenarios, GPT-4 achieved:

- BERTScore: 0.80
- CLIPScore: 0.91
- BLEU: 0.52
- METEOR: 0.42
- ROUGE: 0.56
- SPICE: 0.56

When integrating segmentation masks from U-Net+ResNet34(RSD), the succeeding model GPT-4o demonstrated enhanced performance:

- BERTScore: 0.84
- CLIPScore: 0.91
- BLEU: 0.49
- METEOR: 0.32
- ROUGE: 0.51
- SPICE: 0.51

These results indicate that combining precise segmentation with advanced LLMs like GPT-4o improves diagnostic interpretation. The proposed pipeline is illustrated in Figure 5.1.

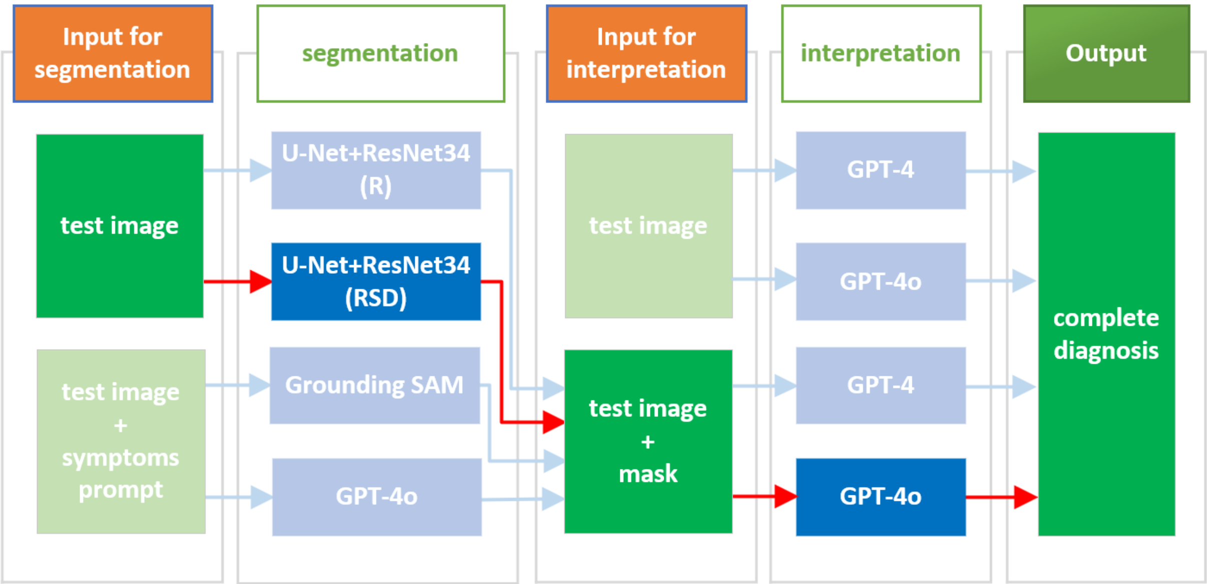


Figure 5.1: The depicted diagnostic pipeline for canine ophthalmic conditions demonstrates improved performance when both the original images and their corresponding segmented masks from U-Net+ResNet34(RSD) are utilized as inputs for GPT-4o, enhancing the accuracy of the medical diagnoses derived from visual symptoms.

The limitations identified in this study include:

- **Model's Sensitivity to Image Quality**: The diagnostic accuracy is highly dependent on the quality of input images and segmentation masks. Poor image quality or inaccurate segmentation can negatively impact model performance.
- **Difficulty with Certain Conditions**: Models occasionally struggled with conditions like Corneal Ulcer, suggesting the need for further refinement in handling less visually distinct symptoms.
- **Variability in LLM Capabilities**: Some LLMs, like Llama-3, underperformed in diagnostic interpretation, indicating variability in model capabilities and the necessity for continuous evaluation and improvement of these models.

Future research should focus on:

- **Extending the Dataset**: Incorporating more diverse images and medical symptoms, including rare conditions and different breeds, to improve model generalization.
- **Advanced Model Architectures**: Exploring the use of transformer-based models for segmentation to capture global contextual information, which could be beneficial for detecting subtle symptoms.
- **Multimodal Data Integration**: Combining image data with clinical history and other modalities to provide a more holistic diagnostic approach.
- **Clinical Testing**: Testing the diagnostic pipeline in clinical settings to validate and refine the models based on practical use, which should provide valuable feedback for further development.

The experimental results confirm both hypotheses:

- **Hypothesis 1** is supported by the successful development of the computer vision U-Net+ResNet34(RSD) model for segmentation and implementation of GPT-4o for interpretation, which accurately recognize canine ocular conditions in images taken in unconstrained environments.
- **Hypothesis 2** is confirmed by demonstrating that modifications to the U-Net network's input and architecture—by using a combination of real and synthetic images, custom attention mechanisms in the form of heatmaps, integrating advanced backbones, and employing combined loss functions—lead to better segmentation of canine eye conditions.

This study demonstrates the potential of integrating advanced computer vision and language models to assist in the diagnosis of canine ocular diseases. The findings lay the groundwork for developing robust, AI-driven diagnostic tools that can operate effectively in real-world veterinary settings, ultimately contributing to improved animal healthcare outcomes.

Chapter 6

Conclusion

This research has confirmed both initial hypotheses:

1. **Hypothesis 1**: By developing the computer vision U-Net+ResNet34(RSD) model for segmentation and implementing GPT-4/GPT-4o for interpretation, we demonstrated that certain canine ocular conditions can be accurately recognized in still images taken in unconstrained environments.
2. **Hypothesis 2**: Modifying the U-Net network's input and architecture by using a custom attention mechanism (heatmaps), integrating advanced backbones, and employing combined loss functions resulted in improved segmentation of canine eye conditions.

These confirmations highlight the effectiveness of advanced deep learning techniques in diagnosing canine eye diseases from images captured in real-world settings. The key contributions of this research are:

1. **An image dataset for machine learning of canine eye diseases**: We developed and published the *DogEyeSeg4* dataset, which includes a diverse set of real-world clinical images of canine eyes. This dataset reflects variations in lighting, breed, and image quality, providing a robust foundation for training and evaluating machine learning models in unconstrained environments.
2. **Deep convolutional neural network model for recognition of canine eye clinical symptoms and diseases from still images in unconstrained environments**: We successfully implemented and tested convolutional neural network models that accurately detect and recognize canine ocular symptoms and diseases in images captured under real-world conditions. The models demonstrated high accuracy in localizing the eye region and identifying key ocular symptoms, even in instances such as variable lighting and patient movement.
3. **Deep neural network based on U-Net for segmentation of canine eye clinical symptoms from still images in unconstrained environments**: By enhancing the standard U-Net architecture with advanced backbone networks like ResNet34 and employing combined loss functions (Dice and Focal Loss), along with augmented input, we developed

the U-Net+ResNet34(RSD) model. This model achieved superior performance in segmenting key ocular symptoms, providing precise delineation necessary for accurate diagnosis.

