

* Batch-13 *

DIABETIC RETINOPATHY DETECTION

USING CNN ALGORITHMS

BOJJI ANIRUDH-043
BOLLU SIDDHARTH REDDY-055
Guided by Dr Sirisha Daggubati





CONTENT

01

Abstract

02

Problem
Statement

03

Existing
system

04

Proposed
statement

05

Requirement
specifications

06

Design

07

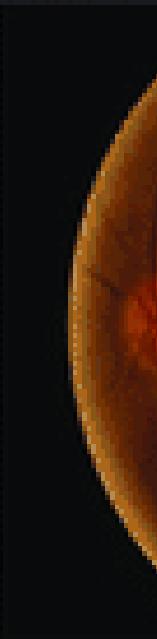
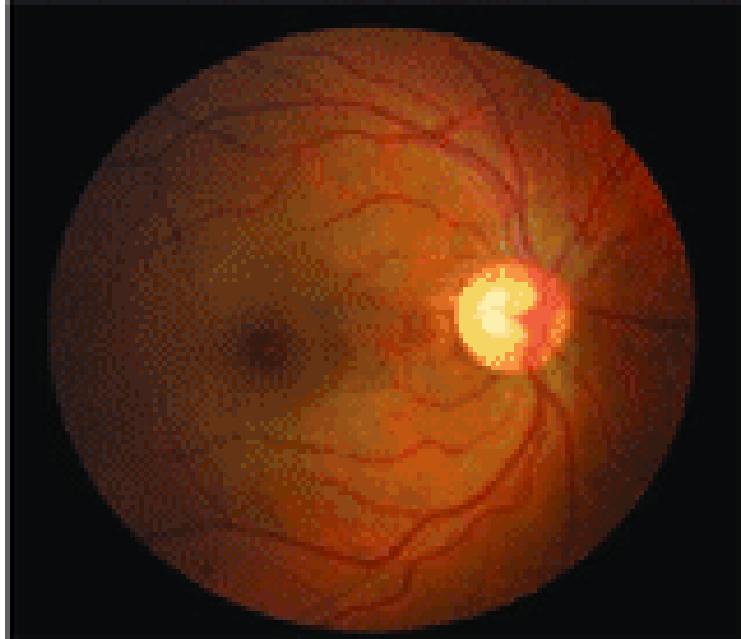
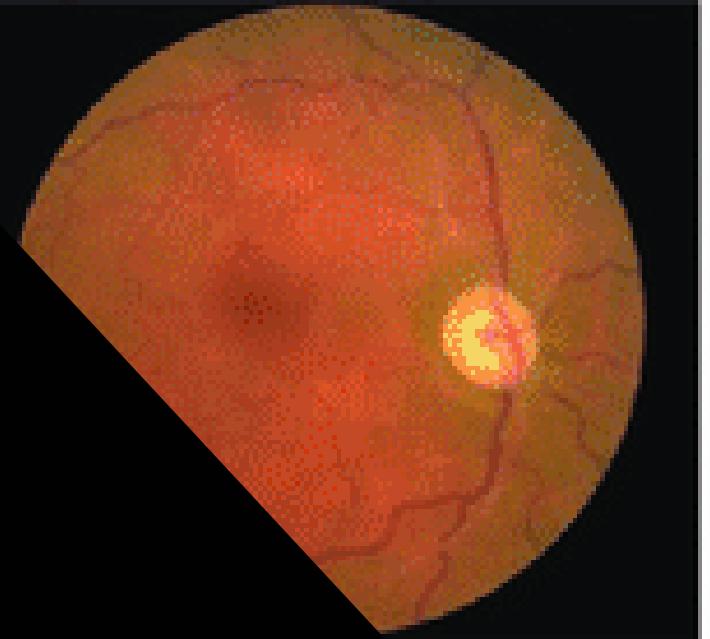
Conclusion

08

Future scope

ABSTRACT

- Diabetic retinopathy (DR) is a chronic eye disease that affects people with diabetes and can lead to vision loss if left untreated. It is a leading cause of blindness in diabetic patients.
- Early detection and treatment can prevent severe vision loss. Machine learning algorithms have shown great potential in detecting diabetic retinopathy from retinal images.
- The objective of our project is to implement multi class supervised machine learning algorithm that could classify the given input images under multiple classes. The accuracy of the ML model will be evaluated using performance metrics such as sensitivity, specificity, and area under the receiver operating characteristic curve.



PROBLEM STATEMENT

- Diabetic Retinopathy is a global health problem.
- Symptoms of Diabetic Retinopathy include:
 - Blurred vision
 - Difficulty seeing colors
 - Eye floaters
- Without treatment, Diabetic Retinopathy can cause vision loss.
- People may not have early symptoms of Diabetic Retinopathy.
- Having a comprehensive dilated eye exam at least once a year can help:
 - Catch the condition early
 - Prevent complications
- The project aim is to:
 - Identify the presence of Diabetic Retinopathy
 - Determine the severity of the disease.

EXISTING SYSTEM

- Existing System:**

Currently Diabetic retinopathy is being diagnosed with a comprehensive dilated eye exam. For this exam, drops are placed in your eyes to widen (dilate) your pupils. With your eyes dilated, your doctor takes pictures of the inside of your eyes. Then your doctor will inject a special dye into your arm vein and take more pictures as the dye circulates through your eyes blood vessels.

- Drawbacks:**

The above mentioned method is complicated and does not guarantee accuracy

PROPOSED SYSTEM

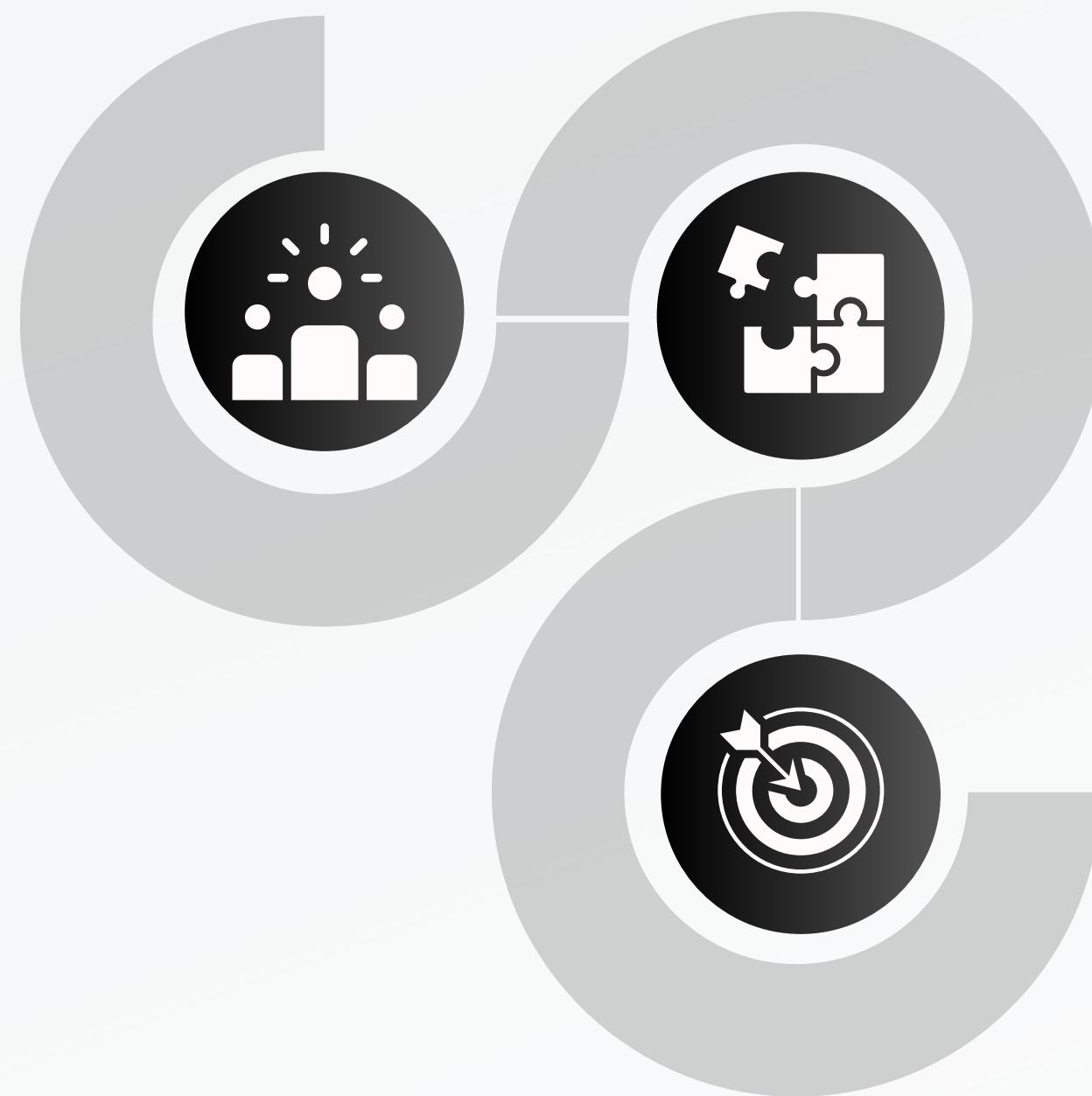
01 • In this project Retinal image undergo preprocessing, then they further go through the different CNN layers.

