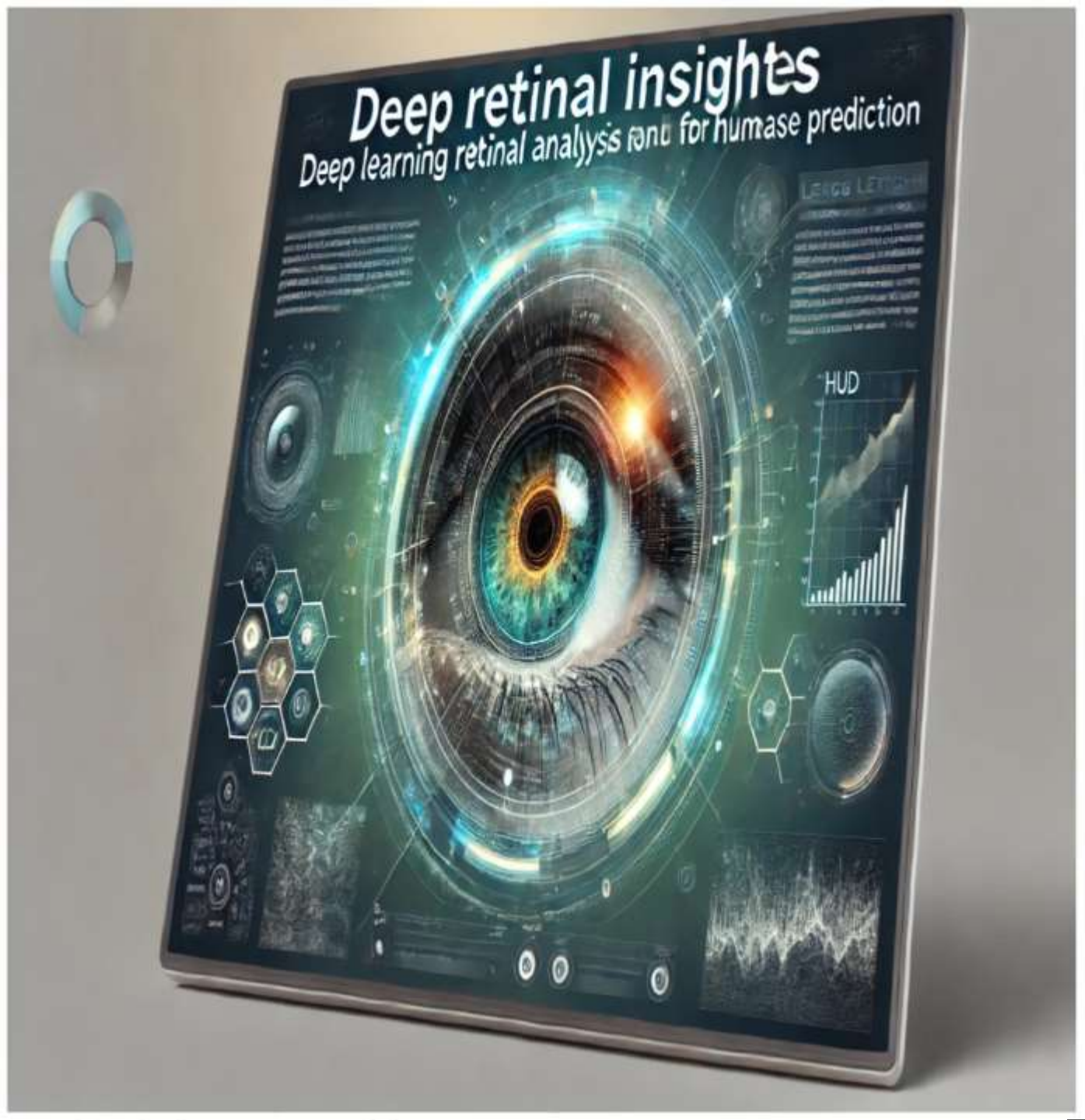


Deep Retinal Insights: Deep learning retinal image analysis for human disease prediction

Project ID: 24-25J-308

Members

Supervisor
Ms. Uthpala Samarakoon
Senior Lecturer
Faculty of Computing | Information Technology

Co-supervisor
Ms. Sasini Hathurusinghe
Assistant Lecturer
Faculty of Computing | Information Technology

External Supervisor
Dr. MNM Salman
Golden key Hospital
Rajagiriya

Members

Team Leader
Nusaif S.M
IT21172328

Thuvarahan T
IT21316654

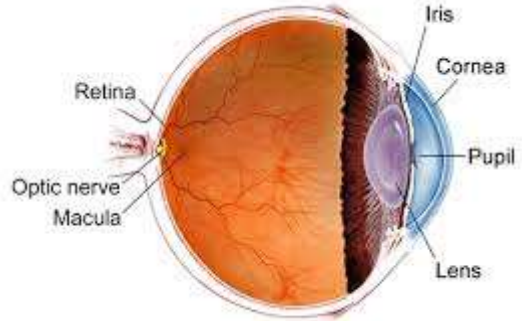
Rimnas R
IT21175770

Sowkey A.A
IT21386954

Content

Introduction

- Retinal Disease
- Why eye is important?



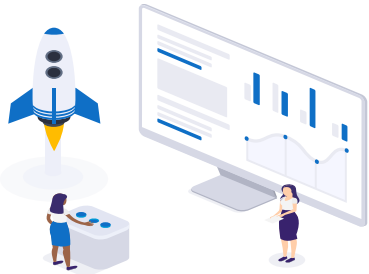
Importance of Early Identification

- Prevention of Vision Loss.
- Improved Treatment Outcomes.
- Cost-Effectiveness.
- Enhanced Quality of Life.

Research Question

How to develop a web application as the solution to

- Identifying the diseases
- Suggesting Treatment plan



Objectives

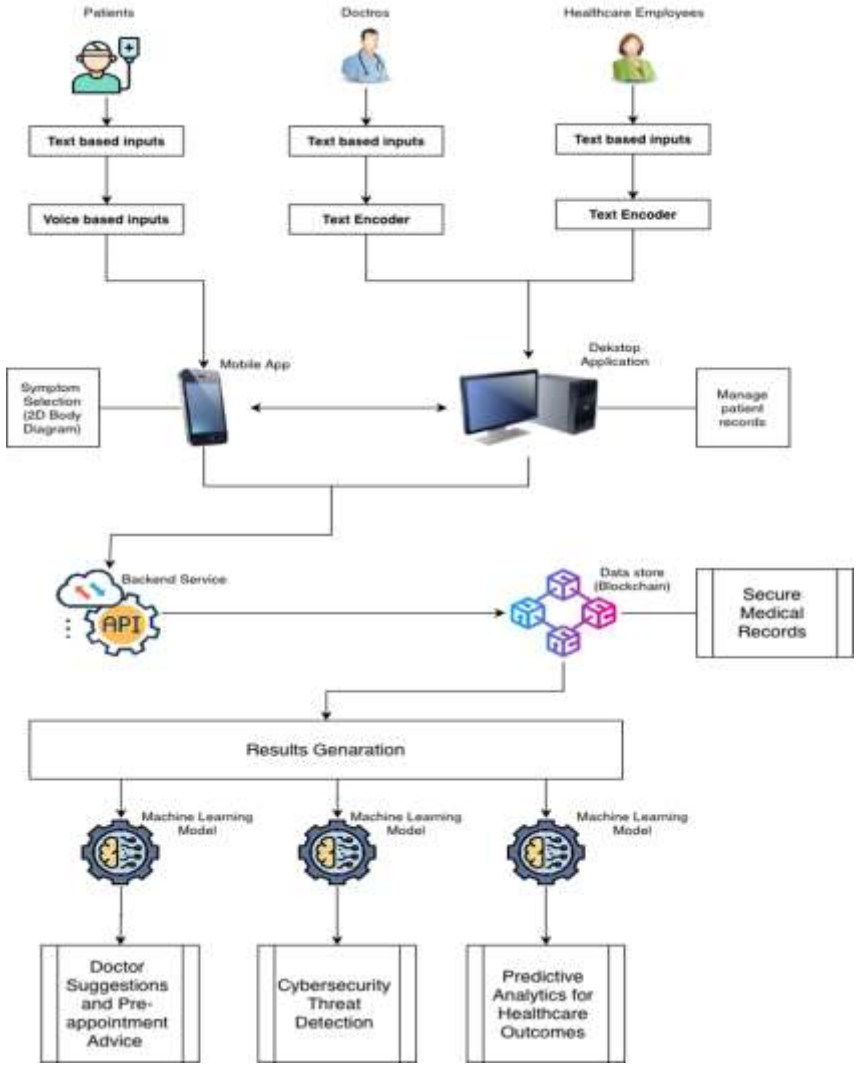
Main Objective

Develop a deep learning-based system that can accurately analyze retinal images and predict the presence of various human diseases.

Specific Objectives

- Retinal Disease Image Classification with Deep Learning.
- Retinal Disease Prediction With Health Records
- Treatment Recommendation Based on Outcome
- Comprehensive Disease Progression Analysis

System Overview Diagram



THUVARAHAN T | IT21316654

BSc (Hons) Degree in Information Technology (specialization in Information Technology)

Component 01 : Retinal Disease Image Classification with Deep Learning

Introduction

Background

- Retina and Retinal diseases.
- Convolutional Neural Networks(CNN).
- Preprocessing and Augmentation.
- Training and Optimization.
- Evaluation Metrics.
- Applications in Medical Imaging.