4. **An improved method for segmentation of canine eye conditions based on U-Net**: We introduced modifications to the U-Net network's input and architecture, including synthesized images and the use of a custom attention mechanism in the form of heatmaps generated from SSD eye detection. These enhancements led to improved segmentation accuracy and reduced false positives.

Along with these major contributions, an additional achieved contribution is:

- **The development of a comprehensive diagnostic pipeline integrating image segmentation and diagnostic interpretation.** We developed a diagnostic pipeline that combines image segmentation using U-Net+ResNet34(RSD) with diagnostic interpretation by GPT-4o. This pipeline effectively merges computer vision and natural language processing techniques, demonstrating the potential of AI-driven tools to enhance veterinary ophthalmology. The integration of precise segmentation with advanced language models improved diagnostic accuracy, laying the groundwork for robust models that can operate effectively in real-world veterinary settings

In summary, this research demonstrates the potential of integrating advanced computer vision techniques with deep learning models to enhance the diagnosis of canine ocular diseases. The development of the *DogEyeSeg4* dataset and the improved U-Net-based segmentation methods contribute valuable resources and methodologies for future research in veterinary ophthalmology. The successful application of models like U-Net+ResNet34(RSD) and GPT-4o showcases the feasibility of deploying such technologies in real-world clinical settings, ultimately aiming to improve diagnostic accuracy and patient outcomes in veterinary medicine.

Despite these advancements, the research identified limitations such as the models' sensitivity to image quality and difficulties in diagnosing subtle conditions like corneal ulcers. Addressing these challenges is significant for developing more accessible and reliable diagnostic tools. Future research should focus on expanding the dataset, refining model architectures, and integrating multimodal data to further enhance automated veterinary diagnostics and develop more comprehensive approaches to animal health care.

Bibliography

[1] M. Ashtari-Majlan, M. M. Dehshibi, and D. Masip, “Deep learning and computer vision for glaucoma detection: A review,” 2023. [Online]. Available: https://arxiv.org/abs/2307.16528

[2] ——, “Glaucoma diagnosis in the era of deep learning: A survey,” *Expert Systems with Applications*, vol. 256, pp. 1–19, 08 2024.

[3] A. C. Thompson, A. A. Jammal, and F. A. Medeiros, “A Review of Deep Learning for Screening, Diagnosis, and Detection of Glaucoma Progression,” *Translational Vision Science and Technology*, vol. 9, no. 2, pp. 42–42, 07 2020. [Online]. Available: https://doi.org/10.1167/tvst.9.2.42

[4] R. Chaudhary and A. Kumar, “Cataract detection using deep learning model on digital camera images,” in *2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom)*, 2022, pp. 489–493.

[5] T. Ren, S. Liu, A. Zeng, J. Lin, K. Li, H. Cao, J. Chen, X. Huang, Y. Chen, F. Yan, Z. Zeng, H. Zhang, F. Li, J. Yang, H. Li, Q. Jiang, and L. Zhang, “Grounded sam: Assembling open-world models for diverse visual tasks,” 2024. [Online]. Available: https://arxiv.org/abs/2401.14159

[6] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu, and L. Zhang, “Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection,” mar 20 2023, arXiv:2303.05499 [cs]. [Online]. Available: http://arxiv.org/abs/2303.05499

[7] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, “Segment Anything,” apr 5 2023, arXiv:2304.02643 [cs]. [Online]. Available: http://arxiv.org/abs/2304.02643

[8] F. Abdullah, R. Imtiaz, H. A. Madni, H. A. Khan, T. M. Khan, M. A. U. Khan, and S. S. Naqvi, “A Review on Glaucoma Disease Detection Using Computerized Techniques,” *IEEE Access*, vol. 9, pp. 37 311–37 333, 2021.

[9] Y. Hagiwara, J. E. W. Koh, J. H. Tan, S. V. Bhandary, A. Laude, E. J. Ciaccio, L. Tong, and U. R. Acharya, “Computer-aided diagnosis of glaucoma using fundus images: A review,” *Computer Methods and Programs in Biomedicine*, vol. 165, pp. 1–12, 10 2018.

[10] M. S. Haleem, L. Han, J. van Hemert, and B. Li, “Automatic extraction of retinal features from colour retinal images for glaucoma diagnosis: A review,” *Computerized Medical Imaging and Graphics*, vol. 37, no. 7-8, pp. 581–596, 10 2013.

[11] L. Li, M. Xu, X. Wang, L. Jiang, and H. Liu, “Attention Based Glaucoma Detection: A Large-Scale Database and CNN Model,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Long Beach, CA, USA: IEEE, 6 2019, pp. 10 563–10 572, [Online; accessed 2022-11-22]. [Online]. Available: https://ieeexplore.ieee.org/document/8953932/

[12] N. Chakrabarty and S. Chatterjee, “A Novel Approach to Glaucoma Screening using Computer Vision,” in *2019 International Conference on Smart Systems and Inventive Technology (ICSSIT)*. Tirunelveli, India: IEEE, 11 2019, pp. 881–884, [Online; accessed 2022-11-22]. [Online]. Available: https://ieeexplore.ieee.org/document/8987803/

[13] Zhuo Zhang, Feng Shou Yin, Jiang Liu, Wing Kee Wong, Ngan Meng Tan, Beng Hai Lee, Jun Cheng, and Tien Yin Wong, “Origa$^{\text{-light}}$: An online retinal fundus image database for glaucoma analysis and research,” in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*. Buenos Aires: IEEE, 8 2010, pp. 3065–3068, [Online; accessed 2022-11-22]. [Online]. Available: http://ieeexplore.ieee.org/document/5626137/

[14] J. Sivaswamy, S. R. Krishnadas, G. Datt Joshi, M. Jain, and A. U. Syed Tabish, “Drishti-GS: Retinal image dataset for optic nerve head(ONH) segmentation,” in *2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)*, 4 2014, pp. 53–56, iSSN: 1945-8452.

[15] S. Kanse and D. Yadav, “Retinal Fundus Image for Glaucoma Detection: A Review and Study,” *Journal of Intelligent Systems*, vol. 28, pp. 43–56, jan 28 2019.