.

02 • The initial layers identifies features like edges, curves etc. i.e it is responsible for recognizing the edge properties of fundus image.

03 • The last convolution layer learns the features for classification of fundus images into various DR grades like:

- No_DR
- Mild
- Moderate
- Severe
- Proliferative



REQUIREMENT SPECIFICATIONS

Software Requirements:

- Operating system : windows, linux**
- Processor : minimum intel i3**
- Ram : minimum 4 gb**
- Hard disk : minimum 250gb**

Hardware Requirements:

A PC/Computer with

- An intel i3 or higher Processor**
- RAM- 2GB or higher**

DESIGN & SCOPE



Module-1

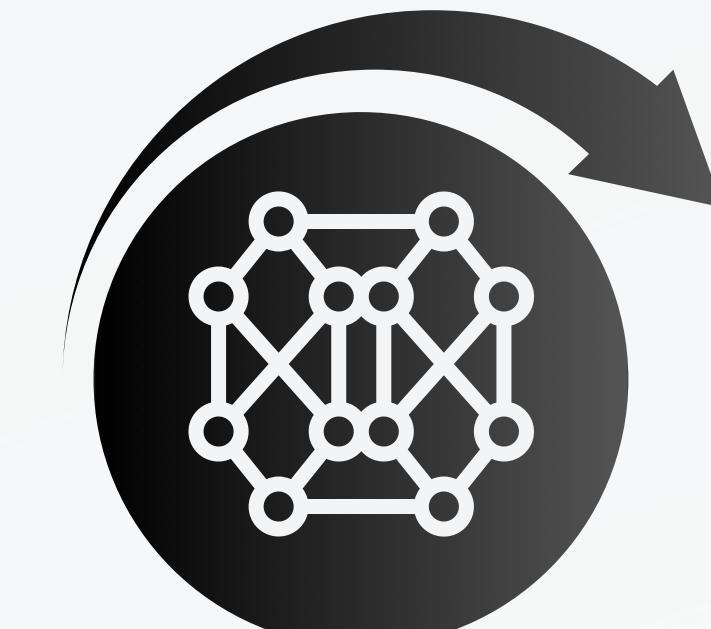
Input to the System

Retinal images are given as input



Module-2
Image Processing

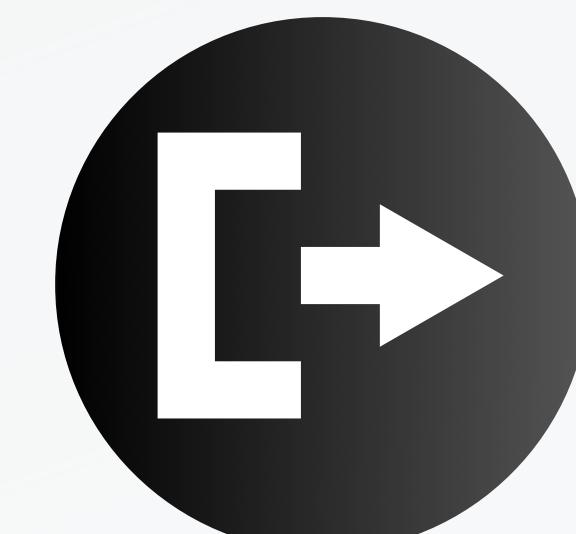
OpenCV is used to manipulate the images and to extract information.



Module-3

CNN Algorithm

Consists of an input layer, 5 hidden layers and a dense layer.



Module-4
Output

Classification of retinal images into 5 categories.