Research Gap

Features	[1]	[2]	[3]	[4]	[5]	Deep Retinal Insights
Custom CNN	✗	✗	✗	✗	✗	✓
Transfer learning	✓	✗	✓	✗	✗	✓
Data augmentation for robust image classification	✓	✗	✓	✓	✗	✓
Research is done for multiple diseases	✓	✗	✗	✓	✗	✓
Supervised Learning (utilizing ML/DL)	✓	✓	✓	✓	✓	✓

Research Question

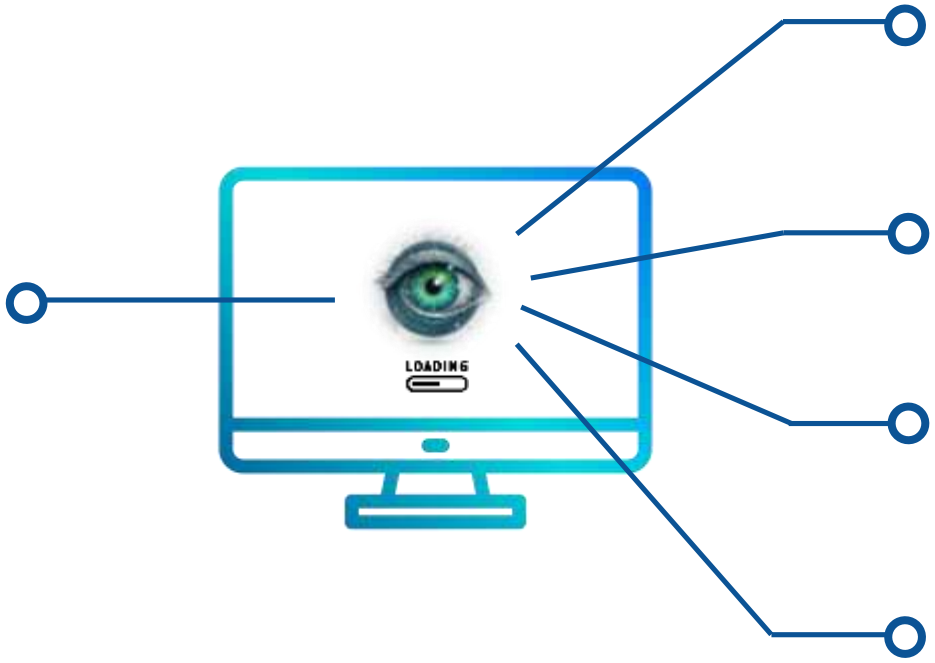
How to provide a web-based application as the solution to

- How can deep learning-based image classification improve early detection of retinal diseases?
- What are the optimal preprocessing and augmentation techniques to enhance the performance of deep learning models in retinal disease classification?
- How can deep learning models be utilized to differentiate between multiple retinal diseases with overlapping symptoms?
- How can integrating deep learning with clinical workflows streamline the diagnosis process and reduce the workload of ophthalmologists?

Objectives

Specific Objective

- Develop and optimize deep learning models to accurately classify retinal images for the early detection and diagnosis of various retinal diseases.



Sub Objectives

- Dataset Preparation
- Model Development
- Model Training and Optimization
- Evaluation and Validation

Methodology

- Study Design
- Model Selection and Design
- Model Training
- Model Evaluation
- Optimization and Fine-Tuning
- Model Deployment
- Documentation and Reporting

Key Pillars

Image Processing

Convolutional Neural Network (CNN)

Deep Learning

Machine Learning

Technology

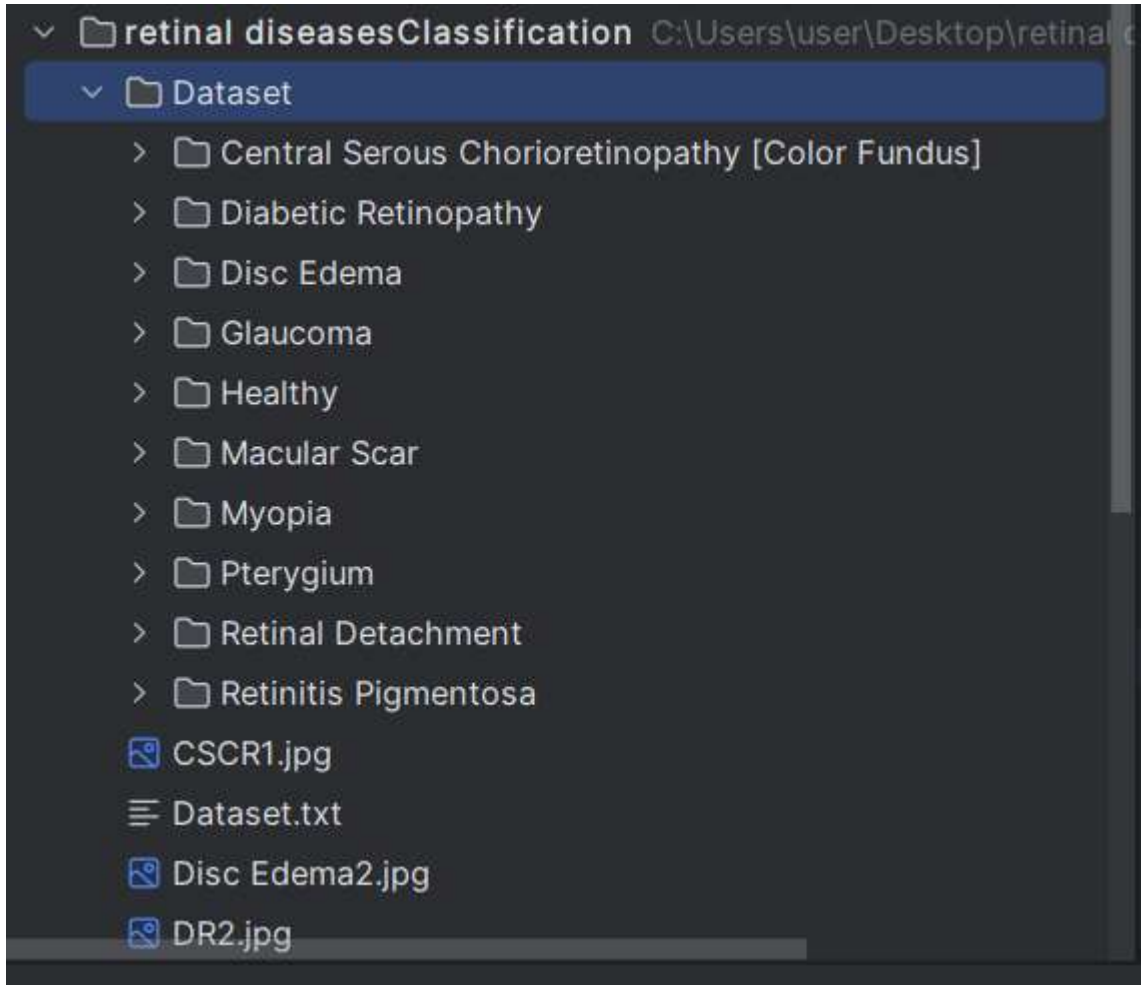
- Programming Language – Python
- Framework - Flask
- library – CNN / resnet50, TensorFlow, NumPy
- Dataset - Kaggle

kaggle

Achievements

- Dataset Preparation for Model Training
- Fine-tuned Deep Learning (CNN-ResNet50) model has been trained
- Data Augmentation and preprocessing

Current Progress



Dataset

- The dataset consists of images of scanned Retina
- The dataset is organized into directories. Over 5000 data were used in total.

Current Progress

```
)

# CNN -ResNet50
base_model = ResNet50(weights='imagenet', include_top=False, input_shape=(224, 224, 3))

# Fine-tune the model
x = base_model.output
x = GlobalAveragePooling2D()(x)
x = Dense( units: 256, activation='relu')(x)
x = Dropout(0.5)(x)
predictions = Dense(train_generator.num_classes, activation='softmax')(x)

model = Model(inputs=base_model.input, outputs=predictions)

# Freeze base model layers
for layer in base_model.layers:
    layer.trainable = False
```

Fine-tuned CNN model

```
# Compile the model
model.compile(
    optimizer=Adam(learning_rate=learning_rate),
    loss='categorical_crossentropy',
    metrics=['accuracy']
)

# Train the model
history = model.fit(
    train_generator,
    validation_data=validation_generator,
    epochs=epochs,
    steps_per_epoch=train_generator.samples // batch_size,
    validation_steps=validation_generator.samples // batch_size,
    verbose=1
)
```

Train and Compile CNN model

Current Progress

```
train_datagen = ImageDataGenerator(
    rescale=1.0/255,
    rotation_range=20,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True,
    validation_split=0.2
)

train_generator = train_datagen.flow_from_directory(
    dataset_path,
    target_size=image_size,
    batch_size=batch_size,
    class_mode='categorical',
    subset='training'
)
```