[16] F. Fumero, S. Alayón, J. Sanchez, J. Sigut, and M. Gonzalez-Hernandez, “Rim-ONE: An open retinal image database for optic nerve evaluation,” jul 30 2011, pp. 1–6.

[17] B. Luo, J. Shen, Y. Wang, and M. Pantic, “The iBUG Eye Segmentation Dataset,” p. 9 pages, 2019, dimensions: 9 pages medium: application/pdf publisher: Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik GmbH, Wadern/Saarbruecken, Germany version: 1.0.

[18] S. J. Garbin, Y. Shen, I. Schuetz, R. Cavin, G. Hughes, and S. S. Talathi, “Openeds: Open Eye Dataset,” may 17 2019, arXiv:1905.03702 [cs, eess, stat]. [Online]. Available: http://arxiv.org/abs/1905.03702

[19] H. Proenca, S. Filipe, R. Santos, J. Oliveira, and L. A. Alexandre, “The UBIRIS.v2: A Database of Visible Wavelength Iris Images Captured On-the-Move and At-a-Distance,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1529–1535, 8 2010.

[20] W. Fuhl, G. Kasneci, and E. Kasneci, “Teyed: Over 20 million real-world eye images with Pupil, Eyelid, and Iris 2d and 3d Segmentations, 2d and 3d Landmarks, 3d Eyeball, Gaze Vector, and Eye Movement Types,” in *2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 10 2021, pp. 367–375, arXiv:2102.02115 [cs, eess]. [Online]. Available: http://arxiv.org/abs/2102.02115

[21] A. R. Strom, M. Hässig, T. M. Iburg, and B. M. Spiess, “Epidemiology of canine glaucoma presented to University of Zurich from 1995 to 2009. Part 1: Congenital and primary glaucoma (4 and 123 cases),” *Veterinary Ophthalmology*, vol. 14, no. 2, pp. 121–126, 2011, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1463-5224.2010.00855.x.

[22] D. A. J. Johnsen, D. J. Maggs, and P. H. Kass, “Evaluation of risk factors for development of secondary glaucoma in dogs: 156 cases (1999–2004),” *Journal of the American Veterinary Medical Association*, vol. 229, no. 8, pp. 1270 – 1274, 2006, publisher-place: Schaumburg, IL, USA publisher: American Veterinary Medical Association.

[23] M. Boevé and F. Stades, “[Glaucoma in dogs and cats. Review and retrospective evaluation of 421 patients. I. Pathobiological background, classification and breed predisposition],” *Tijdschrift voor diergeneeskunde*, vol. 110, no. 6, pp. 219—227, 3 1985.

[24] J. Y. Kim, H. E. Lee, Y. H. Choi, S. J. Lee, and J. S. Jeon, “Cnn-based diagnosis models for canine ulcerative keratitis,” *Scientific Reports*, vol. 9, no. 1, p. 14209, oct 2 2019, number: 1 publisher: Nature Publishing Group.

[25] M. G. Nam and S.-Y. Dong, “Classification of companion animals’ ocular diseases: Domain adversarial learning for imbalanced data,” *IEEE Access*, vol. 11, pp. 143 948–143 955, 2023.

[26] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation,” may 18 2015, arXiv:1505.04597 [cs]. [Online]. Available: http://arxiv.org/abs/1505.04597

[27] O. Petit, N. Thome, C. Rambour, and L. Soler, “U-Net Transformer: Self and Cross Attention for Medical Image Segmentation,” 2021, publisher: [object Object] version: 2. [Online]. Available: https://arxiv.org/abs/2103.06104

[28] R. Azad, E. K. Aghdam, A. Rauland, Y. Jia, A. H. Avval, A. Bozorgpour, S. Karimijafarbigloo, J. P. Cohen, E. Adeli, and D. Merhof, “Medical Image Segmentation Review: The success of U-Net,” 2022, publisher: [object Object] version: 1. [Online]. Available: https://arxiv.org/abs/2211.14830

[29] V. Badrinarayanan, A. Kendall, and R. Cipolla, “Segnet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation,” oct 10 2016, arXiv:1511.00561 [cs]. [Online]. Available: http://arxiv.org/abs/1511.00561

[30] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, *Encoder-Decoder with*

Atrous Separable Convolution for Semantic Image Segmentation. Cham: Springer International Publishing, 2018, vol. 11211, pp. 833–851, collection-title: Lecture Notes in Computer Science DOI: 10.1007/978-3-030-01234-2_49. [Online]. Available: https://link.springer.com/10.1007/978-3-030-01234-2_49

[31] M. Pirhala, J. Goga, V. Kurilova, J. Pavlovicova, and S. Kajan, *Segmentation of Significant Regions in Retinal Images: Perspective of U-Net Network Through a Comparative Approach*. Cham: Springer International Publishing, 2022, vol. 1527, pp. 29–40, collection-title: Communications in Computer and Information Science DOI: 10.1007/978-3-030-96878-6_3. [Online]. Available: https://link.springer.com/10.1007/978-3-030-96878-6_3

[32] P. K. Karn and W. H. Abdulla, "Advancing Ocular Imaging: A Hybrid Attention Mechanism-Based U-Net Model for Precise Segmentation of Sub-Retinal Layers in OCT Images," *Bioengineering*, vol. 11, no. 3, p. 240, feb 28 2024.

[33] N. Siddique, S. Paheding, C. P. Elkin, and V. Devabhaktuni, "U-Net and Its Variants for Medical Image Segmentation: A Review of Theory and Applications," *IEEE Access*, vol. 9, pp. 82 031–82 057, 2021.

[34] 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," 2021, version: 2. [Online]. Available: https://arxiv.org/abs/2103.14030

[35] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "Segformer: Simple and Efficient Design for Semantic Segmentation with Transformers," 2021, version: 3. [Online]. Available: https://arxiv.org/abs/2105.15203

[36] X. Jia, J. Bartlett, T. Zhang, W. Lu, Z. Qiu, and J. Duan, *U-Net vs Transformer: Is U-Net Outdated in Medical Image Registration?* Cham: Springer Nature Switzerland, 2022, vol. 13583, pp. 151–160, collection-title: Lecture Notes in Computer Science DOI: 10.1007/978-3-031-21014-3_16. [Online]. Available: https://link.springer.com/10.1007/978-3-031-21014-3_16

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," dec 10 2015, arXiv:1512.03385 [cs] version: 1. [Online]. Available: http://arxiv.org/abs/1512.03385

[38] M. Tan and Q. V. Le, "Efficientnet: Rethinking Model Scaling for Convolutional Neural Networks," sep 11 2020, arXiv:1905.11946 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1905.11946

[39] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," apr 10 2015, arXiv:1409.1556 [cs]. [Online]. Available: http://arxiv.org/abs/1409.1556

[40] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception

Architecture for Computer Vision," dec 11 2015, arXiv:1512.00567 [cs] version: 3. [Online]. Available: http://arxiv.org/abs/1512.00567

[41] T. Nemoto, N. Futakami, E. Kunieda, M. Yagi, A. Takeda, T. Akiba, E. Mutu, and N. Shigematsu, "Effects of sample size and data augmentation on U-Net-based automatic segmentation of various organs," *Radiological Physics and Technology*, vol. 14, no. 3, pp. 318–327, 9 2021.