STEPS IN IMAGE PROCESSING

01

Coloring

Converting it into
Grayscale

02

Cropping

Removing
unnecessary
black areas

03

Blurring

Gaussian filter
is applied to
blur

04

Masking

To Enhance the
High frequency
Components

05

Resizing

Resize the
image to 224
*224 pixels

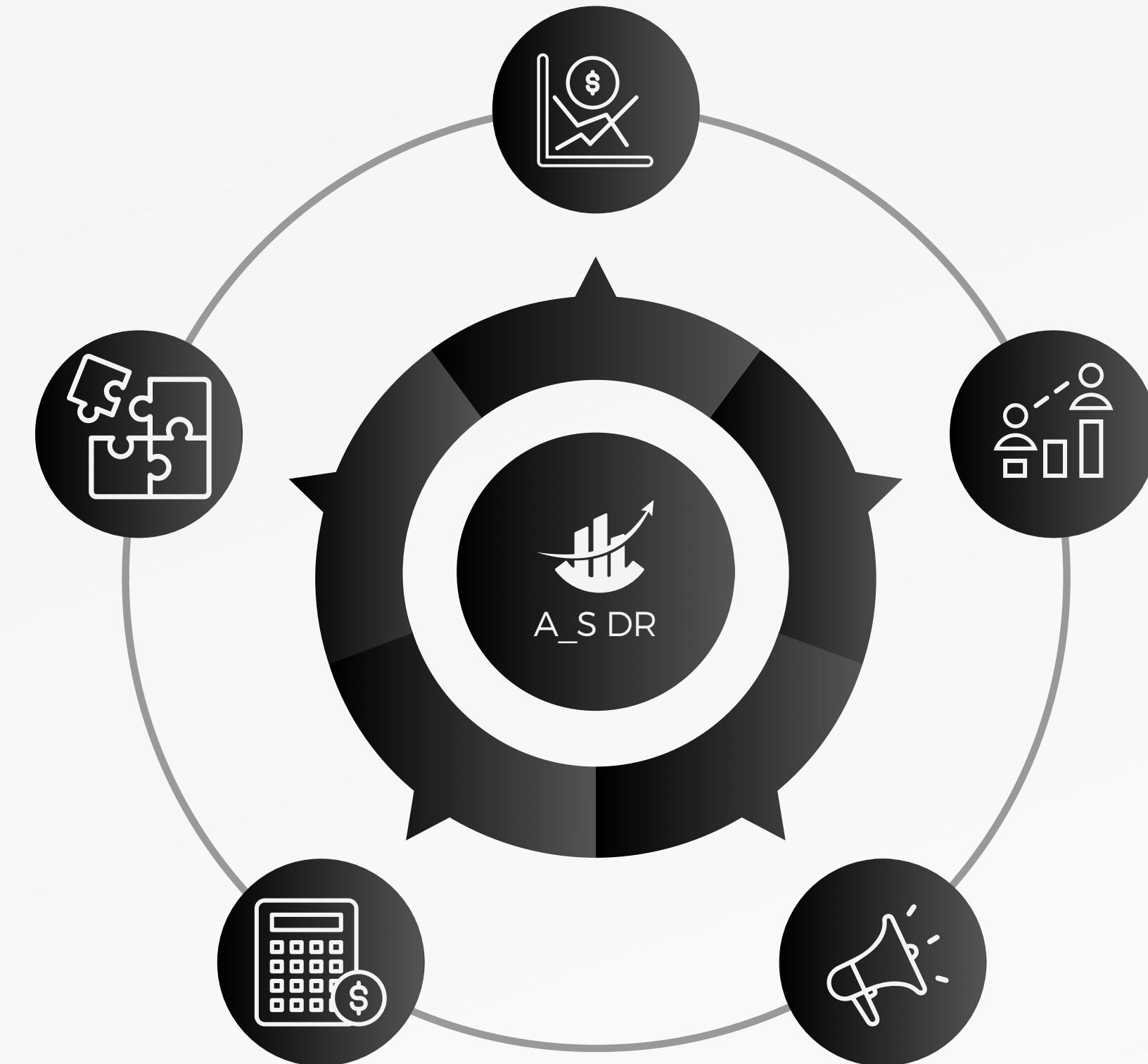
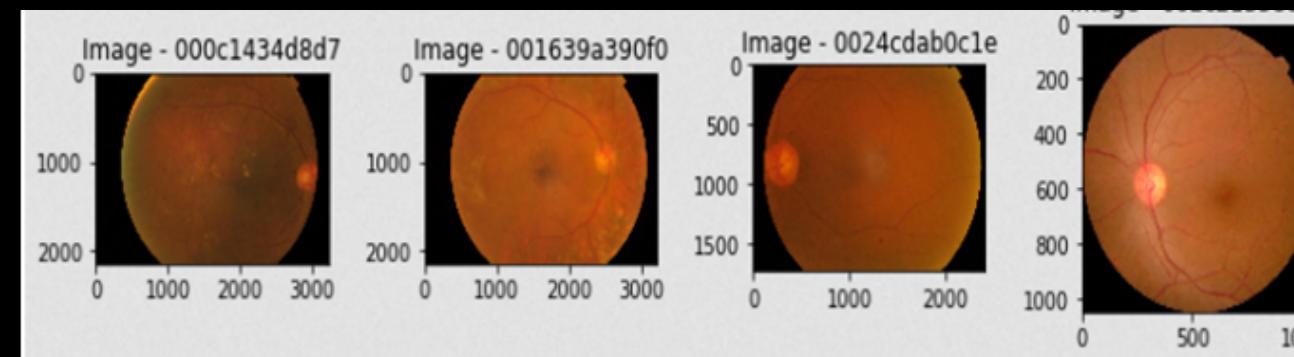
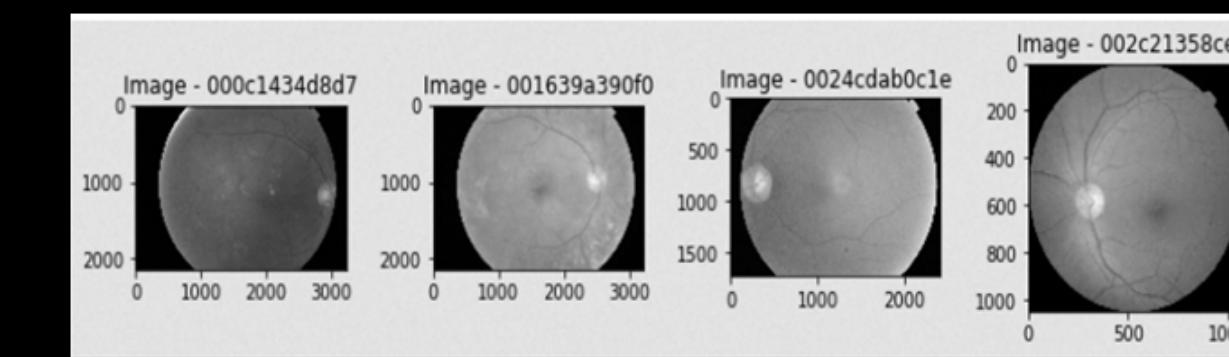