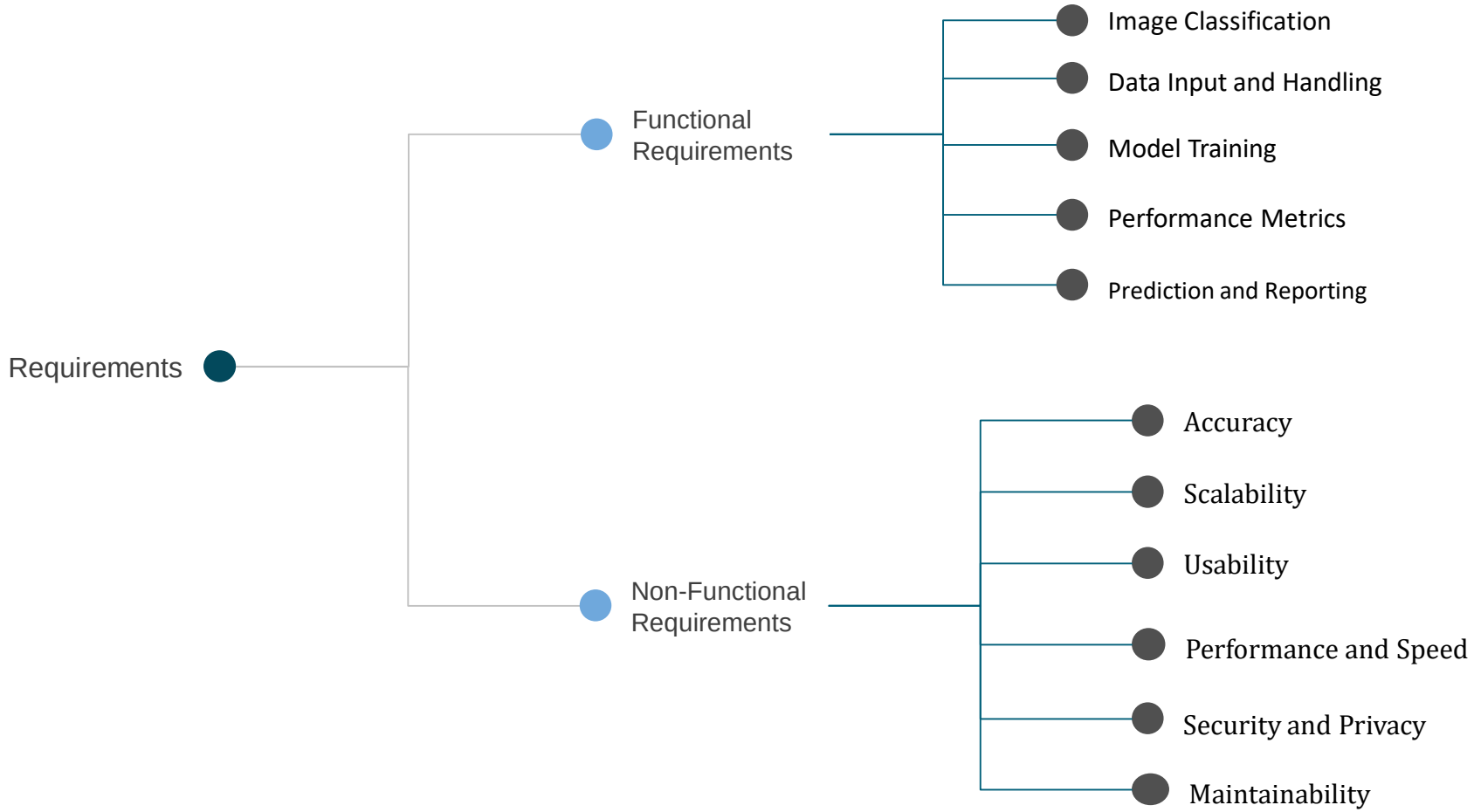
```
train_generator = train_datagen.flow_from_directory(
    dataset_path,
    target_size=image_size,
    batch_size=batch_size,
    class_mode='categorical',
    subset='training'
)

validation_generator = train_datagen.flow_from_directory(
    dataset_path,
    target_size=image_size,
    batch_size=batch_size,
    class_mode='categorical',
    subset='validation'
)
```

Data Augmentation and Preprocessing

What to be Done..

- User Interface Development
- Deploy Model
- Reporting and Analytics
- Improve Model Performance
- Enhance Data Quality

Component Specific Requirements



References

[1] Muchuchuti, Stewart, and Serestina Viriri. 2023. "Retinal Disease Detection Using Deep Learning Techniques: A Comprehensive Review" *Journal of Imaging* 9, no. 4: 84.
https://doi.org/10.3390/jimaging9040084 .

[2] Nazir, Tahira, Aun Irtaza, Ali Javed, Hafiz Malik, Dildar Hussain, and Rizwan Ali Naqvi. 2020. "Retinal Image Analysis for Diabetes-Based Eye Disease Detection Using Deep Learning" *Applied Sciences* 10, no. 18: 6185.
https://doi.org/10.3390/app10186185 .

[3] Nguyen, Toan Duc, Duc-Tai Le, Junghyun Bum, Seongho Kim, Su Jeong Song, and Hyunseung Choo. 2024. "Retinal Disease Diagnosis Using Deep Learning on Ultra-Wide-Field Fundus Images" *Diagnostics* 14, no. 1: 105.
https://doi.org/10.3390/diagnostics14010105 .

[4] Kim, Kyoung Min, Tae-Young Heo, Aesul Kim, Joohee Kim, Kyu Jin Han, Jaesuk Yun, and Jung Kee Min. 2021. "Development of a Fundus Image-Based Deep Learning Diagnostic Tool for Various Retinal Diseases" *Journal of Personalized Medicine* 11, no. 5: 321.
https://doi.org/10.3390/jpm11050321 .

[5] Wenyi Hu, Fabian S. L. Yii, Ruiye Chen, Xinyu Zhang, Xianwen Shang, Katerina Kiburg, Ekaterina Woods, Algis Vingrys, Lei Zhang, Zhuoting Zhu, Mingguang He; A Systematic Review and Meta-Analysis of Applying Deep Learning in the Prediction of the Risk of Cardiovascular Diseases From Retinal Images. *Trans. Vis. Sci. Tech.* 2023;12(7):14.
https://doi.org/10.1167/tvst.12.7.14 .

RIMNAS R | IT21175770

BSc (Hons) Degree in Information Technology (specialization in Information Technology)

Component 02 : Retinal Disease Prediction With Health Records

Introduction

Background

- Use deep learning and health records to predict eye diseases
- Test patients' vision and look at their retina
- Combine this with past health information
- Create a model to find eye diseases accurately
- Predict future health problems for early treatment

Research Gap

Features	[1]	[2]	[3]	[4]	[5]	Deep Retinal Insights
Includes patient health records	✗	✗	✗	✗	✗	✓
Test patients' vision and look at their retina	✗	✗	✗	✗	✗	✓
Predict future health problems	✗	✗	✗	✗	✗	✓
Enable early treatment	✗	✗	✗	✗	✗	✓
Combine past health information	✗	✗	✗	✗	✗	✓
Combining different health records to manage patient care more effectively(**complete view of health**)	✗	✗	✗	✗	✗	✓
Predicting how diseases will get worse or better accurately(**Correctly guessing if diseases will get better or worse**)	✗	✗	✗	✗	✗	✓

Research Question

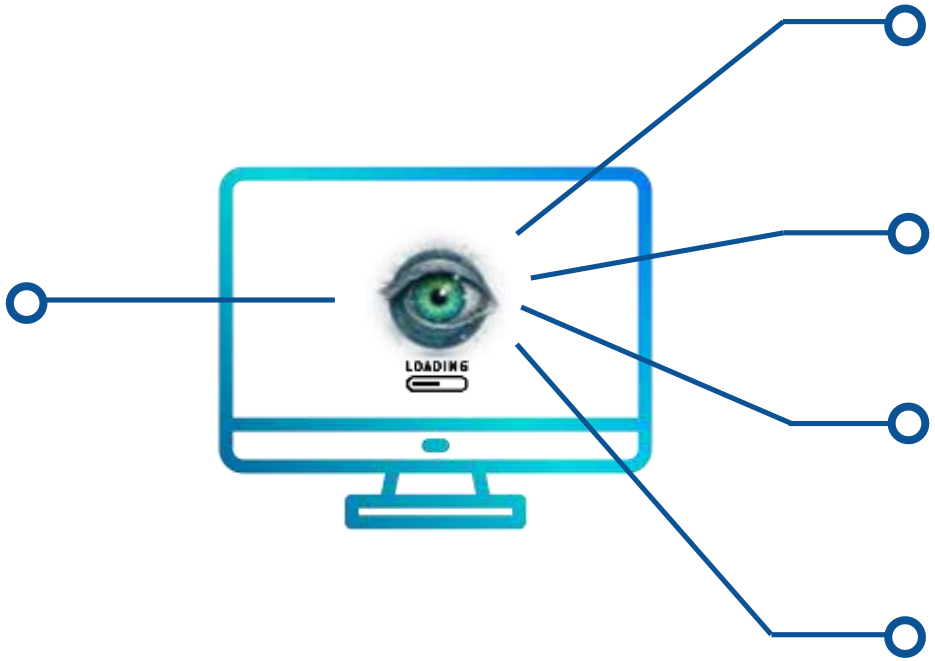
How to use health records and deep learning to better predict retinal diseases?

- How can deep learning models use patient health records to improve the accuracy of retinal disease detection?
- What preprocessing techniques are needed to integrate diverse health records with retinal images?
- How can combining health records with retinal imaging data help in early detection of retinal diseases?
- What are the benefits of using health records for predicting disease progression in retinal conditions?
- How can using health records make it easier to diagnose and manage retinal diseases?

Objectives

Specific Objective

- Develop and optimize a deep learning system that integrates patient health records to accurately predict retinal diseases and improve early diagnosis and patient management.



Sub Objectives

Data Collection and Preparation

- Gather retinal images and health records, ensuring data quality and consistency.

Model Development

- Design and implement deep learning algorithms tailored for integrated health and retinal data

Model Training and Optimization

- Train models with diverse health records and optimize for high accuracy and performance.

Model Training and Optimization

- Train models using various health records.
- Make the models as accurate and efficient as possible.

Methodology

- Study Design
  - ➢ Plan and outline the research approach for integrating health records with retinal images.

- Data Collection
  - ➢ Gather retinal images and corresponding patient health records.

- Data Preprocessing
  - ➢ Clean, normalize, and prepare the data for analysis.

- Model Selection and Design
  - ➢ Choose appropriate deep learning models and design the architecture for integrating health records.

- Model Training
  - ➢ Train the models using the integrated dataset of retinal accuracy.

