Machine Learning Team Project Report

Team 4

- Hyunwon Nam (202020820)
- Yonghun Jung (202020354)
- Hyeseong Kim (202020715)
- Subin Kang (202220799)
- Sebin Kwon (202220781)

Table of Contents

1. Introduction

1.1 Background:

This project aims to develop a classification model to determine the presence of cataracts in dogs. Cataracts are a major disease that impairs vision in dogs, making early detection and accurate diagnosis important. By developing this classification model, we expect to extend it into a web/app service that allows for quick detection without visiting a veterinary hospital.

1.2 Describe requirements, assumptions, risks, and constraints:

The main requirement of the project is to develop a diagnostic model with high accuracy. Especially for cataracts, identifying the early (incipient) stage is crucial for treatment, thus requiring not only accuracy but also high recall.

The assumption is that the image data provided by AI-Hub is sufficiently representative and accurately labeled.

A major risk is whether simple image data is sufficient to judge the severity of cataracts. In actual diagnosis, various tests such as slit-lamp examination and retinal examination using medical equipment are combined to make a judgment. Therefore, it is uncertain if accurate classification is possible with just simple images.

1.3 Determine project goals and success criteria:

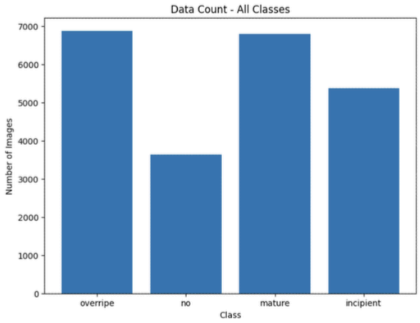
The success criteria of the project are evaluated based on the model's accuracy, F1 score, and recall. The goal is to achieve over 80% in these evaluation metrics.

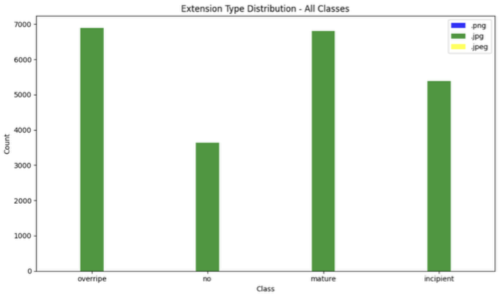
2. Data

2.1 Describe the data you use or how you collected the data:

We used the cataract image dataset of dog eyes provided by AI-Hub. The classes are {no, incipient, mature, overripe}. Each class contains 6,800 images. Upon checking the actual data, it was found that each class had between 3,800 to 6,800 images.

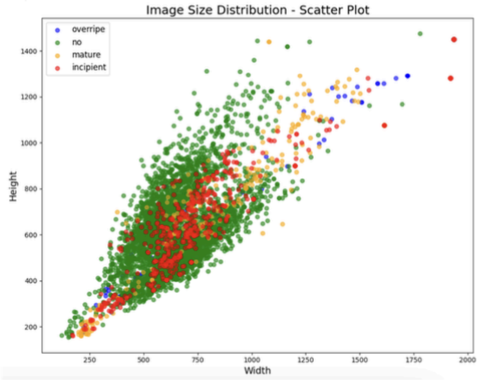
2.2 Exploratory Data Analysis (EDA):





After EDA on the actual received data, we confirmed that each class had 3,800 to 6,800 images with varying class distributions. To balance the classes and reduce memory usage during training, we used 2,000 images per class. Also, we confirmed that all images were in jpg format.

The scatter plot of raw image data sizes showed diverse image sizes per class. Subsequently, in preprocessing, we resized all images to 224 x 224.