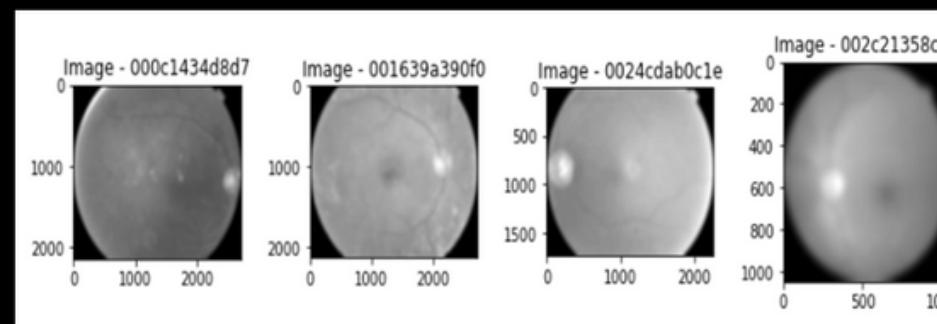
IMAGE PRE PROCESSING



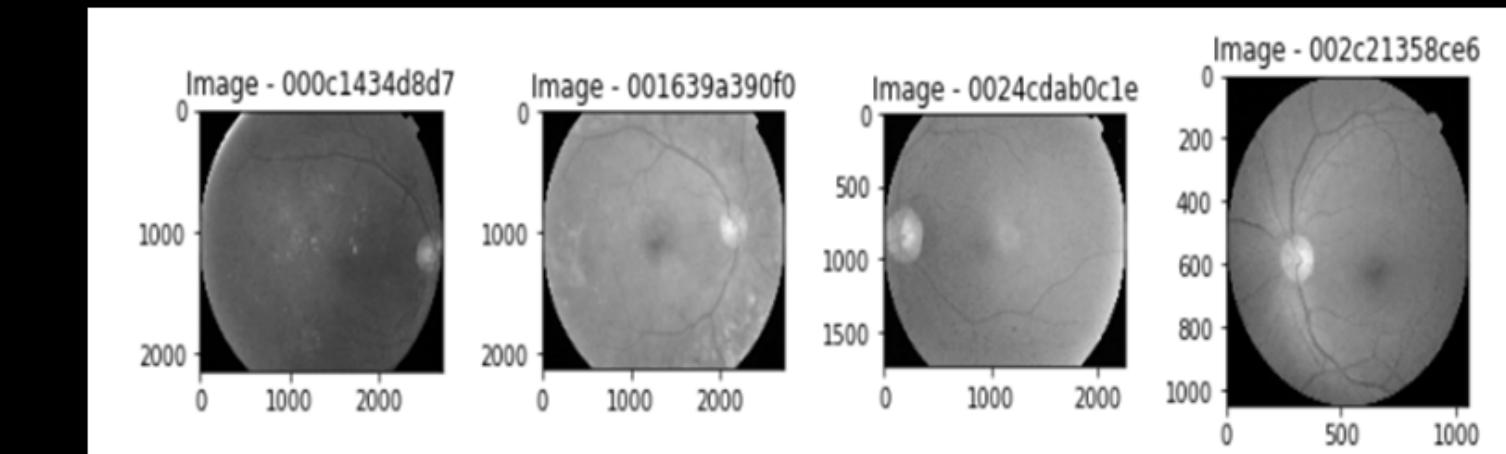
Input images before preprocessing



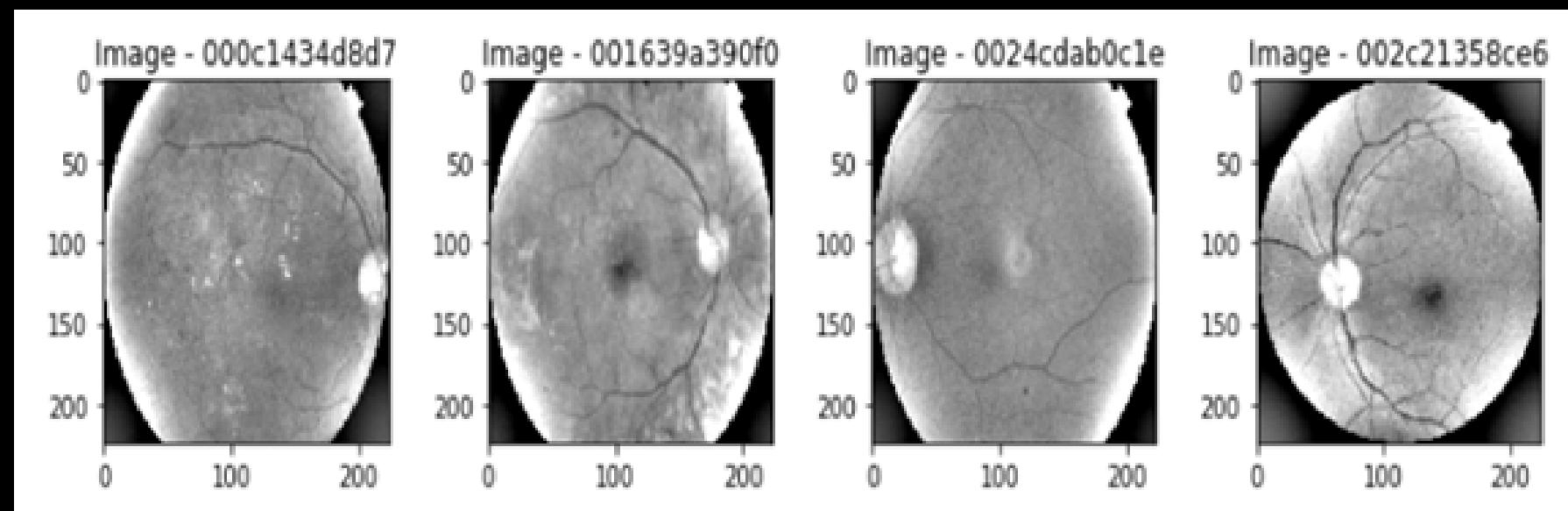
Input images after converted to grayscale



Images after cropping



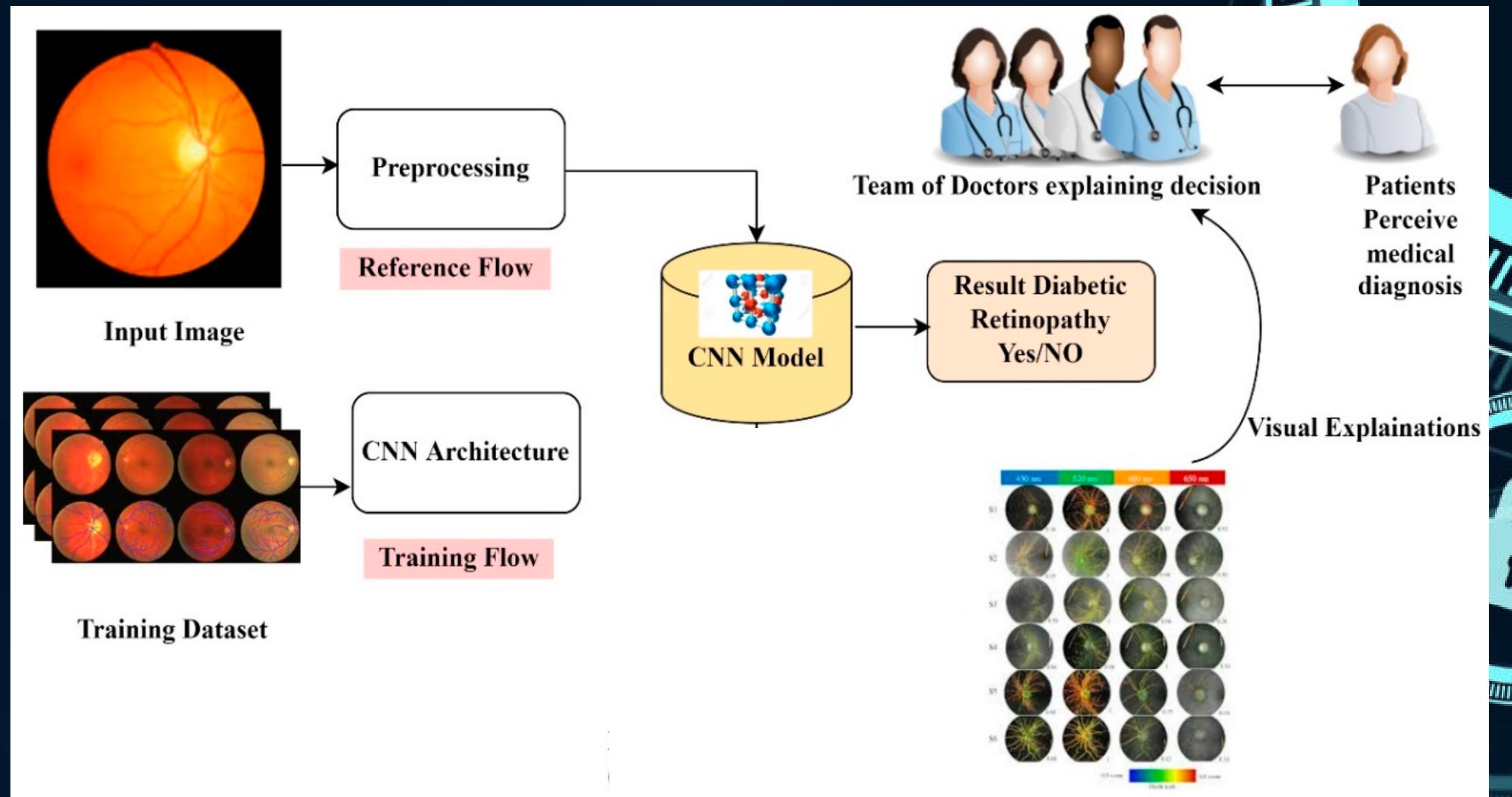
Input images after Gaussian Filtering



Final images after Image Preprocessing

SYSTEM IMPLEMENTATION

IMPLEMENTATION



ALGORITHM:

1. Dataset:

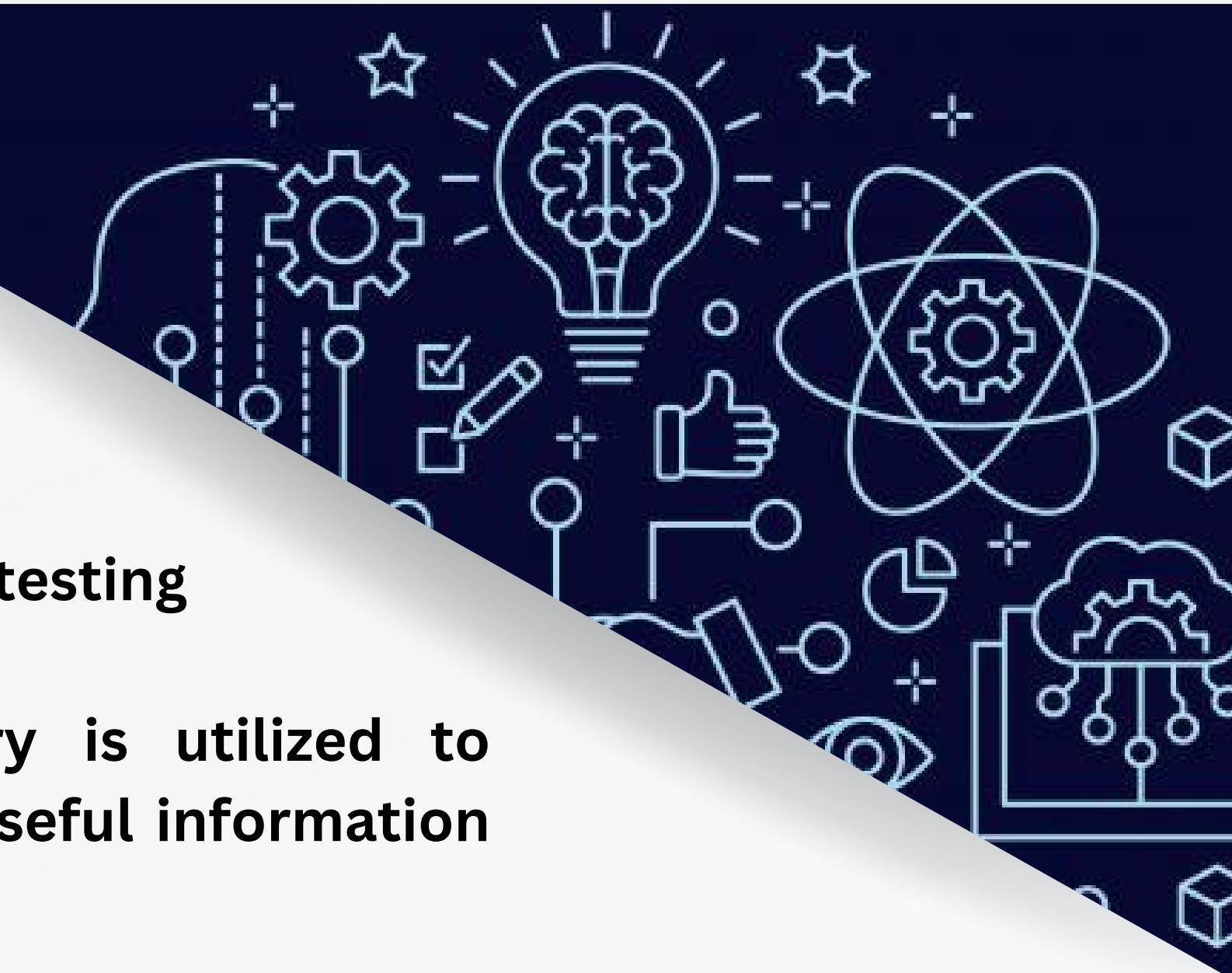
- Consists of 1500 images
- 85% of the images are used for training and 15% for testing

2. Image Preprocessing:

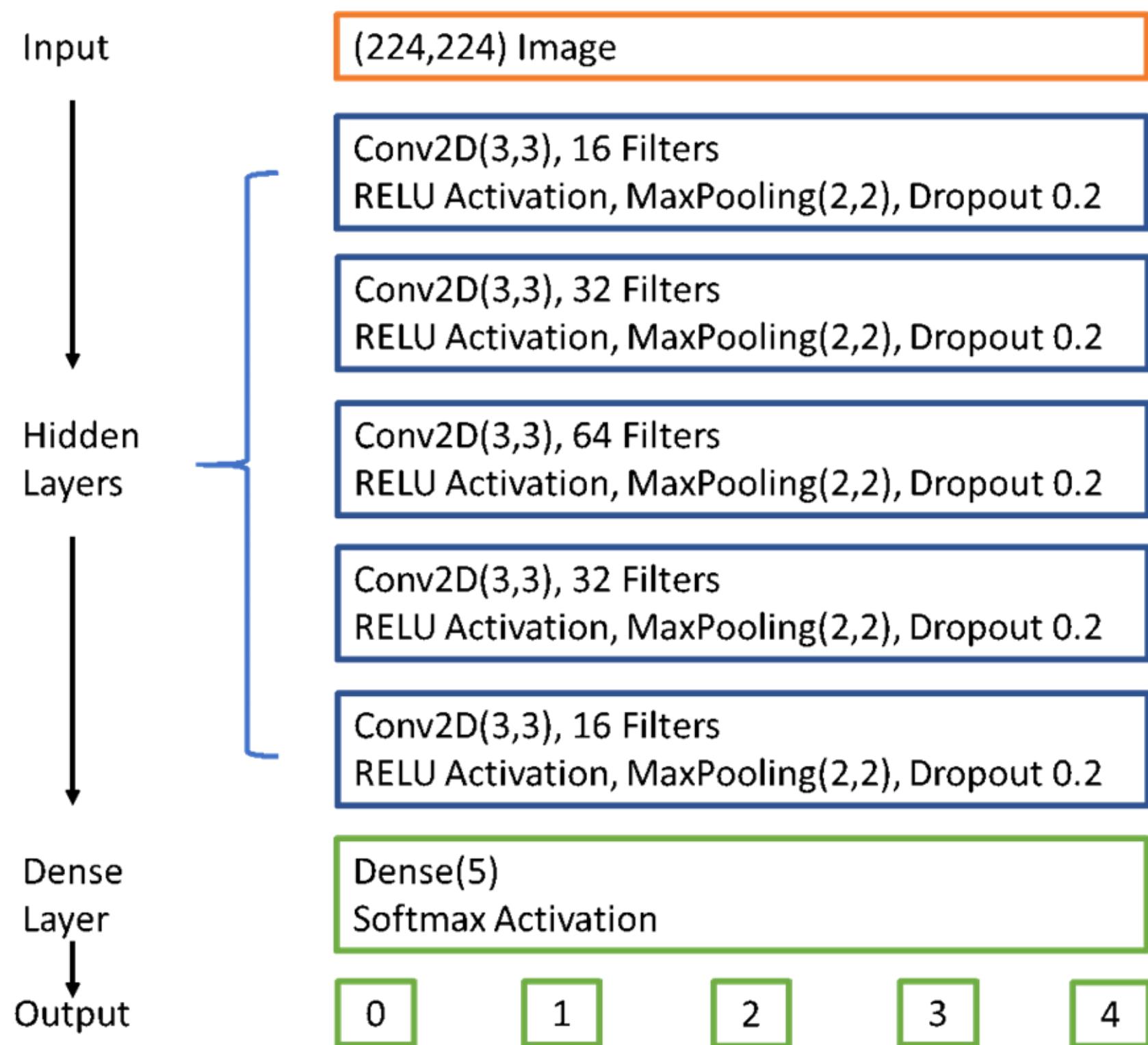
- OpenCV (Open Source Computer Vision) library is utilized to manipulate the pictures in order to extract some useful information from them.