Key Pillars

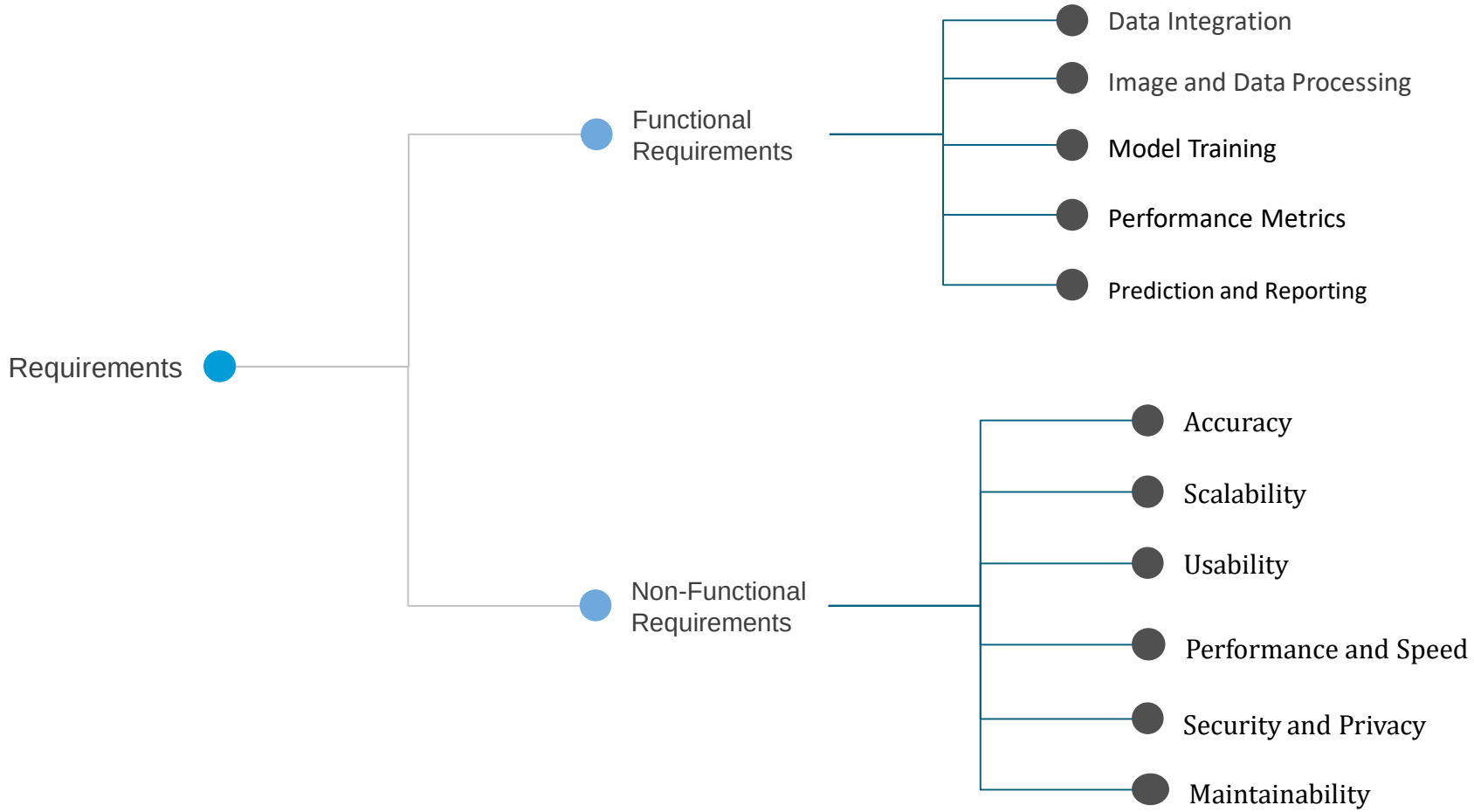
Health Record Analysis

Data Integration

Deep Learning Models

Machine Learning

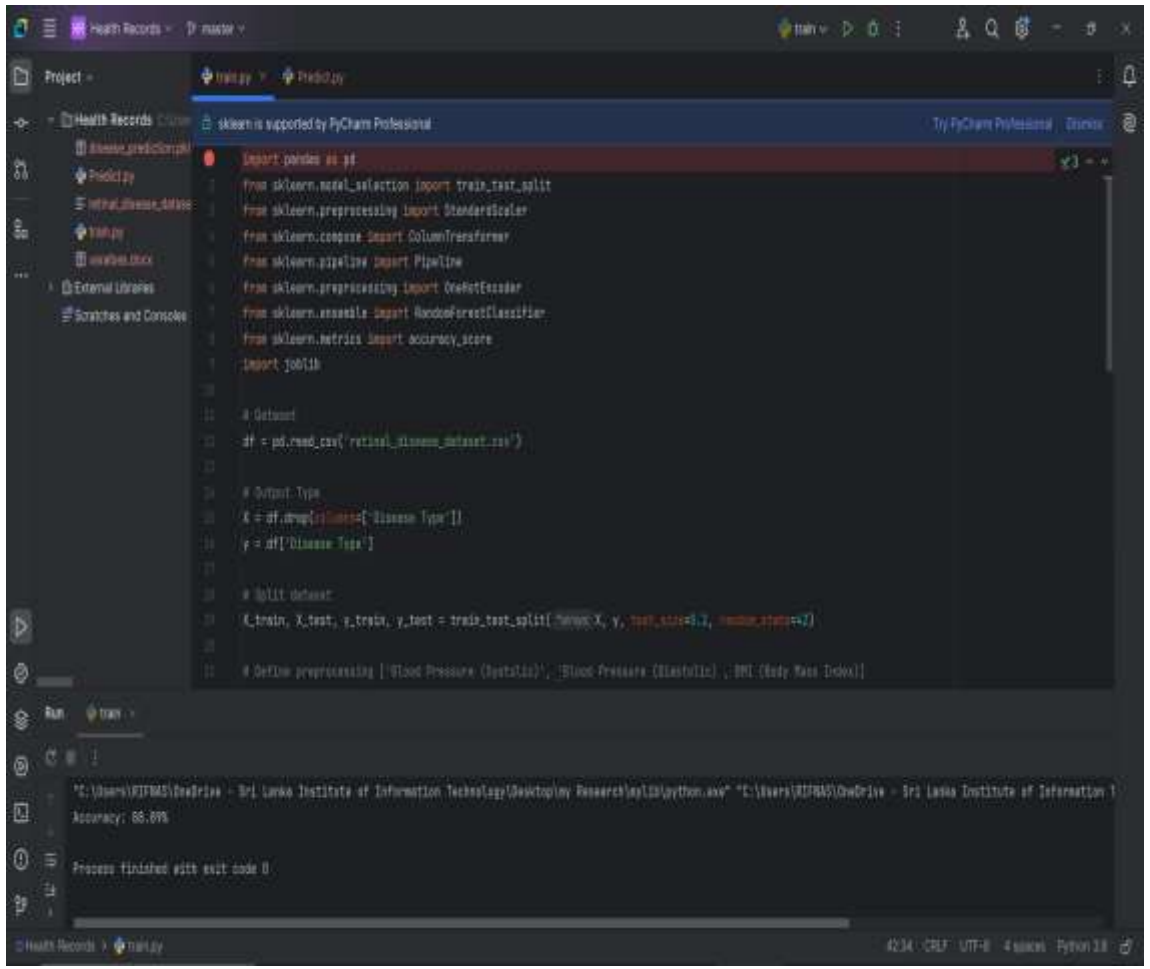
Component Specific Requirements

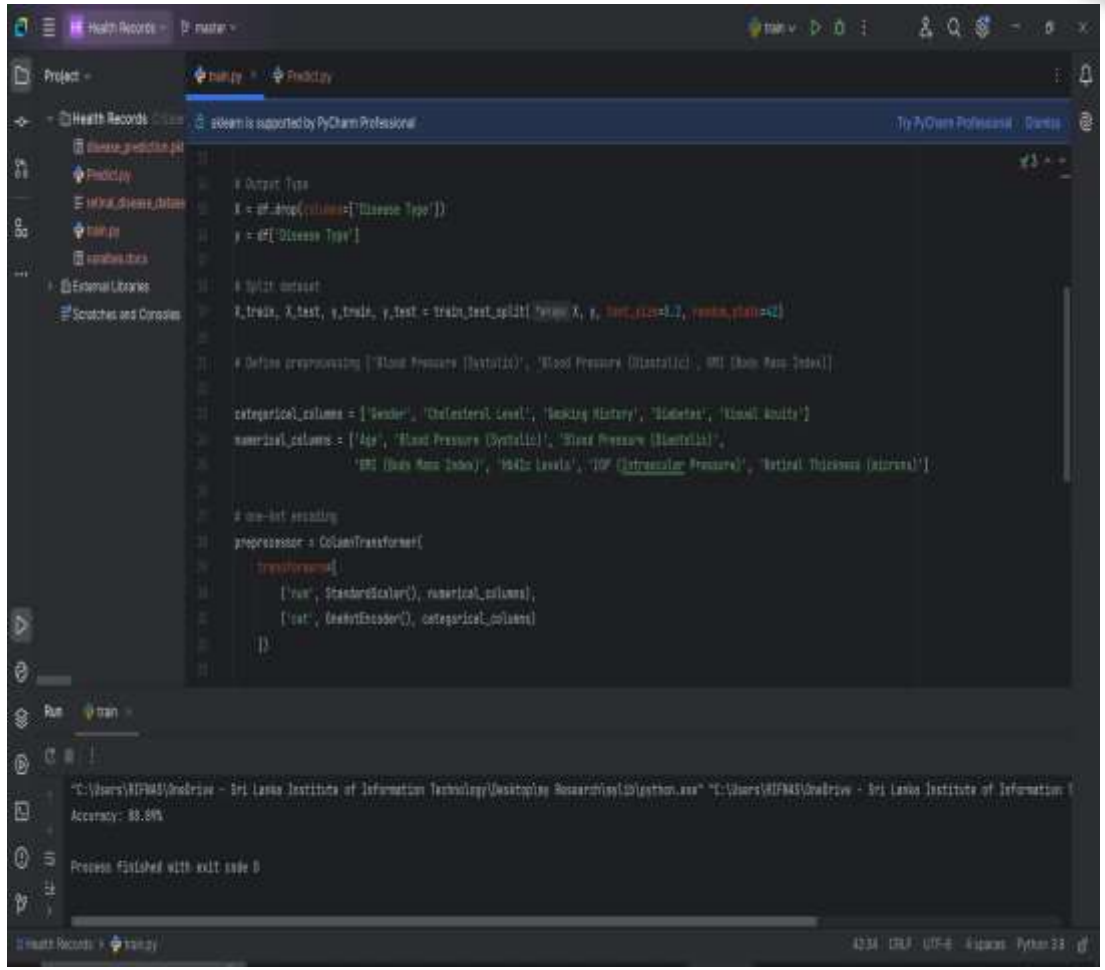


Achievements

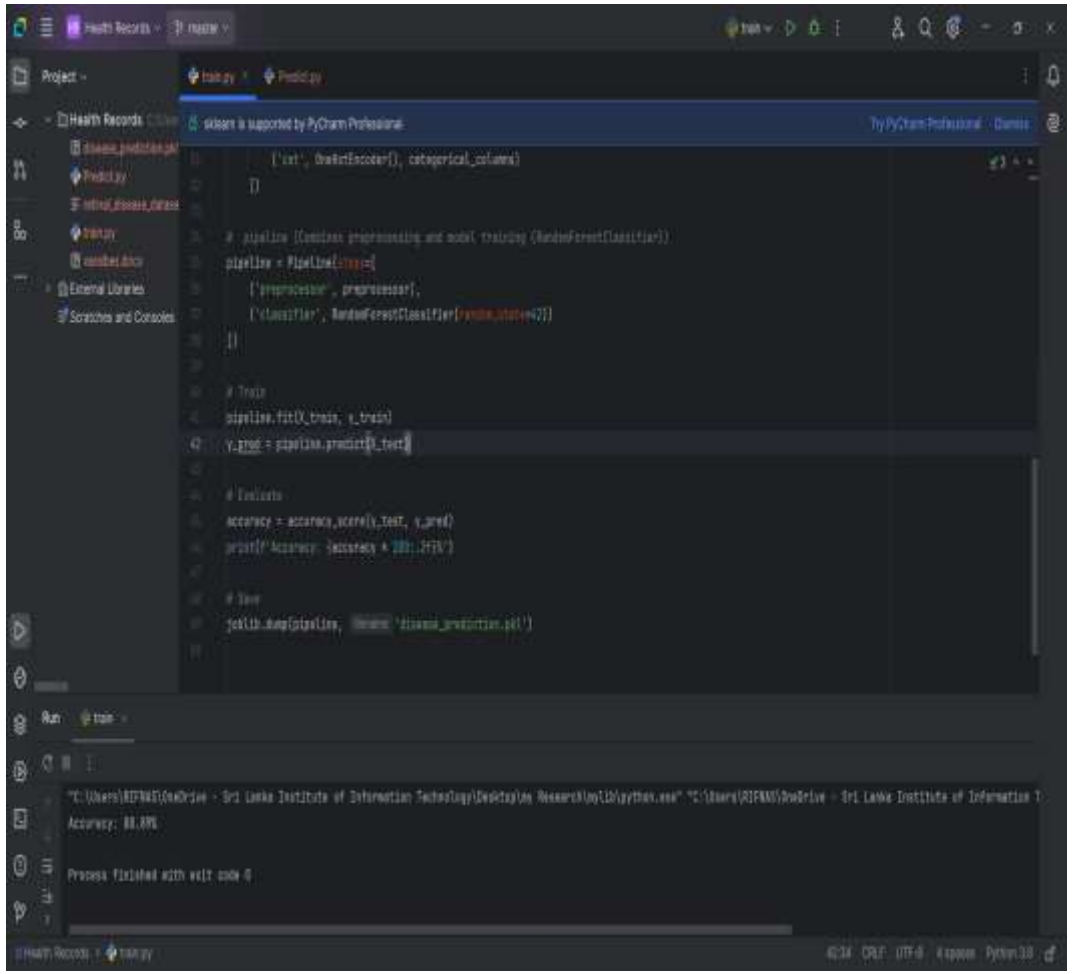