[42] M. Buric, B. Kovacic, and M. Ivasic-Kos, "The Dog Eye Guardian App: From Image to Diagnosis with AI Insights," in *2024 9th International Conference on Smart and Sustainable Technologies (SpliTech)*. Bol and Split, Croatia: IEEE, jun 25 2024, pp. 1–6, [Online; accessed 2024-08-23]. [Online]. Available: https://ieeexplore.ieee.org/document/10612583/

[43] A. Dandekar, R. A. M. Zen, and S. Bressan, "A Comparative Study of Synthetic Dataset Generation Techniques," in *Database and Expert Systems Applications*, ser. Lecture Notes in Computer Science, S. Hartmann, H. Ma, A. Hameurlain, G. Pernul, and R. R. Wagner, Eds. Cham: Springer International Publishing, 2018, pp. 387–395.

[44] L. Yang, Z. Zhang, Y. Song, S. Hong, R. Xu, Y. Zhao, W. Zhang, B. Cui, and M.-H. Yang, "Diffusion Models: A Comprehensive Survey of Methods and Applications," jun 23 2024, arXiv:2209.00796 [cs]. [Online]. Available: http://arxiv.org/abs/2209.00796

[45] R. Po, W. Yifan, V. Golyanik, K. Aberman, J. T. Barron, A. H. Bermano, E. R. Chan, T. Dekel, A. Holynski, A. Kanazawa, C. K. Liu, L. Liu, B. Mildenhall, M. Nießner, B. Ommer, C. Theobalt, P. Wonka, and G. Wetzstein, "State of the Art on Diffusion Models for Visual Computing," oct 11 2023, arXiv:2310.07204 [cs]. [Online]. Available: http://arxiv.org/abs/2310.07204

[46] L. Zhang, A. Rao, and M. Agrawala, "Adding Conditional Control to Text-to-Image Diffusion Models," nov 26 2023, arXiv:2302.05543 [cs]. [Online]. Available: http://arxiv.org/abs/2302.05543

[47] N. Carlini, J. Hayes, M. Nasr, M. Jagielski, V. Sehwag, F. Tramèr, B. Balle, D. Ippolito, and E. Wallace, "Extracting Training Data from Diffusion Models," 2023, version: 1. [Online]. Available: https://arxiv.org/abs/2301.13188

[48] A. Stöckl, "Evaluating a Synthetic Image Dataset Generated with Stable Diffusion," 2022, version: 2. [Online]. Available: https://arxiv.org/abs/2211.01777

[49] C. H. Savage, H. Park, K. Kwak, A. D. Smith, S. A. Rothenberg, V. S. Parekh, F. X. Doo, and P. H. Yi, "General-Purpose Large Language Models Versus a Domain-Specific Natural Language Processing Tool for Label Extraction From Chest Radiograph Reports," *American Journal of Roentgenology*, p. AJR.23.30573, jan 17 2024.

[50] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux,

P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed, "Mistral 7b," 2023, publisher: arXiv version: 1. [Online]. Available: https://arxiv.org/abs/2310.06825

[51] K. Saab, T. Tu, and W.-H. Weng, "Capabilities of Gemini Models in Medicine," 2024, publisher: [object Object] version: 2. [Online]. Available: https://arxiv.org/abs/2404.18416

[52] W. Huang, X. Ma, H. Qin, X. Zheng, C. Lv, H. Chen, J. Luo, X. Qi, X. Liu, and M. Magno, "How Good Are Low-bit Quantized LLaMA3 Models? An Empirical Study," 2024, publisher: [object Object] version: 1. [Online]. Available: https://arxiv.org/abs/2404.14047

[53] T. Woelfle, J. Hirt, P. Janiaud, L. Kappos, J. P. A. Ioannidis, and L. G. Hemkens, "Benchmarking Human-AI Collaboration for Common Evidence Appraisal Tools," apr 22 2024, [Online; accessed 2024-05-14]. [Online]. Available: http://medrxiv.org/lookup/doi/10.1101/2024.04.21.24306137

[54] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "Bertscore: Evaluating Text Generation with BERT," 2019, publisher: [object Object] version: 3. [Online]. Available: https://arxiv.org/abs/1904.09675

[55] J. Hessel, A. Holtzman, M. Forbes, R. Le Bras, and Y. Choi, "Clipscore: A Reference-free Evaluation Metric for Image Captioning," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, 2021, pp. 7514–7528, [Online; accessed 2024-05-14]. [Online]. Available: https://aclanthology.org/2021.emnlp-main.595

[56] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL '02*. Philadelphia, Pennsylvania: Association for Computational Linguistics, 2001, p. 311, [Online; accessed 2024-05-14]. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1073083.1073135

[57] M. Denkowski and A. Lavie, "Meteor Universal: Language Specific Translation Evaluation for Any Target Language," in *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Baltimore, Maryland, USA: Association for Computational Linguistics, 2014, pp. 376–380, [Online; accessed 2024-05-14]. [Online]. Available: http://aclweb.org/anthology/W14-3348

[58] K. Ganesan, "Rouge 2.0: Updated and Improved Measures for Evaluation of Summarization Tasks," 2018, publisher: [object Object] version: 1. [Online]. Available: https://arxiv.org/abs/1803.01937

[59] P. Anderson, B. Fernando, M. Johnson, and S. Gould, "Spice: Semantic Propositional Image Caption Evaluation," 2016, publisher: [object Object] version: 1. [Online].

Available: https://arxiv.org/abs/1607.08822

[60] M. Burić, M. Ivašić-Kos, and S. Grozdanić, “Dogeyeseg4: Dog Eye Segmentation 4-Class Ophthalmic Disease Dataset,” [Online; accessed 2024-08-22]. [Online]. Available: https://urn.nsk.hr/urn:nbn:hr:195:405214

[61] S. Grozdanić, S. Đukić, S. Luzhetskiy, N. Milčić-Matić, and T. Lazić, *Atlas bolesti oka pasa i mačaka*. Beograd: Oculus Vet, 2020.