3. Convolutional Neural Network:

- Convolutional neural network (CNN) is used for the classification of retina images into 5 categories.
- This algorithm learn directly from image data, using patterns to classify images and eliminating the need for manual feature extraction.
- Keras library is used to create the CNN model

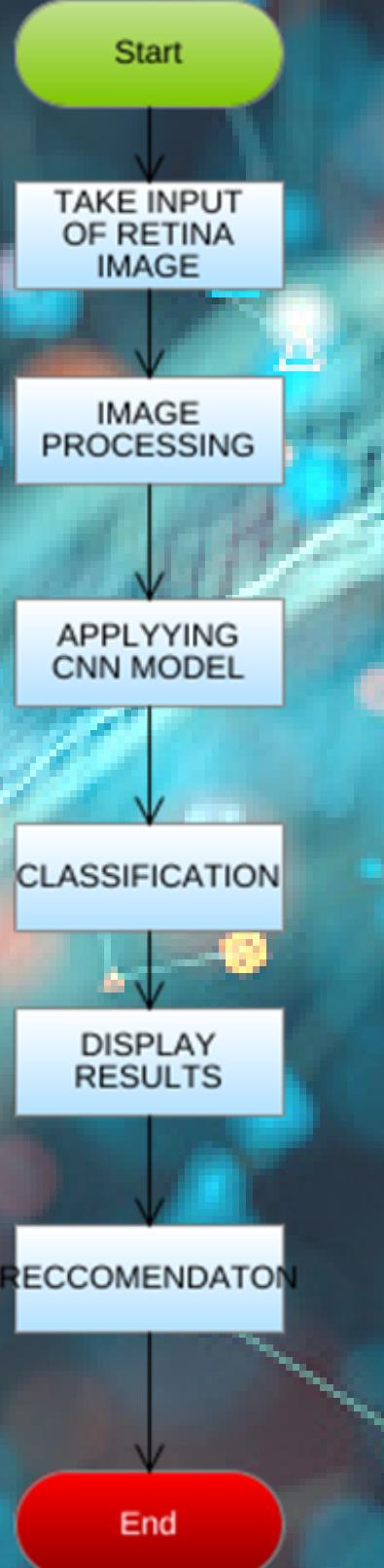


MODEL STRUCTURE:



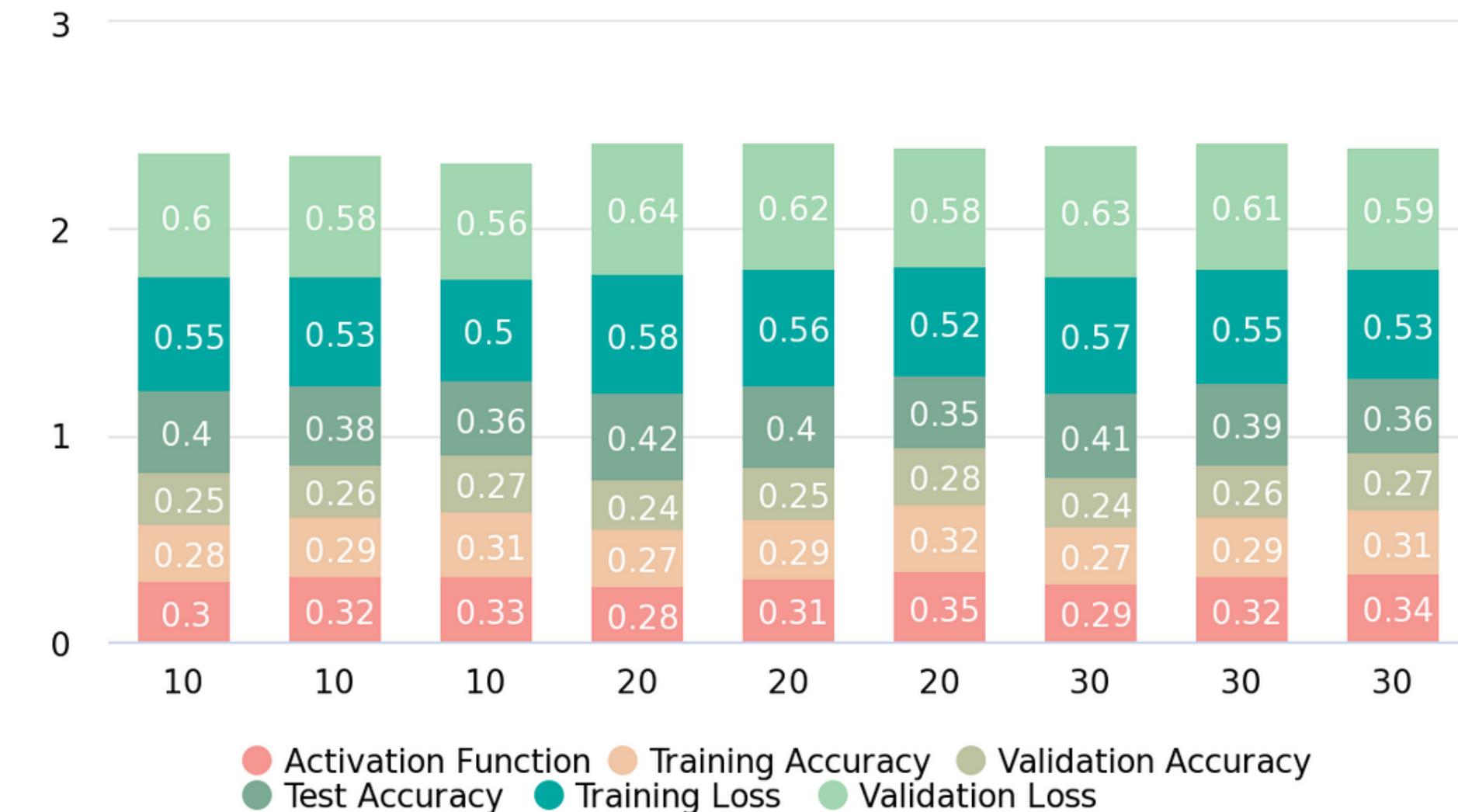
- Model consists of an input layer, 5 hidden layers and a dense layer.
- The hidden layers consist of 3 operations; Convolution, Activation and Pooling. Due to overfitting issues we also added Dropout to each hidden layer of our model