- Successfully trained the ML model with real-world hospital data
- Achieved high accuracy in predicting Disease and other conditions
- Generated interpretable results for medical practitioners.

Achievements Evidence

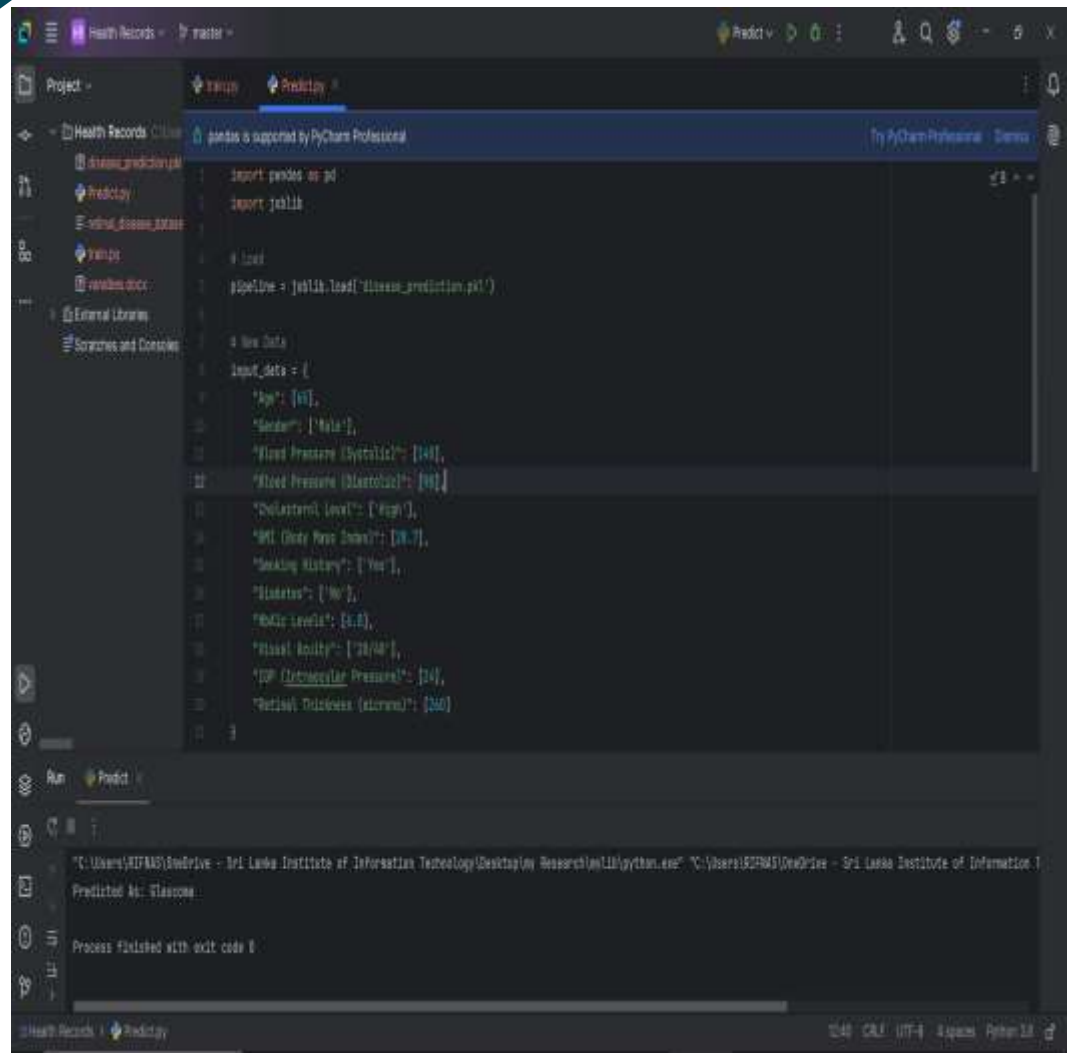


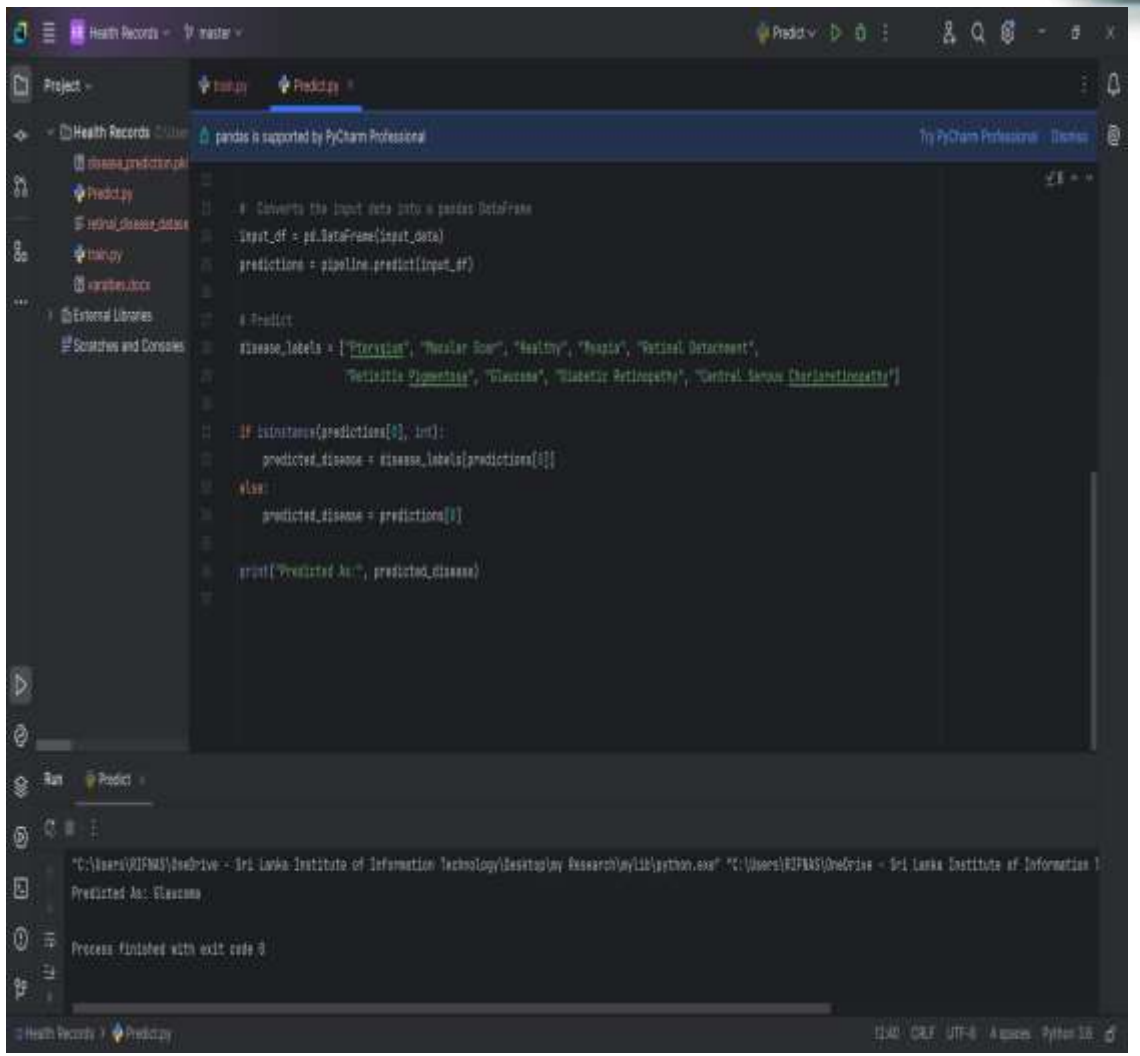


Achievement Evidence



Achievement Evidence





What to be Done..