[62] M. Buric, S. Grozdanic, and M. Ivasic-Kos, “Diagnosis of ophthalmologic diseases in canines based on images using neural networks for image segmentation,” *Heliyon*, p. e38287, 9 2024.

[63] *The EU’s General Data Protection Regulation (GDPR) in a Research Context*. Cham: Springer International Publishing, 2019, pp. 55–71, dOI: 10.1007/978-3-319-99713-1_5. [Online]. Available: http://link.springer.com/10.1007/978-3-319-99713-1_5

[64] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative Adversarial Networks,” 2014, version: 1. [Online]. Available: https://arxiv.org/abs/1406.2661

[65] D. P. Kingma and M. Welling, “Auto-Encoding Variational Bayes,” 2013, version: 11. [Online]. Available: https://arxiv.org/abs/1312.6114

[66] J. Ho, A. Jain, and P. Abbeel, “Denoising Diffusion Probabilistic Models,” 2020, version: 2. [Online]. Available: https://arxiv.org/abs/2006.11239

[67] G. Paulin and M. IvasicKos, “Review and analysis of synthetic dataset generation methods and techniques for application in computer vision,” *Artificial Intelligence Review*, vol. 56, no. 9, pp. 9221–9265, 9 2023.

[68] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-Resolution Image Synthesis with Latent Diffusion Models,” in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. New Orleans, LA, USA: IEEE, 6 2022, [Online; accessed 2024-09-17]. [Online]. Available: https://ieeexplore.ieee.org/document/9878449/

[69] J. Thompson, H. Dong, K. Liu, F. He, M. Popescu, and D. Xu, *A Rule-Based Approach for Generating Synthetic Biological Pathways*. Cham: Springer International Publishing, 2022, vol. 13483, pp. 105–116, collection-title: Lecture Notes in Computer Science DOI: 10.1007/978-3-031-20837-9_9. [Online]. Available: https://link.springer.com/10.1007/978-3-031-20837-9_9

[70] S. Lee, B. Hoover, H. Strobelt, Z. J. Wang, S. Peng, A. Wright, K. Li, H. Park, H. Yang, and D. H. Chau, “Diffusion explainer: Visual explanation for text-to-image stable diffusion,” 2024. [Online]. Available: https://arxiv.org/abs/2305.03509

[71] M. Fries1111, “Wildlifex animals model,” 2024, accessed: 2024-10-28. [Online]. Available: https://civitai.com/models/262239/wildlifex-animals

[72] S. 161222, “Realistic vision v6.0 b1 hypervae model,” 2024, accessed: 2024-10-28. [Online]. Available: https://civitai.com/models/4201/realistic-vision-v60-b1

[73] Civitai Community, “Civitai generative ai,” 2024, accessed: 2024-10-28. [Online]. Available: https://civitai.com

[74] M. Buric and M. Ivasic-Kos, “Next-Generation Computer Vision in Veterinary Medicine: A Study on Canine Ophthalmology,” jul 22 2024, [Online; accessed 2024-09-27]. [Online]. Available: https://www.techrxiv.org/users/802988/articles/1188856-next-generation-computer-vision-in-veterinary-medicine-a-study-on-canine-ophthalmology?commit=f4bdfc66ea37f9905cc12704c501b3c4848289f1

[75] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, “Lora: Low-Rank Adaptation of Large Language Models,” *arXiv preprint arXiv:2106.09685*, 2023. [Online]. Available: https://openreview.net/forum?id=nZeVKeeFYf9

[76] F. Zhang and M. Pilanci, “Riemannian Preconditioned LoRA for Fine-Tuning Foundation Models,” *arXiv preprint arXiv:2403.19776*, 2024. [Online]. Available: https://arxiv.org/abs/2403.19776

[77] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, “Image inpainting,” in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques - SIGGRAPH ’00*. Not Known: ACM Press, 2000, pp. 417–424, [Online; accessed 2024-09-17]. [Online]. Available: http://portal.acm.org/citation.cfm?doid=344779.344972

[78] J. Song, C. Meng, and S. Ermon, “Denoising Diffusion Implicit Models,” in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=St1giarCHLP

[79] B. L. Kidder, “Advanced image generation for cancer using diffusion models,” *Biology Methods and Protocols*, vol. 9, no. 1, p. bpae062, 08 2024. [Online]. Available: https://doi.org/10.1093/biomethods/bpae062

[80] M. Buric, M. Ivasic-Kos, and S. Martinčić-Ipšić, “The Disease of the Canine Eye - From Image to Diagnosis Using AI,” *WSCG International Conferences in Central Europe on Computer Graphics, Visualization and Computer Vision*, jun 3 2024, prague/Pilsen, Czech Republic.

[81] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, “3d U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation,” 2016, version: 1. [Online]. Available: https://arxiv.org/abs/1606.06650

[82] S. Jegou, M. Drozdzal, D. Vazquez, A. Romero, and Y. Bengio, “The One Hundred Layers Tiramisu: Fully Convolutional DenseNets for Semantic Segmentation,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Honolulu, HI, USA: IEEE, 7 2017, pp. 1175–1183, [Online; accessed 2024-09-19].

[Online]. Available: http://ieeexplore.ieee.org/document/8014890/

[83] V. Dumoulin and F. Visin, “A guide to convolution arithmetic for deep learning,” 2016, version: 2. [Online]. Available: https://arxiv.org/abs/1603.07285

[84] J. Deng, W. Dong, R. Socher, L.-J. Li, Kai Li, and Li Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. Miami, FL: IEEE, 6 2009, pp. 248–255, [Online; accessed 2022-11-04]. [Online]. Available: https://ieeexplore.ieee.org/document/5206848/

[85] F. Ramzan, M. U. G. Khan, A. Rehmat, S. Iqbal, T. Saba, A. Rehman, and Z. Mehmood, “A Deep Learning Approach for Automated Diagnosis and Multi-Class Classification of Alzheimer’s Disease Stages Using Resting-State fMRI and Residual Neural Networks,” *Journal of Medical Systems*, vol. 44, no. 2, p. 37, 2 2020.

[86] M. S. Islam, F. A. Foysal, N. Neehal, E. Karim, and S. A. Hossain, “Inceptb: A CNN Based Classification Approach for Recognizing Traditional Bengali Games,” *Procedia Computer Science*, vol. 143, pp. 595–602, 2018.