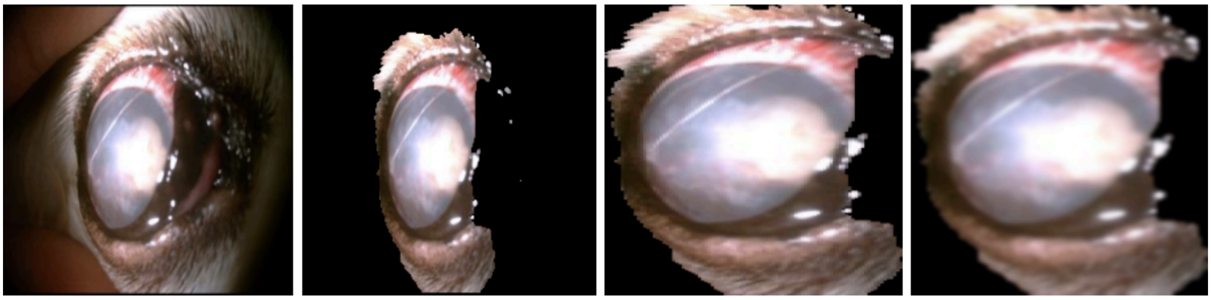
2.3 Data cleaning / pre-processing:

Data pre-processing-1: We selected 2,000 images per class, removed black backgrounds, resized the images to 224 x 224, and zero-centered the pixel values. We compared the performance based on image extension (jpg, png) and color (color, gray) resulting in four datasets {png-color, jpg-color, png-gray, jpg-gray}, which were tested with the initial classification model.

Data pre-processing-2: Initial model performance was unsatisfactory. To understand the effect of fur on model performance, we used the Canny edge algorithm to check the contours and found many fur outlines. To mitigate this, we used the grab-cut algorithm to remove fur areas, leaving only the eyeball. We then retested with four datasets {png-color, jpg-color, png-gray, jpg-gray}.

Data pre-processing-3: After using the grab-cut algorithm, black areas were left as backgrounds. To address this, we removed the black backgrounds, resized images to 224 x 224, and retested the four datasets {png-color, jpg-color, png-gray, jpg-gray}. Using the ViT model showed an improvement from 0.79 to 0.84 in accuracy.

Data pre-processing-4: Despite minimizing fur areas, some fur remained, which could still affect performance. Therefore, we applied blur to make the images slightly blurry, reducing the impact of fur. After testing with the four datasets, the blur processing showed no significant performance improvement with the ViT model.



Final Data Preprocessing: The final dataset selected used the grab-cut algorithm followed by black background removal, resized to 224 x 224, and the png-color format.

Data outlier: After applying the grab-cut algorithm, we identified images with all-black pixels in the overrip, no, and incipient classes, constituting about 0.1% of each class. These were removed to avoid negatively impacting the model. About 20 images per class were removed, maintaining dataset balance.

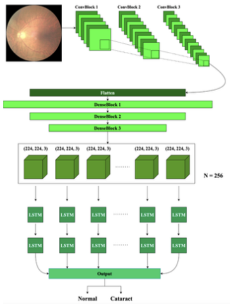
3. Method

3.1 Model / Algorithm description:

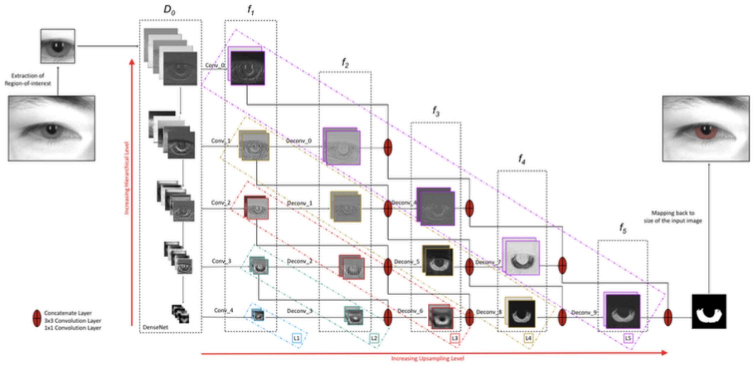
Initial Models: CNN, VGG16 (not pre-trained), ResNet50 (not pre-trained). The accuracy was CNN: 0.61, VGG16: 0.67, ResNet50: 0.51, all below target accuracy.