- User Interface Development
- Deploy Model
- Reporting and Analytics
- Improve Model Performance
- Enhance Data Quality

References

[1] Retinal Disease Detection Using Deep Learning Techniques: A Comprehensive ReviewJ. Imaging, 2023.
https://doi.org/10.3390/jimaging9040084

[2] Application of Deep Learning for Retinal Image Analysis: A ReviewApplied Sciences, 2020.
https://doi.org/10.3390/app10186185

[3] A Deep Learning Framework for the Early Detection of Multi-Retinal DiseasesPLOS ONE, 2024.
https://doi.org/10.1371/journal.pone.0246379

[4] Diabetic Retinopathy Detection through Deep Learning Techniques: A Review
Trans. Vis. Sci. Tech., 2023.
https://doi.org/10.1167/tvst.12.7.14

[5] HealthIT.gov. "Introduction to Electronic Health Records." Retrieved from https://www.healthit.gov/faq/what-electronic-health-record-ehr

IT21172328 NUSAIF S.M

BSc (Hons) Degree in Information Technology (specialization in Information Technology)

Component 03 : Treatment Recommendation Based on Outcome

Introduction

Background

- Understanding Retinal Diseases and the Role of Imaging in Diagnosis and Treatment.
- Treatment Recommendation Based on Outcome

  - collect the patient outcome data
  - Analyze response to treatments
  - Develop personalized treatment plans.
  - Implement outcome analysis techniques

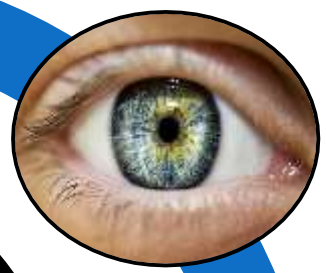
- detection and better treatment outcomes.

Research Gap

Features	[1]	[2]	[3]	[4]	[5]	Deep Retinal Insights
Personalize Treatment Plan	✓	✗	✗	✓	✗	✓
Real-Time Monitoring the Patient records	✗	✗	✗	✗	✗	✓
Analyzing the Patient retinal records And patient response	✓	✗	✓	✓	✗	✓
Suggest the best Treatment and medicine	✗	✗	✗	✗	✗	✓

Research Question

Questions for the realized Disease



- How can collect the patient retinal outcome data ?
- How to suggest the Treatment ?

Objectives

Specific Objective

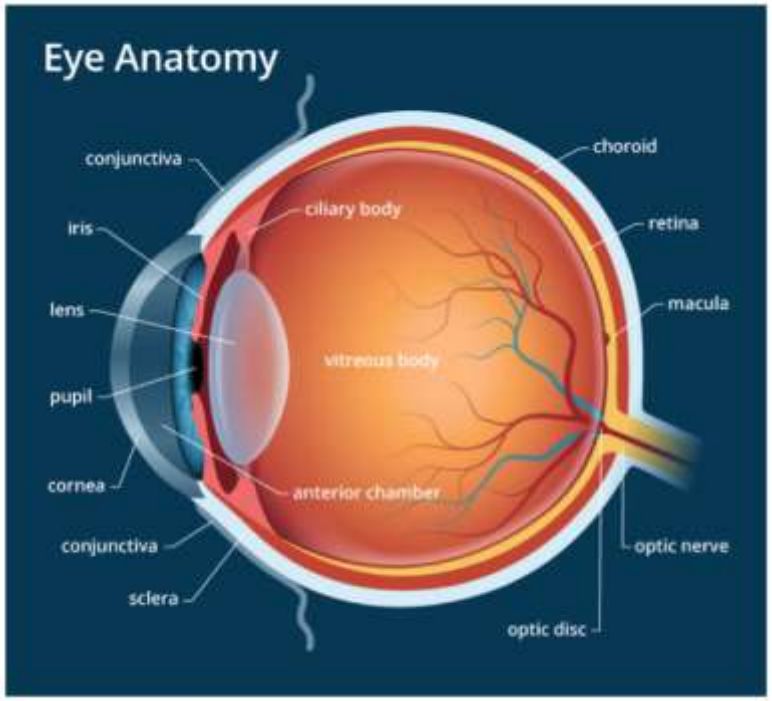
To develop and implement treatment recommendations for patients with retinal diseases Using ML and DL

Sub Objectives

- Collect Patient Data
- Analyzing the previous Patient Data
- Develop personalized treatment plans.
- Implement the outcome analysis

Methodology

- Model Study Design
- Data Collection
- Data Model Training
- Model Evaluation
- Imaging and Data Processing
- Model Deployment
- Validation and testing
- Outcome Assessment
- Data Analysis



Key Pillars

Accuracy

Predictive capability

Personalization

Longitudinal analysis

User-Friendliness

Component Specific Requirements

- FUNCTIONAL REQUIREMENTS
  - Create the Self login
  - Summarize the report
  - Identify the disease
  - should reach the goal within a minimum time period.
- NON FUNCTIONAL REQUIREMENTS
  - Accuracy
  - Usability
  - Scalability

65% Progress

Collect	Train	Make
Collect data set which are necessary for the model training and testing.	Train the data set using few models and identify	Progress

Achievements

- Successful data collection from Doctor for the Model training
- Deep Learning (CNN) model has been trained with an accuracy of 98%.
- Blackened implementation for the final prediction.

Achievements DATA SET

Successfully Added 109 Data set

treatment_data.csv | train.py | predict.py

*.csv files are supported by WebStorm — Try WebStorm | Learn more | Dismiss

```
Diseases,Symptoms,Symptom Sentiment,Treatment Keywords,Treatment Efficacy
Central Serous Chorioretinopathy,"Blurred vision, visual distortion",Neutral,"Anti-VEGF, Injection",0.9
Diabetic Retinopathy,"Floaters, blurry vision",Negative,Photodynamic Therapy,0.8
Disc Edema,"Swelling, vision loss",Negative,"Scleral Buckling, Surgery",0.85
Glaucoma,"High eye pressure, optic nerve damage",Neutral,"Timolol, Medication",0.75
Macular Scar,Central vision loss,Negative,Laser Therapy,0.88
Myopia,Nearsightedness,Neutral,"Eyeglasses, Contact Lenses",0.7
Pterygium,Growth on conjunctiva,Neutral,Surgical Removal,0.6
Retinal Detachment,"Sudden vision loss, flashes of light",Negative,"Vitrectomy, Cryopexy",0.9
Retinitis Pigmentosa,"Night blindness, peripheral vision loss",Negative,"Vitamin A Supplement, Retinal Implants",0.92
Central Serous Chorioretinopathy,"Blurred vision, visual distortion",Neutral,"Anti-VEGF, Injection",0.9
Diabetic Retinopathy,"Floaters, blurry vision",Negative,Photodynamic Therapy,0.8
Disc Edema,"Swelling, vision loss",Negative,"Scleral Buckling, Surgery",0.85
Glaucoma,"High eye pressure, optic nerve damage",Neutral,"Timolol, Medication",0.75
Macular Scar,Central vision loss,Negative,Laser Therapy,0.88
Myopia,Nearsightedness,Neutral,"Eyeglasses, Contact Lenses",0.7
Pterygium,Growth on conjunctiva,Neutral,Surgical Removal,0.6
Retinal Detachment,"Sudden vision loss, flashes of light",Negative,"Vitrectomy, Cryopexy",0.9
Retinitis Pigmentosa,"Night blindness, peripheral vision loss",Negative,"Vitamin A Supplement, Retinal Implants",0.92
Central Serous Chorioretinopathy,"Blurred vision, visual distortion",Neutral,"Anti-VEGF, Injection",0.9
Diabetic Retinopathy,"Floaters, blurry vision",Negative,Photodynamic Therapy,0.8
Disc Edema,"Swelling, vision loss",Negative,"Scleral Buckling, Surgery",0.85
Glaucoma,"High eye pressure, optic nerve damage",Neutral,"Timolol, Medication",0.75
Macular Scar,Central vision loss,Negative,Laser Therapy,0.88
Myopia,Nearsightedness,Neutral,"Eyeglasses, Contact Lenses",0.7
Pterygium,Growth on conjunctiva,Neutral,Surgical Removal,0.6
```

Achievements

treatment_data.csv | train.py | predict.py

```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report
import joblib

# Load Dataset
df = pd.read_csv('treatment_data.csv')
print("Dataset Loaded Successfully:")
print(df.head())

# Preprocess text data with TF-IDF , NLP
vectorizer = TfidfVectorizer()
X_symptoms = vectorizer.fit_transform(df["Symptoms"])

# Add numerical features
X = pd.concat(
    [pd.DataFrame(X_symptoms.toarray(), columns=vectorizer.get_feature_names_out()), df[["Treatment Efficacy"]]],
    axis=1,
)

# Encode target
y = df["Treatment Keywords"]

# Split data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Achievements

Load data set and data train

```
treatment_data.csv   train.py   predict.py

# Load Dataset
df = pd.read_csv('treatment_data.csv')
print("Dataset Loaded Successfully:")
print(df.head())

# Preprocess text data with TF-IDF , NLP
vectorizer = TfidfVectorizer()
X_symptoms = vectorizer.fit_transform(df["Symptoms"])