[87] M. Hakim BENDIABDALLAH and N. SETTOUTI, “A comparison of u-net backbone architectures for the automatic white blood cells segmentation,” *WAS Science Nature (WASSN) ISSN: 2766-7715*, vol. 4, no. 1, Sep. 2021. [Online]. Available: https://worldascience.com/journals/index.php/wassn/article/view/24

[88] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” *Proceedings of the 27th International Conference on International Conference on Machine Learning*, p. 807–814, 2010.

[89] A. L. Maas, A. Y. Hannun, A. Y. Ng, and others, “Rectifier nonlinearities improve neural network acoustic models,” in *Proc. icml*, vol. 30. Atlanta, GA, 2013, p. 3, issue: 1.

[90] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *International Conference on Artificial Intelligence and Statistics*, 2010. [Online]. Available: https://api.semanticscholar.org/CorpusID:5575601

[91] C. M. Bishop, *Pattern recognition and machine learning*, ser. Information science and statistics. New York: Springer, 2006.

[92] M. Buric and M. Ivasic-Kos, “Ai in Veterinary Ophthalmology: Canine Eye Diseases Diagnosis with Image Segmentation and Large Language Models,” in *World Conference on Smart Trends in Systems, Security and Sustainability - WorldS4*. London, UK: In Press, jul 23 2024.

[93] D. Brin, V. Sorin, Y. Barash, E. Konen, B. S. Glicksberg, G. N. Nadkarni, and E. Klang, “Assessing gpt-4 multimodal performance in radiological image analysis,” *European Radiology*, aug 2024. [Online]. Available: https://doi.org/10.1007/s00330-024-11035-5

[94] C. Wang, X. Liu, and D. Song, “Language Models are Open Knowledge Graphs,” 2020, version: 1. [Online]. Available: https://arxiv.org/abs/2010.11967

[95] M. Hussain, “Yolo-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial Defect Detection,” *Machines*, vol. 11, no. 7, p. 677, jun 23 2023.

[96] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” jan 24 2018, arXiv:1703.06870 [cs]. [Online]. Available: http://arxiv.org/abs/1703.06870

[97] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “Ssd: Single Shot MultiBox Detector,” 2015, publisher: arXiv version: 5. [Online]. Available: https://arxiv.org/abs/1512.02325

[98] S. Hykes, “Docker, Inc.” https://www.docker.com/, jan 5 2024. [Online]. Available: https://www.docker.com/

[99] G. Van Rossum and F. L. Drake, *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009.

[100] TensorFlow Developers, “Tensorflow,” jul 11 2024, dOI: 10.5281/ZENODO.4724125. [Online]. Available: https://zenodo.org/doi/10.5281/zenodo.4724125

[101] G. Bradski, “The OpenCV Library,” *Dr. Dobb’s Journal of Software Tools*, 2000.

[102] A. Clark, “Pillow (PIL Fork) Documentation,” 2015. [Online]. Available: https://buildmedia.readthedocs.org/media/pdf/pillow/latest/pillow.pdf

[103] A. Ronacher, “Flask,” https://flask.palletsprojects.com/en/3.0.x/, aug 5 2024. [Online]. Available: https://flask.palletsprojects.com/en/3.0.x/

[104] A. Thamizharasan, M. S. Murugan, and S. Parthiban, “Surgical Management of Cherry Eye in a Dog,” *Intas Polivet*, 2016.

[105] R. M. Tripathi, D. K. Kashyap, and D. K. Giri, “Surgical Management of Cherry Eye in a Dog,” *Intas Polivet*, 2016.

[106] 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,” *CoRR*, vol. abs/2103.00020, 2021. [Online]. Available: https://arxiv.org/abs/2103.00020

[107] A. F. Siegel, *ANOVA*. Elsevier, 2016, pp. 469–492, dOI: 10.1016/B978-0-12-804250-2.00015-8. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/B9780128042502000158

[108] J. W. Tukey, “Comparing Individual Means in the Analysis of Variance,” *Biometrics*, vol. 5, no. 2, p. 99, 6 1949.

[109] A. F. M. Alkarkhi and W. A. A. Alqaraghuli, *Easy statistics for food science with R: Abbas F.M. Alkarkhi (Malaysian Institut of Chemical & Bioengineering Technology Universiti Kuala Lumpur, UniKL, MICET, 78000 Melaka, Malaysia), Wasin A.A. Alqaraghuli (Skill Education Center, PA, A-07-03 Pearl Avenue, Sungai Chua, 43000 Kajang, Selangor, Malaysia)*. London San Diego Cambridge Kidlington: Academic Press, 2019.

[110] D. C. Montgomery, *Design and analysis of experiments*, ninth edition ed. Hoboken, NJ: John Wiley & Sons, Inc, 2017.

[111] S. Shahriar, B. Lund, N. R. Mannuru, M. A. Arshad, K. Hayawi, R. V. K. Bevara, A. Mannuru, and L. Batool, “Putting gpt-4o to the sword: A comprehensive evaluation of language, vision, speech, and multimodal proficiency,” 2024. [Online]. Available: https://arxiv.org/abs/2407.09519

List of Figures

3.1. Example of images in DogEyeSeg4 dataset with corresponding masks showing closeup of an eye (upper row) and whole head (lower row) [62]. 9
3.2. visual representation of medical symptoms in the top-left to bottom-right order: S1 - cloudiness or haziness of the cornea; S2 - sclera redness; S3 - excessive tears; and a S4 - colored mass protrusion in the corner of the eye[62]. 10
3.3. Distribution of classes in the DogEyeSeg4 Dataset. The number represents the recurrence of certain ocular diseases in images. Ocular symptoms which accompany Corneal edema, Episcleral congestion, Epiphora and Cherry Eye are Cloudiness or haziness of the cornea (S1), Redness of the sclera (S2), Excessive tearing (S3) and A colored mass in the corner of the eye (S4) respectively [62]. 11
3.4. Example of applied augmentations on original image in top row: horizontal flip (middle row left), horizontal shift (middle row right), rotation (bottom row left), and vertical shift (bottom row right) [62]. 12
3.5. Stable diffusion generated images with fixed posture of different dog breeds. Images shows high degree of reality with certain difficulty like double set of ears [74]. 14
3.6. Close-up Stable Diffusion images using custom LoRA describing various medical conditions based on DogEyeSeg4 dataset [74]. 15
3.7. Stable Diffusion generated image containing symptoms using Inpainting showing (a) healthy eyes, left eye with: (b) prolapsed eyelid gland, (c) red sclera, (d) cloudy cornea, (e) epiphora and (f) all previously mentioned symptoms [74]. . . 16
3.8. The standard U-Net architecture where left encoder part and right decoder part form the letter U. The encoder effectively captures detailed features from the input images, which are then upsampled by the decoder part of U-Net. The decoder uses transposed convolutions to restore the image to its original size, using skip connections from the corresponding encoder layers to refine the segmentation output with high-resolution details retained [62]. 19
3.9. The VGG-16 architecture [62]. 20
3.10. The ResNet18 architecture [62]. 20
3.11. The Inception V3 architecture [62]. 21