3.2 Model comparison:

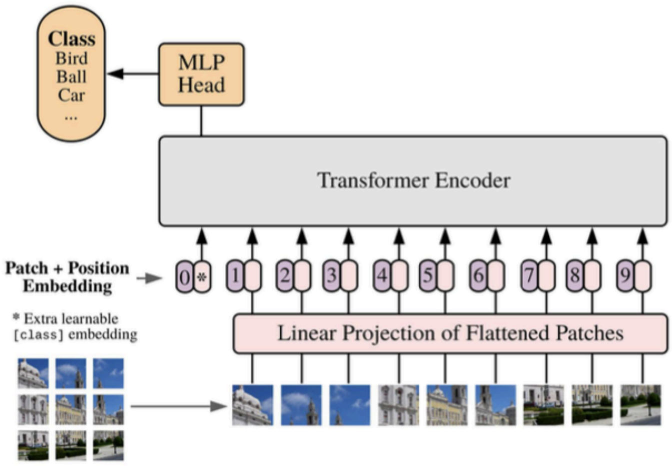
CNN-LSTM (reject): Used in the paper “A CNN-LSTM COMBINATION NETWORK FOR CATARACT DETECTION USING EYE FUNDUS IMAGES.” However, for our data, it showed an accuracy of 0.46, which was low. Even after modifying to VGG16-LSTM, it was 0.41.



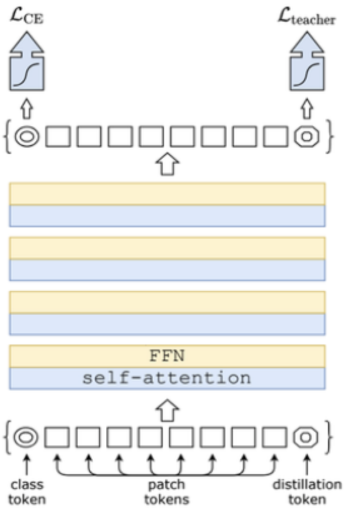
Pyramid-Net (reject): Suggested in "MTCD: Cataract Detection via Near Infrared Eye Images." Showed an accuracy of 0.58 on our data.



ViT (accept): A Transformer model that tokenizes images. Using pre-trained weights and adding two layers, it showed an initial accuracy of 0.79, making it the selected model.


Class
Bird
Ball
Car
...
MLP Head
Transformer Encoder
Patch + Position Embedding
0
*
1
2
3
4
5
6
7
8
9
* Extra learnable [class] embedding
Linear Projection of Flattened Patches

DeiT (reject): "Training data-efficient image transformers & distillation through attention" showed an accuracy of 0.60 due to difficulties in implementing the teacher function.



Final Model: Pre-trained ViT was selected, showing an accuracy of 0.849 with final preprocessed data.

3.3 Hyperparameter tuning:

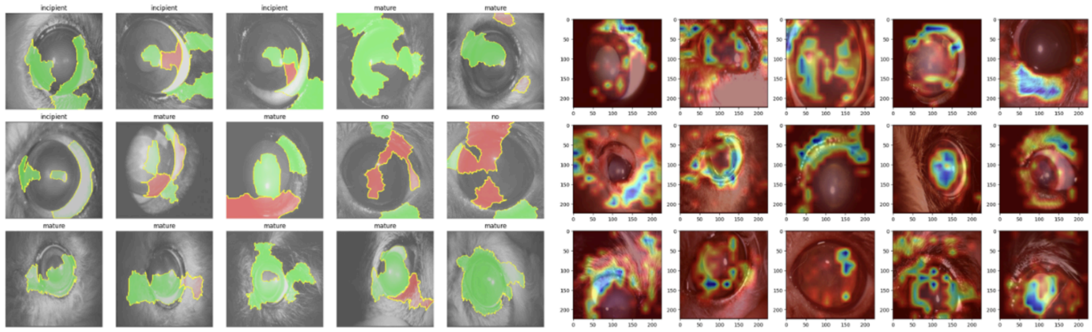
Used the optuna library for tuning learning rate (1e-5 to 1e-1) and batch size (16, 32, 64). Best accuracy was 0.842. Final settings were lr = 1e-4, batch size = 16.