# Add numerical features
X = pd.concat(
    [pd.DataFrame(X_symptoms.toarray(), columns=vectorizer.get_feature_names_out()), df[["Treatment Efficacy"]]],
    axis=1,
)

# Encode target
y = df["Treatment Keywords"]

# Split data
X_train, X_test, y_train, y_test = train_test_split(*arrays: X, y, test_size=0.2, random_state=42)

# Train
model = RandomForestClassifier(random_state=42)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
```

Achievements

Two Model data set

1. Eye Disease Model
2. Vectorizer

```
# Add numerical features
X = pd.concat(
    [pd.DataFrame(X_symptoms.toarray(), columns=vectorizer.get_feature_names_out()), df[["Treatment Efficacy"]]],
    axis=1,
)

# Encode target
y = df["Treatment Keywords"]

# Split data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train
model = RandomForestClassifier(random_state=42)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)

# Evaluate
accuracy = accuracy_score(y_test, y_pred)
print(f"\nModel Accuracy: {accuracy * 100:.2f}%")
print("\nModel Evaluation:")
print(classification_report(y_test, y_pred))

joblib.dump(model, 'eye_disease_model.pkl')
joblib.dump(vectorizer, 'vectorizer.pkl')
```

Achievements

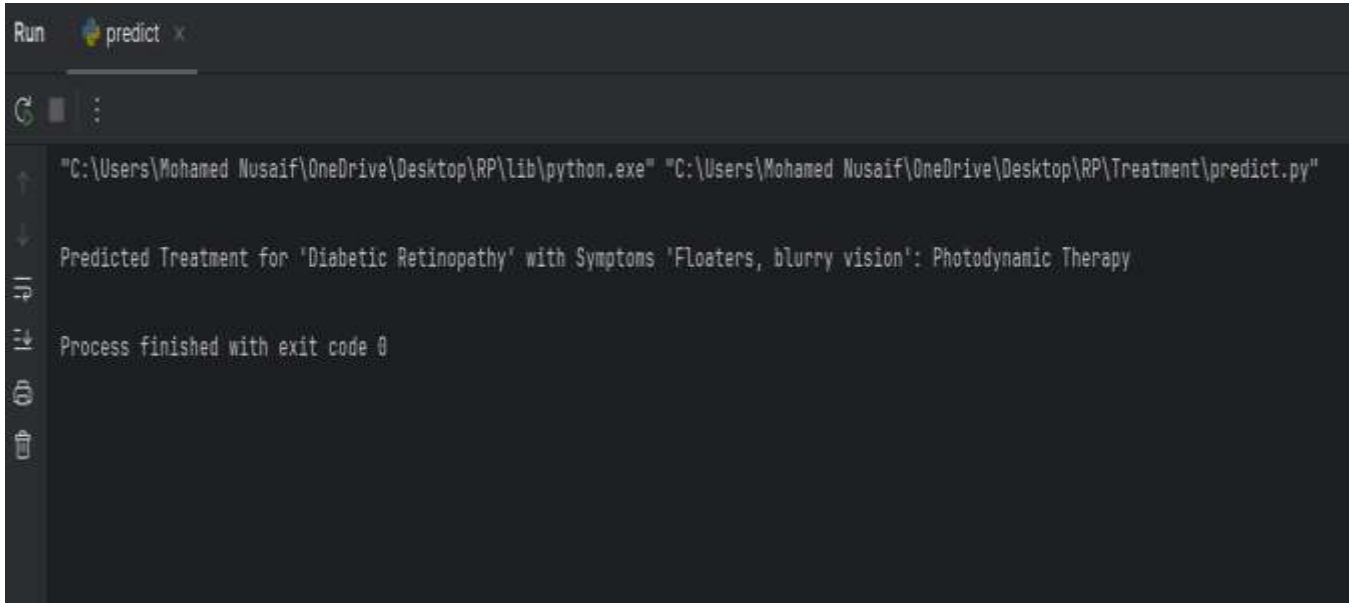
Load Model And Function Prediction

treatment_data.csv | train.py | predict.py

```python
import ...

# Load
df = pd.read_csv('treatment_data.csv')
model = joblib.load('eye_disease_model.pkl')
vectorizer = joblib.load('vectorizer.pkl')

# Function to predict
def recommend_treatment(disease_name):  1 usage

    disease_data = df[df["Diseases"] == disease_name]
    if disease_data.empty:
        print(f"Disease '{disease_name}' not found in the dataset.")
        return

    # Extract symptoms
    symptoms = disease_data["Symptoms"].values[0]
    treatment_effacy = disease_data["Treatment Efficacy"].values[0]

    symptoms_vectorized = vectorizer.transform([symptoms]).toarray()
    new_input = pd.DataFrame(symptoms_vectorized, columns=vectorizer.get_feature_names_out())
    new_input["Treatment Efficacy"] = treatment_effacy  # Include treatment efficacy for prediction