3.12. Schematic of the enhanced U-Net architecture, integrating a backbone (e.g., VGG, ResNet) within both encoder and decoder blocks. The Encoder Blocks (EB) incorporate convolutional layers from the backbone, downscaling the input while extracting progressively complex features. Only the feature-extracting layers of the backbone are used, omitting fully connected layers. The Decoder Blocks (DB) then upsample these features, using skip connections from corresponding EBs to preserve high-resolution details, ultimately reconstructing the output image [62]. 21
3.13. Architecture of a Transformer Decoder block in the GPT model. The process begins with the input—text tokens with positional information, transformed into vector embeddings encoding both meaning and position. These embeddings pass through layers of multi-head attention and feed-forward networks, with normalization applied for stability. Multiple Transformer Decoder Blocks (denoted as Nx) capture contextual relationships. The model's output is a probability distribution over possible next tokens, generated by a linear layer and softmax function, enabling text generation based on the input sequence. [74]. . 28
3.14. An example of input prompt to GPT-4 and GPT-4o requesting segmentation of medical conditions and most probable diagnosis [74]. 29
3.15. Grounding Dino and SAM integration flowchart [74]. 32
3.16. Grounding Dino prompting examples showing diversity with resulting detection and its prediction score. Prompts used for generating detection from top to bottom are: (a) "colored mass protrusion in the corner of the eye", (b) "colored mass protrusion in the eye" and (c) "red mass protrusion". Third prompt (c) is used for segmentation by SAM implementation (d) [74]. 33
3.17. Single Shot Multibox Detector diagram for eye detection. 34
3.18. A web-based application flowchart for eye symptoms segmentation and medical condition prediction. 35
3.19. An overlay mask acting as capturing images guide for accurate segmentation [42]. 36

4.1. Images of an American bulldog (left) and a crossbred dog (right) show a pink-colored mass protruding from the medial canthus of both eyes, accompanied by signs of inflammation, including pain and excessive tearing, indicative of Cherry Eye [104, 105]. 38
4.2. The test dataset contains 25 samples, evenly distributed across five different diseases. Symptom 1 represents a colored mass protruding in the corner of the eye, Symptom 2 refers to cloudiness or haziness in the cornea, Symptom 3 signifies redness of the sclera, and Symptom 4 denotes excessive tearing. 40

4.3. Representation of confusion matrix with horizontal argument as Ground Truth (GT) and vertical as Prediction showing correctly identified regions as TP, incorrectly identified regions as FP, missed regions as FN and correctly identified negative regions as TN. 42
4.4. The workflow diagram for analyzing canine ophthalmic conditions demonstrates the process of evaluating an image of a dog's eye. This involves utilizing model for segmentation, followed by the use of LLM (Large Language Model) for interpreting the medical symptoms. The outcome is a segmented image alongside a comprehensive diagnostic report. 45
4.5. Average training loss (upper) and IOU (lower) values for 100 epochs showing expecting curve of well performing model training [62]. 47
4.6. Flowchart describing the results of the first experiment where U-Net with ResNet34 backbone and combination of Dice and Focal loss provide best performance model when accuracy and time is taken into consideration for further research. . 49
4.7. Visual representation of 5-fold cross-validation where the dataset folds in ratio 80:20 providing 5 different models which are validated on unseen part of the dataset [62]. 50
4.8. Display of a QQ-plot which evaluates the standardized residuals of a model against a theoretical normal distribution. The data points closely follow the expected normal line, exhibiting only slight deviations at the ends, suggesting that the residuals generally adhere to a normal distribution [62]. 52
4.9. Examples of the segmentation analysis of close-up eye images with diagnostic labels. The original images are shown in the left column, the middle column represents the ground truth (GT), and the predictions are displayed in the right column. In the segmentation results, the top row correctly identifies two classes according to the ground truth, the middle row misses the detection of the excessive tearing class, and the bottom row incorrectly identifies an additional instance of redness of the sclera. 56
4.10. Segmentation of test images featuring whole head figures with diagnoses is depicted. The left column shows the original images, the center column displays the ground truth (GT), and the right column illustrates the predictions. In the top row, a colored mass in the corner of the eye is correctly predicted while redness of the sclera is mistakenly identified. The middle row incorrectly detects cloudiness or haziness of the cornea but successfully segments a colored mass in the corner of the eye. The bottom row falsely identifies a combination of redness of the sclera and cloudiness or haziness of the cornea where only a colored mass in the corner of the eye is actually present. 57

4.11. The training dataset used to develop the U-Net+ResNet34(R) model consists exclusively of real images, while the U-Net+ResNet34(RSD) model incorporates a combination of both real and synthetic images along with their corresponding masks. 58

4.12. Dice Index of four different segmentation methods for individual classes. The U-Net+ResNet34(RSD) shows best results in three classes and closely follows result of U-Net+ResNet34(R) when forth class is observed. 60

4.13. Example of binary segmentation without (2nd row) and with (3rd row) heatmap as 4 channel input. 62

4.14. Example of binary segmentation on whole head without (2nd row) and with (3rd row) heatmap as 4 channel input 62

4.15. Binary segmentation example showing that focusing too narrowly on the eye region can result in false negatives (3rd row) compared to partly segmented excessive tearing (2nd row). 63

4.16. Example where lack of detection reflects in absence of segmentation of a right eye (3rd row) but increases FPs if not applied at all (2nd row). 63

4.17. True positive diagnoses by each LLM across 15 cases, comparing single and dual diagnoses. 70

4.18. Mean dual diagnosis BERTScore across all LLMs evaluated for each symptom combination. 70

4.19. An analysis of the differences between results provided using single and dual diagnoses, using CLIPScore metric, evaluated on GPT-4 across various symptom combinations. 71

4.20. An Evaluation of the Diagnostic Differences Between GPT-4 and GPT-4o in five Canine Eye Diseases showing performance in favor of GPT-4o in four out of five diseases. Uveitis was diagnosed by GPT-4o with 100% prediction rate in single and dual diagnosis and with and without provided U-Net+ResNet34(RSD) segmentation masks. 73