3.4 Visualization:

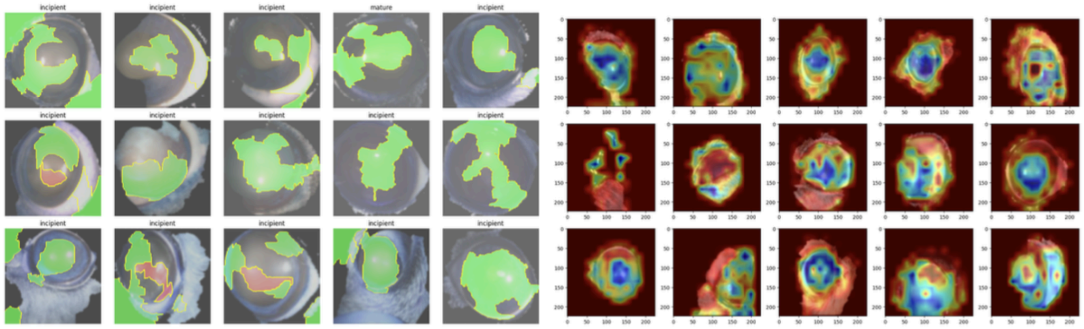
LIME: Provides explanations for each prediction using local linear regression.

Grad-CAM: Visualizes regions contributing to classification using saliency maps.
Both visualizations showed the model correctly identifying areas of cataracts in the final preprocessed data.

Initial Model by LIME & GradCAM



Final Model by LIME & GradCAM

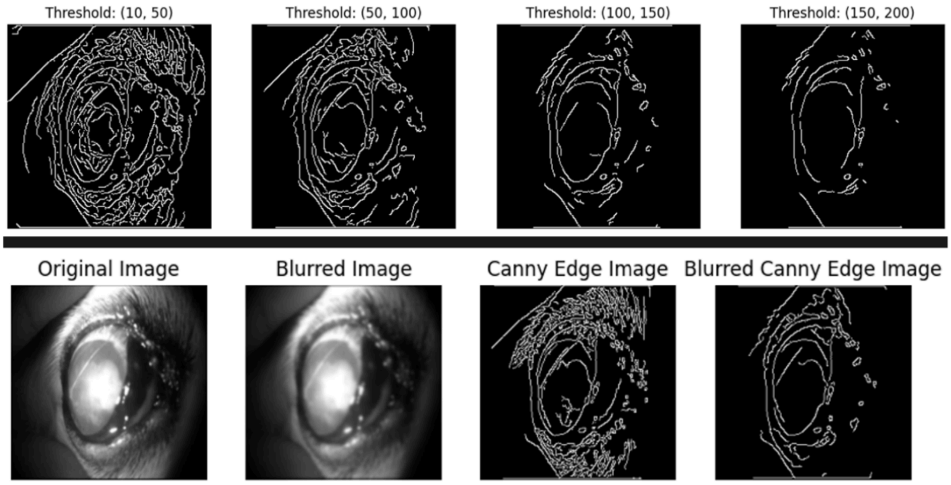


4. Results/Discussion

4.1 Analyze your result:

The final preprocessed data with pre-trained ViT (lr = 1e-4, batch size = 16) achieved an accuracy of 0.849. The visualizations also indicated appropriate detection areas.