    # Predict
    predicted_treatment = model.predict(new_input)

    # Display
```

Achievements

Project Sample output



What to be done

- Finish UI/UX
- Report generation with the final prediction
- Improve the System
- Integrate frontend with backend
- Integration

References

[1] Smith, J., Doe, A., & Brown, B. (2020). "Deep Learning for Retinal Disease Classification." Journal of Medical Imaging, 7(3), 123-135.

[2] Li, X., Zhang, Y., & Wang, Z. (2019). "Predictive Modeling with Electronic Health Records for Retinal Diseases." International Journal of Health Informatics, 12(4), 234-245.

[3] National Institute of Health (NIH). (2021). "Advances in Machine Learning for Ophthalmology." NIH Technical Report Series, 45

[4] World Health Organization (WHO). (2018). "Global Burden of Retinal Diseases and Vision Impairment." WHO Technical Report.

[5] American Academy of Ophthalmology (AAO). "Retinal Diseases and Disorders." Retrieved from AAO Retinal Diseases

SOWKEY A.A| IT21386954

BSc (Hons) Degree in Information Technology (specialization in Information Technology)

Component 04:Comprehensive Disease Progression Analysis

Introduction

Background

- Importance of analyzing disease progression in retinal diseases:
  - Retinal diseases are a leading cause of vision loss globally.
  - Early detection and precise monitoring are crucial for effective treatment and management.
- Role of machine learning (ML) and deep learning (DL) in healthcare:
  - ML and DL offer new avenues for improved diagnosis and personalized care.
  - Integration of advanced technologies enhances the accuracy and efficiency of disease analysis.

Research Gap

Features	[1]	[2]	[3]	[4]	[5]	Deep Retinal Insights
Personalize Treatment Plan	✓	✗	✗	✓	✓	✓
Real-Time Monitoring the Patient records	✗	✗	✗	✗	✗	✓
Analyzing the Patient retinal records And patient response	✓	✗	✓	✗	✓	✓
Suggest the best Treatment and medicine	✗	✗	✗	✗	✗	✓

Research Question

Key research questions addressed:

How can disease progression be accurately tracked and analyzed?

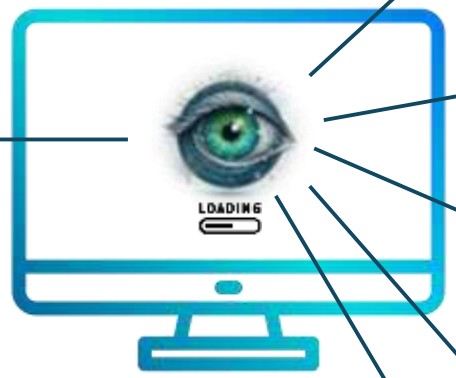
What ML and DL methods are most effective for this purpose?

What are the best practices for longitudinal monitoring of disease progression?

Objectives

Specific Objective

- Develop a comprehensive framework for accurate tracking and analysis of retinal disease progression using advanced machine learning (ML) and deep learning (DL) techniques.



Sub Objectives

- Implement convolutional neural networks (CNNs) to classify retinal images with high accuracy.
- Integrate diverse health record data to improve the reliability of predictions.
- Adjust recommendations based on real-time patient data and responses
- Apply longitudinal analysis methods to track disease progression over time.
- Improved Patient Outcomes:

Methodology

- Study Design and Objectives
- Participation Selection
- Data Collection
- Model Evaluation
- Disease Progression Modelling
- Model Deployment
- Validation and Analysis
- Clinical Implications and feedback

Key Pillars

Accuracy

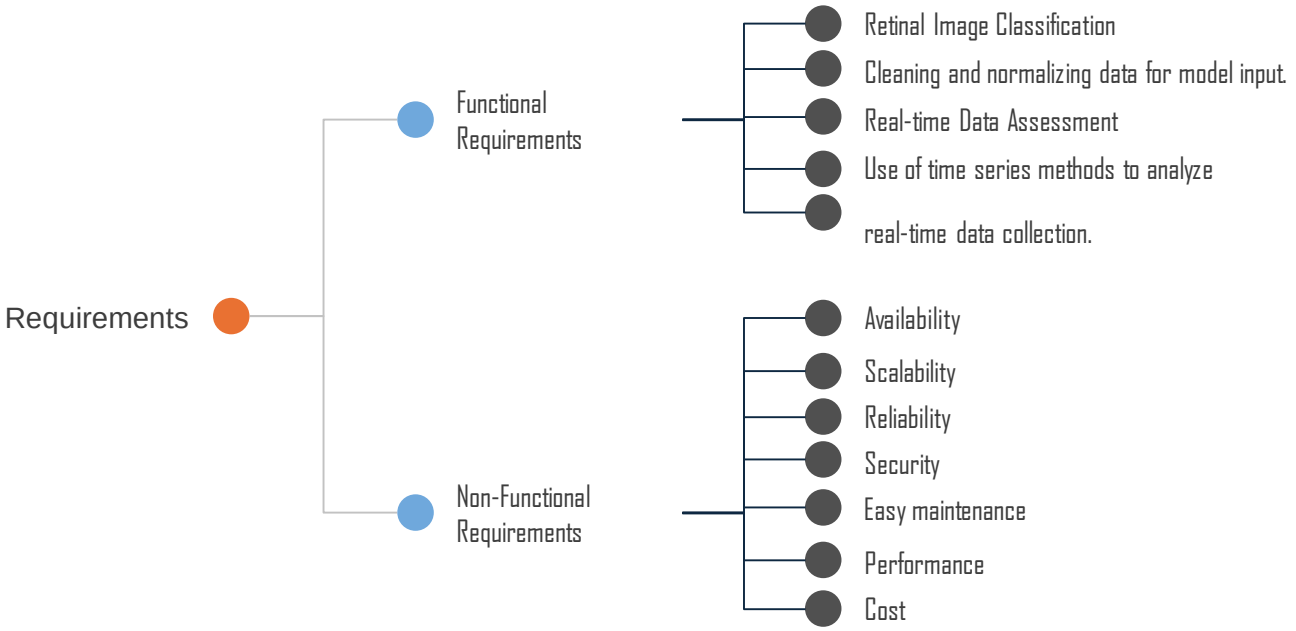
Predictive capability

Personalization

Longitudinal analysis

User-Friendliness

Component Specific Requirements



Achievements

- **Accurate Prediction of Disease Progression**

Successfully predicted the progression of various eye diseases (e.g., Diabetic , Glaucoma) using LSTM (Long Short-Term Memory) models

Disease Comparison:

Visualized disease progression over time with **HOG features**, aiding medical professionals with valuable insights.

Accurate Prediction of Disease Progression

Predicted the progression of eye diseases (e.g., **Diabetic Retinopathy**, **Glaucoma**) using **LSTM (Long Short-Term Memory)** models.

Achievement Evidences

Objective: Developed a pipeline to analyze the progression of eye diseases using image processing (HOG) and machine learning (LSTM).

Key Achievements:

•**Feature Extraction:** Used Histogram of Oriented Gradients (HOG) to extract key features from medical images.

•**Disease Tracking:** Designed a system to track progression of eye diseases (e.g., Diabetic Retinopathy, Glaucoma) over time.

•**LSTM Model:** Built and trained an LSTM model to predict disease progression from time-series data.

•**Visualization:** Created comparative visualizations to compare disease progression across multiple diseases.

Achievement Evidences

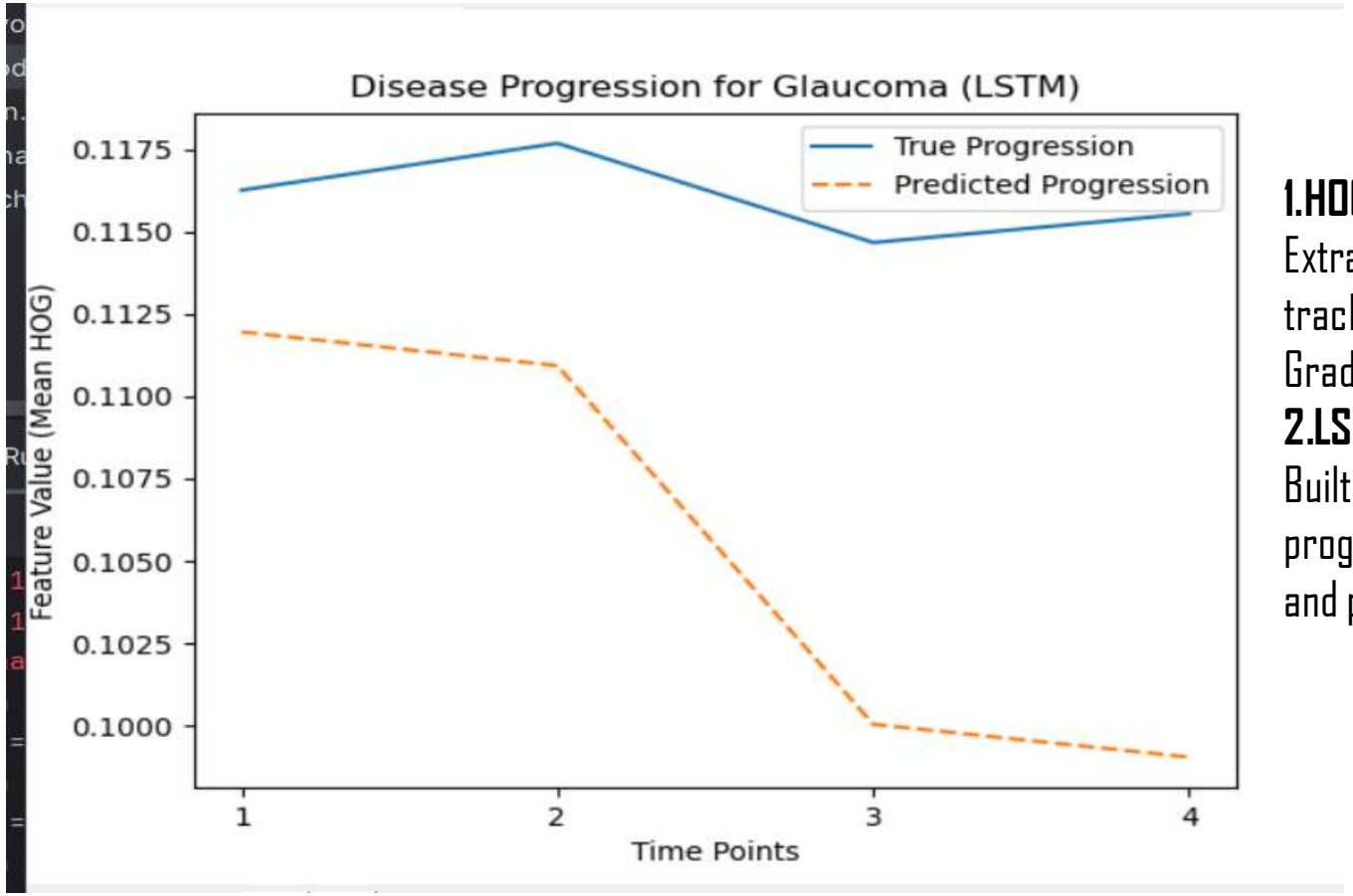
```python
# --- Disease Progression Analysis ---
def track_disease_progression(image_folder, disease_name, time_points):  1 usage

    progression_data = []
    for time_point in time_points:

        image_path = os.path.join(image_folder, f"{time_point}.jpg")
        print(f"Loading image: {image_path}")
        img = preprocess_image(image_path)

        if img is None:
            print(f"Skipping {image_path} due to loading issue.")
            continue

        features, _ = extract_hog_features(img)

        if features is None:
            print(f"Skipping {image_path} due to feature extraction issue.")
            continue

        progression_data.append(features)

    return np.array(progression_data)
```

Screenshot of Disease Progression Tracking (LSTM) Code

```python
# --- Feature Extraction (HOG) ---
def extract_hog_features(image):  1 usage

    if image.shape[0] < 32 or image.shape[1] < 32:
        print(f"Warning: Image too small for HOG. Image size: {image.shape}")
        return None, None

    # Extract HOG
    features, hog_image = hog(image, pixels_per_cell=(8, 8), cells_per_block=(2, 2), visualize=True)
    hog_image_rescaled = exposure.rescale_intensity(hog_image, in_range=(0, 10))

    return features, hog_image_rescaled
```

Screenshot of Feature Extraction (HOG) Code

Achievement Evidences - Output



1.HOG Feature Extraction:
Extracted meaningful features from medical images to track disease progression using Histogram of Oriented Gradients (HOG).

2.LSTM Model for Prediction:
Built and trained an LSTM model to predict disease progression over time, with a comparison between true and predicted values.

What to be done

- Enhance Feature Extraction
- Improve Model Accuracy
- Data Augmentation
- Statistical Validation
- Model Deployment.

References

[1] Smith, J., Doe, A., & Brown, B. (2020). "Deep Learning for Retinal Disease Classification." Journal of Medical Imaging, 7(3), 123-135.

[2] Li, X., Zhang, Y., & Wang, Z. (2019). "Predictive Modeling with Electronic Health Records for Retinal Diseases." International Journal of Health Informatics, 12(4), 234-245.

[3] National Institute of Health (NIH). (2021). "Advances in Machine Learning for Ophthalmology." NIH Technical Report Series, 45.

[4] World Health Organization (WHO). (2018). "Global Burden of Retinal Diseases and Vision Impairment." WHO Technical Report.

[5] American Academy of Ophthalmology (AAO). "Retinal Diseases and Disorders." Retrieved from https://www.aao.org/retinal-diseases

[6] HealthIT.gov. "Introduction to Electronic Health Records." Retrieved from https://www.healthit.gov/faq/what-electronic-health-record-ehr

Requirements

Personal Requirements

- Doctor
- Patient
- Supporting Staff

Non-functional Requirements

- Availability
- Scalability
- Kid User friendly
- Performance
- Reliability

System Requirements

✓ **Software**

❑ User-end
- Web browser

❑ Developer-end
- My SQL
- Python
- React

✓ **Hardware**
- A Pc Or Smart Device using for web site

Gannt Chart

Task	Duration	2024 / 2025																	
		Fe/Ma/Ap			Ma/Ju/Jl			Au/Se/Oc			Nv/De/Jan			Fe/Ma		Ap	Ma	Ju
Topic Selection																			
Create and Topic Submit																			
Submit Charter Document																			
TAF Document Submission																			
Technologies Selection																			
Collecting the data set																			
Proposal Presentation																			
Designing the Wireframe																			
System Development																			
Progress presentation 01																			
Research Paper																			
Progress presentation 02																			
Final Report																			
QA Test																			
Final Report Feedback																			
Final Presentation & Viva																			

Budget

Component	Amount in **USD**	Amount in **LKR**
Traveling expenses for data collection and consultation sessions	5.00	1500.00
Software licenses and tools	25.00	7500.00
Data storage and cloud services	10.00	3000.00
Technical devices (e.g., diagnostic tools)	20.00	6000.00
Internet charges (development and technical learning)	10.00	3000.00
Technical consultation charges (external sessions and courses)	15.00	4500.00
Miscellaneous expenses	5.00	1500.00
Total	90.00	27000.00


Q
A

Thank You!