DATA FLOW DIAGRAM

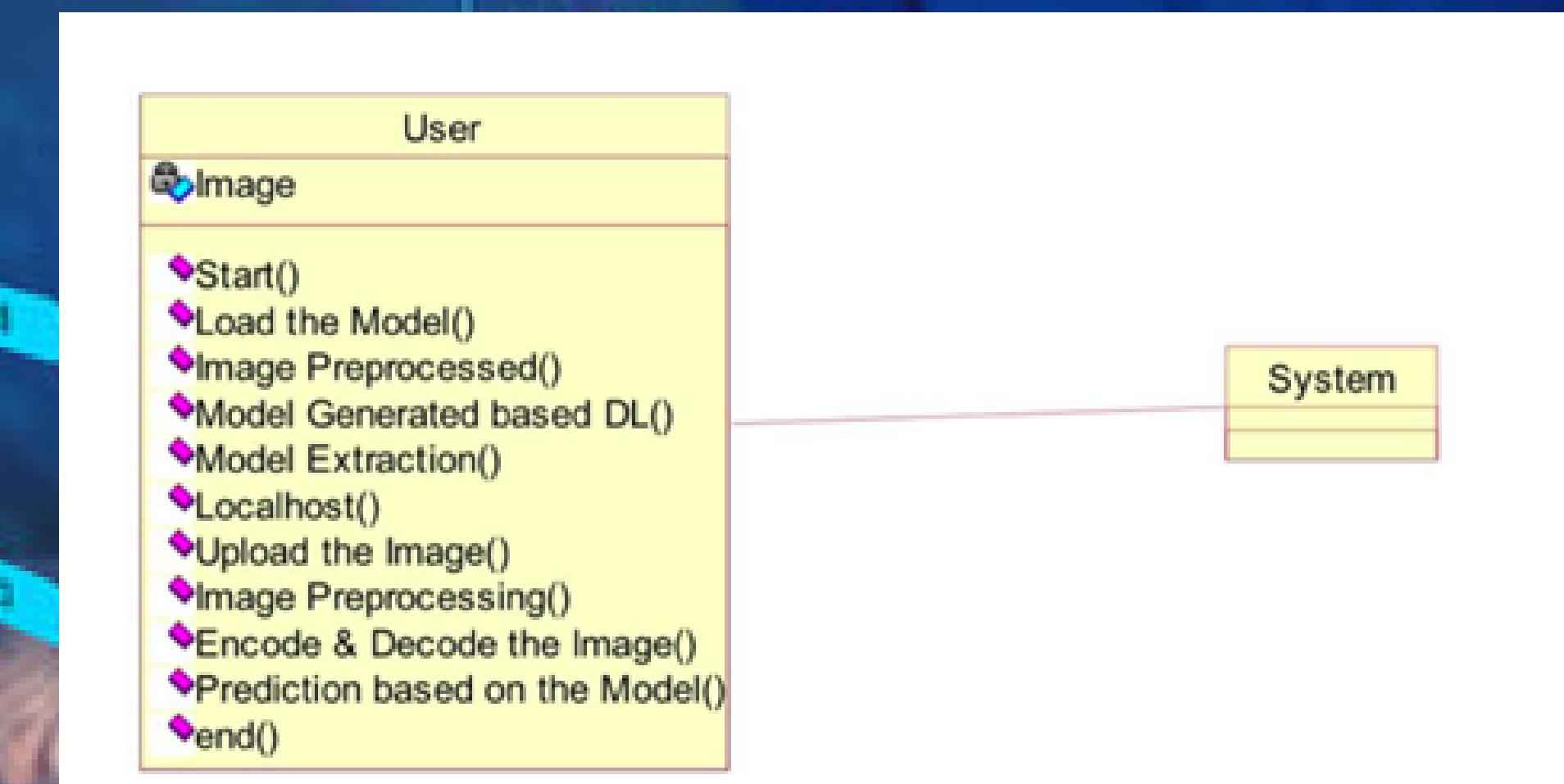
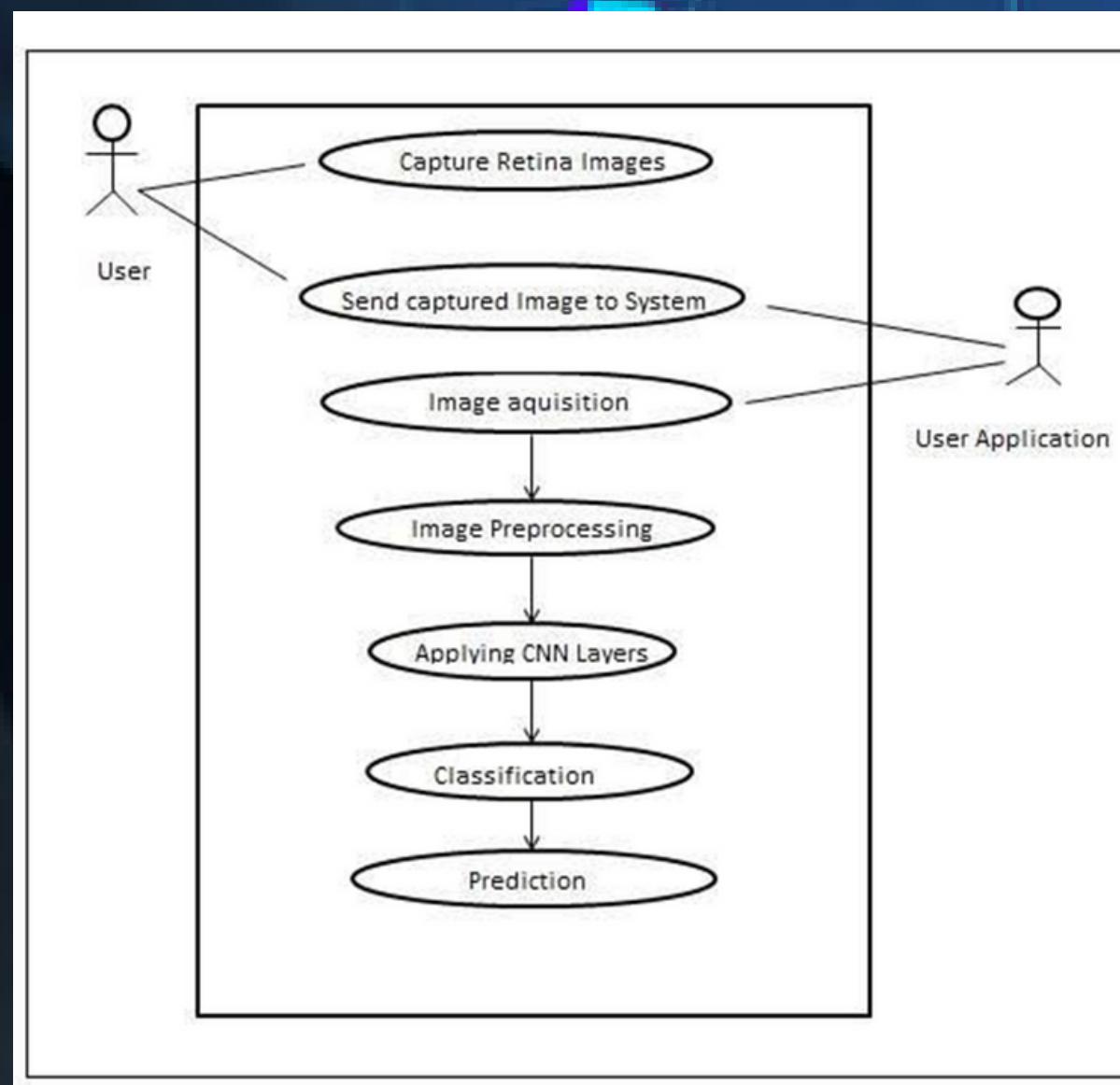


1. Start: The flowchart begins.
2. User Uploads Image: The user uploads an retina image for Diabetic Retinopathy detection.
3. Image Preprocessing: The uploaded image undergoes preprocessing using OpenCV techniques. This may include resizing, noise reduction, contrast enhancement, and normalization.
4. Apply CNN Model: The preprocessed image is input to a pre-trained Convolutional Neural Network (CNN) model specifically designed for Diabetic Retinopathy detection.
5. Diabetic Retinopathy Classification: The CNN model analyzes the image and classifies the severity of Diabetic Retinopathy. The classification may include stages such as mild, moderate, severe, or normal
- . 6. Display Results: The classification results are displayed to the user, indicating the detected severity level of Diabetic Retinopathy.
7. Reccomendation: If the severity level is high or further examination is required, the user may choose to refer the results to an Ophthalmologist for expert consultation.
8. End: The flowchart ends.