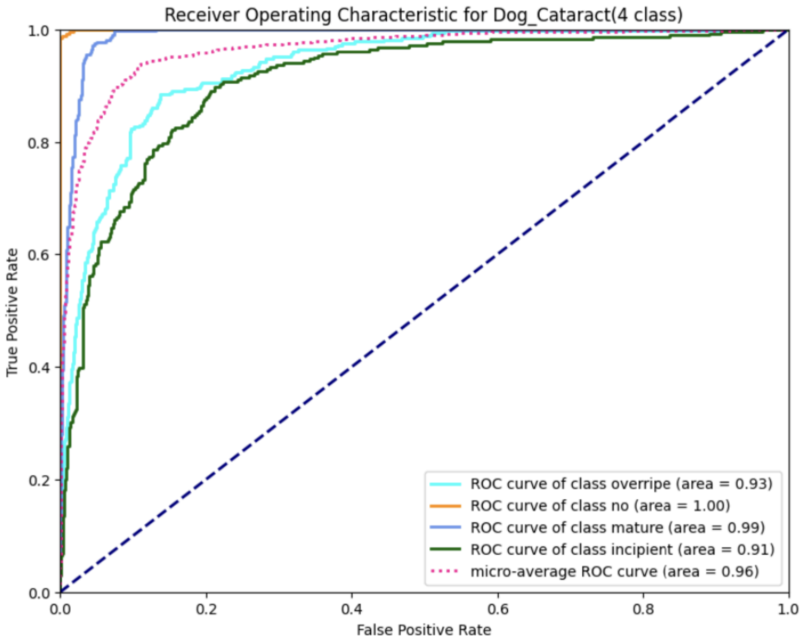
4.2 Unique findings:



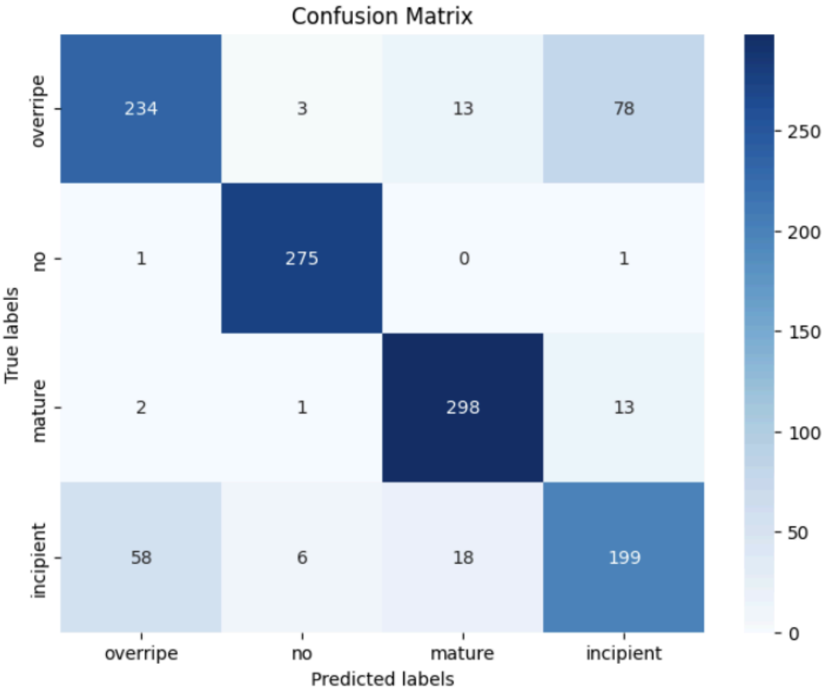
Early preprocessing without fur handling showed minimal effect on model performance. Using canny edge detection highlighted the significant impact of fur. Using grab-cut and blur reduced fur's influence, improving ViT model performance from 0.79 to 0.84. Pre-trained weights proved effective in limited datasets.

4.3 Model evaluation:

ROC curve: AUC value for 'no' class near 1, incipient class AUC 0.91, overall AUC 0.96.



Confusion Matrix: Shows strong predictions for ‘no’ and ‘mature’ classes,



True labels \ Predicted labels	overripe	no	mature	incipient
overripe	234	3	13	78
no	1	275	0	1
mature	2	1	298	13
incipient	58	6	18	199

Out of 277 actual “no” data points, 275 are accurately predicted as “no”. Out of 328 actual “overripe” data points, 234 are accurately predicted as “overripe”. Out of 311 “mature” data points, 298 are accurately predicted as “mature”. Out of 281 “incipient” data points, 199 are accurately predicted as “incipient”. The majority of data for the “no” and “mature” classes is correctly predicted. However, some data in the “incipient” and “overripe” classes are not clearly distinguished.