4.21. BERTScore results for each medical case, categorized into four disease groups, are presented based on GPT-4o when both single and dual diagnosis approaches are applied. 73

5.1. The depicted diagnostic pipeline for canine ophthalmic conditions demonstrates improved performance when both the original images and their corresponding segmented masks from U-Net+ResNet34(RSD) are utilized as inputs for GPT-4o, enhancing the accuracy of the medical diagnoses derived from visual symptoms. 76

List of Tables

4.1. Segmentation models evaluation results in terms of Jaccard coefficient (IoU) percentages. Training time (t^*) needed for training on full dataset using NVIDIA GeForce RTX 2060 GPU with 12GB RAM and average test time per image (t^{**}) on Intel(R) Xeon(R) CPU E5-1603 v4 @ 2.80GHz. 48

4.2. 5-fold cross-validation results using mean IoU in percentage with background class included. 51

4.3. Comparison of models based on the Jaccard Index using the Tukey HSD test. P-values are provided for each pairwise comparison. 53

4.4. 5-fold cross-validation results of U-Net+ResNet34 models based on IoU, Dice Index (DSC), Sensitivity, and Pixel Accuracy. The best results are marked in bold, and average in *italic*. 54

4.5. Confusion matrix for U-Net+ResNet34 Cross-Validation Model 2 (pixels presented in %). 55

4.6. Segmentation evaluation of U-Net+ResNet34(R), U-Net+ResNet34(RSD), GPT-4o, and Grounding SAM methods using IoU, Dice Index (DSC), and Pixel Accuracy. 59

4.7. Evaluation of U-Net+ResNet34(RSD) using IoU and Dice Index with standard deviation, along with Pixel Accuracy, Sensitivity, and Specificity on individual symptoms. 59

4.8. Evaluation of binary U-Net+ResNet34(R) models not considering class overlapping. Results are generally lower compared to multiclass U-Net+ResNet34(R) with significant decrease in sensitivity. 61

4.9. Evaluation of binary U-Net+ResNet34(R) models with Heatmaps. Heatmap includes three specific white spots: one centered in the eye and two at the lower corners of the bounding box around the eye, identified using an SSD object detector. 61

4.10. An illustrative example comparing the results generated by the Mistral LLM when prompted by a dog owner without specifying any symptoms with the outputs from U-Net image segmentation that utilized symptom information akin to that provided by a veterinarian. Glaucoma is the correct diagnosis. 65

4.11. Example of prompts given to five LLM models, along with their responses and associated score values. The correct diagnosis in this case is Cherry Eye. 66
4.12. Comparison of LLMs for Diagnostic Evaluation of a Specific Disease Using Symptom Criteria individually and in combination. 67
4.13. Comparison of LLM Responses for Dual Diagnostic Evaluation 68
4.14. Evaluation of Large Language Models (LLMs) using various text similarity metrics. 69
4.15. Evaluation of GPT-4 and GPT-4o in single and dual medical diagnoses, with and without masks generated by U-Net+ResNet34(RSD) segmentation. 72

List of Published Research Papers in Computer Vision

1. Burić, M., Grozdanić, S., & Ivašić-Kos, M. (2024). *Diagnosis of ophthalmologic diseases in canines based on images using neural networks for image segmentation. Heliyon*, **10**, e38287. `https://doi.org/10.1016/j.heliyon.2024.e38287`
2. Burić, M., & Ivašić-Kos, M. (2024). *AI in veterinary ophthalmology: Canine eye diseases diagnosis with image segmentation and large language models*. In *Proceedings of the World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*. London, UK. [In Press].
3. Burić, M., & Ivašić-Kos, M. (2024). *DogEyeSeg4: Dog eye segmentation 4-class ophthalmic disease dataset*. `https://urn.nsk.hr/urn:nbn:hr:195:405214`, `https://doi.org/10.13140/RG.2.2.28029.58084` [Data set].
4. Burić, M., & Ivašić-Kos, M. (2024). *Next-generation computer vision in veterinary medicine: A study on canine ophthalmology*. *TechRxiv*. `https://doi.org/10.36227/techrxiv.172166061.12686289/v1`
5. Burić, M., & Ivašić-Kos, M. (2024). *The disease of the canine eye – From image to diagnosis using AI*. In *Proceedings of the 32nd International Conference on Computer Graphics, Visualization and Computer Vision (WSCG 2024)*. Pilsen, Czech Republic. `https://doi.org/10.24132/CSRN.3401.41`
6. Burić, M., Kovačić, B., & Ivašić-Kos, M. (2024). *The Dog Eye Guardian App: From image to diagnosis with AI insights*. In *Proceedings of the 9th International Conference on Smart and Sustainable Technologies (SpliTech24)* (pp. 1–6). IEEE, Bol, Croatia.
7. Burić, M., Ivašić-Kos, M., & Paulin, G. (2020). *Object Detection Using Synthesized Data*. In *Proceedings of the ICT Innovations Conference*. Ohrid, North Macedonia.
8. Burić, M., Pobar, M., & Ivašić-Kos, M. (2019). *Adapting YOLO Network for Ball and Player Detection*. In *Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods* (pp. 845–851). Prague, Czech Republic. `https://doi.org/10.5220/0007582008450851`
9. Burić, M., Ivašić-Kos, M., & Pobar, M. (2019). *Player Tracking in Sports Videos*. In *Proceedings of the 2019 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)* (pp. 334–340). IEEE, Sydney, NSW, Australia. `https://`

doi.org/10.1109/CloudCom.2019.00058

10. Burić, M., Pobar, M., & Ivašić-Kos, M. (2018). *Ball Detection Using YOLO and Mask R-CNN*. In *Proceedings of the 2018 International Conference on Computational Science and Computational Intelligence (CSCI)* (pp. 319–323). IEEE, Las Vegas, NV, USA. `https://doi.org/10.1109/CSCI46756.2018.00068`
11. Burić, M., Pobar, M., & Ivašić-Kos, M. (2018). *Object Detection in Sports Videos*. In *Proceedings of the 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (pp. 1034–1039). IEEE, Opatija, Croatia. `https://doi.org/10.23919/MIPRO.2018.8400189`
12. Burić, M., Pobar, M., & Ivašić-Kos, M. (2017). *An Overview of Action Recognition in Videos*. In *Proceedings of the 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (pp. 1098–1103). IEEE, Opatija, Croatia. `https://doi.org/10.23919/MIPRO.2017.7973588`