ACCURACY VS NUMBER OF EPOCHS



UML DIAGRAMS



USECASE DIAGRAM AND CLASS DIAGRAM

IMPORTED MODULES

```
● ● ●  
import numpy  
from keras.datasets import mnist  
from keras.models import Sequential  
from keras.layers import Dense  
from keras.layers import Dropout  
from keras.layers import Flatten  
from keras.layers.convolutional import Convolution2D  
from keras.layers.convolutional import MaxPooling2D  
from keras.utils import np_utils  
import os, cv2  
import numpy as np  
import matplotlib.pyplot as plt  
  
from sklearn.utils import shuffle  
#from sklearn.cross_validation import train_test_split  
from sklearn.model_selection import train_test_split  
from keras import backend as K  
#K.set_image_dim_ordering('th')
```

LOADING DATASET

```
● ● ●  
img_data_list=[]  
labels_list = []  
for dataset in data:  
    img_list=os.listdir("{}//dataset1//{}".format(path,dataset))  
    print ('Loading the images of dataset-'+'{}\n'.format(dataset))  
    label = labels_name[dataset]  
    for img in img_list:  
        print(img)  
        input_img=cv2.imread("{}//dataset1//{}//{}".format(path,dataset,img))  
        #input_img=cv2.cvtColor(input_img, cv2.COLOR_BGR2GRAY)  
  
        input_img_resize=cv2.resize(input_img,(224,224))  
        print(input_img_resize.shape)  
        #input_img= np.expand_dims(input_img_resize, axis=4)  
        img_data_list.append(input_img_resize)  
        labels_list.append(label)
```

MODEL

```
# Defining the model
import keras
input_shape=(224,224,3)
print("input_shape", input_shape)
#Instantiate an empty model
model = Sequential()

# C1 Convolutional Layer
model.add(keras.layers.Conv2D(6, kernel_size=(5, 5), strides=(1, 1), activation='tanh', input_shape=(224,224,3), padding='same'))

# S2 Pooling Layer
model.add(keras.layers.AveragePooling2D(pool_size=(2, 2), strides=(1, 1), padding='valid'))

# C3 Convolutional Layer
model.add(keras.layers.Conv2D(16, kernel_size=(5, 5), strides=(1, 1), activation='tanh', padding='valid'))

# S4 Pooling Layer
model.add(keras.layers.AveragePooling2D(pool_size=(2, 2), strides=(2, 2), padding='valid'))

# C5 Fully Connected Convolutional Layer
model.add(keras.layers.Conv2D(120, kernel_size=(5, 5), strides=(1, 1), activation='tanh', padding='valid'))
#Flatten the CNN output so that we can connect it with fully connected layers
model.add(keras.layers.Flatten())

# FC6 Fully Connected Layer
model.add(keras.layers.Dense(84, activation='tanh'))

#Output Layer with softmax activation
model.add(keras.layers.Dense(num_classes, activation='softmax'))

# Compile the model
model.compile(loss=keras.losses.categorical_crossentropy, optimizer='SGD', metrics=['accuracy'])

# Viewing model_configuration

model.summary()
```

- C1 Convolutional Layer:** This layer applies convolution to the input images with 6 filters, each having a kernel size of 5x5 pixels. The activation function used is "tanh", and "same" padding is applied to maintain the spatial dimensions of the input.
- S2 Pooling Layer:** This layer performs average pooling with a pool size of 2x2 pixels and a stride of 1x1. The padding is set to "valid", which means no padding is applied.
- C3 Convolutional Layer:** Similar to the C1 layer, this layer applies convolution with 16 filters and a kernel size of 5x5 pixels. The activation function used is "tanh", and "valid" padding is applied.
- S4 Pooling Layer:** Another average pooling layer is added with a pool size of 2x2 pixels and a stride of 2x2. Again, "valid" padding is used.
- C5 Fully Connected Convolutional Layer:** This layer applies convolution with 120 filters and a kernel size of 5x5 pixels. The activation function used is "tanh", and "valid" padding is applied.
- Flatten layer:** This layer flattens the output of the previous layer, converting it from a 3D tensor to a 1D vector, so it can be connected to fully connected layers.
- FC6 Fully Connected Layer:** This layer has 84 neurons and uses the "tanh" activation function.
- Output Layer:** The final layer of the model is a dense layer with the number of neurons equal to the number of classes in the classification task. The activation function used is "softmax", which produces a probability distribution over the classes.

FRONT-END

The screenshot shows a web application titled "Diabetic Retinopathy Detection". On the left, there is a sidebar with the heading "INSTRUCTIONS" containing two numbered steps: "1. Upload Image:" and "2. View Uploaded Image:". Step 1 includes a list of instructions: "Look for the "Upload an image" section.", "Click on the "Browse Files" button.", and "Choose an image file (JPG, JPEG, or PNG) from your computer.". Step 2 is listed below. The main content area features a title "STAGES OF DIABETIC RETINOPATHY" above five circular diagrams illustrating the progression of the disease. The stages are labeled: Normal, Mild, Moderate, Severe, and Proliferative. Each diagram shows increasing signs of damage to the retina, such as microaneurisms, new blood vessels, and bleeding. Below this is a "Drag and drop file here" input field with a "Browse files" button. A note states "Limit 200MB per file • JPG, JPEG, PNG". At the bottom of the main content area, there is a small thumbnail of a retina image.

The screenshot shows the same web application after an image has been uploaded. The main content area now displays the "Classification Result" with the text "Predicted Class: Proliferate_DR". Below this, there is a section titled "Precautions" with the instruction "Seek immediate medical attention. Follow the advice of an eye specialist.". On the left, there is a sidebar with the heading "INSTRUCTIONS" containing two numbered steps: "1. Upload Image:" and "2. View Uploaded Image:". Step 1 includes a list of instructions: "Look for the "Upload an image" section.", "Click on the "Browse Files" button.", and "Choose an image file (JPG, JPEG, or PNG) from your computer.". Step 2 is listed below. The main content area also includes a "SAMPLE IMAGE" placeholder and "Preview" and "Classify" buttons. At the bottom of the main content area, there is a small thumbnail of a retina image.

RESULT OUTPUTS

FRONTEND

SAMPLE IMAGE

Preview

Classify

Classification Result

Predicted Class: Severe

Precautions

Consult an eye specialist immediately. Follow their instructions.

SAMPLE IMAGE

Preview

Classify

Classification Result

Predicted Class: Mild

Precautions

Visit an eye specialist regularly.

CONCLUSION

- In most developing nations, there are limited numbers of specialists to examine diabetic retinopathy in screening programs. Hence, there is an immediate need for developing an automated screening system for diabetic retinopathy diagnosis with accuracy.

FUTURE SCOPE

- **Data Augmentation:** Implement techniques like rotation, scaling, and flipping to augment your training data. This can help improve the model's ability to generalize to new and unseen images.
- **Transfer Learning:** Explore pre-trained models such as VGG, ResNet, or Inception, and leverage transfer learning to initialize your model with their learned features. Fine-tuning the pre-trained models on your specific dataset can lead to better performance and faster convergence.
- **Ensemble Learning:** Build an ensemble of multiple CNN models with different architectures or trained on different subsets of the data. Combine their predictions to make final predictions, which often leads to improved performance and robustness.

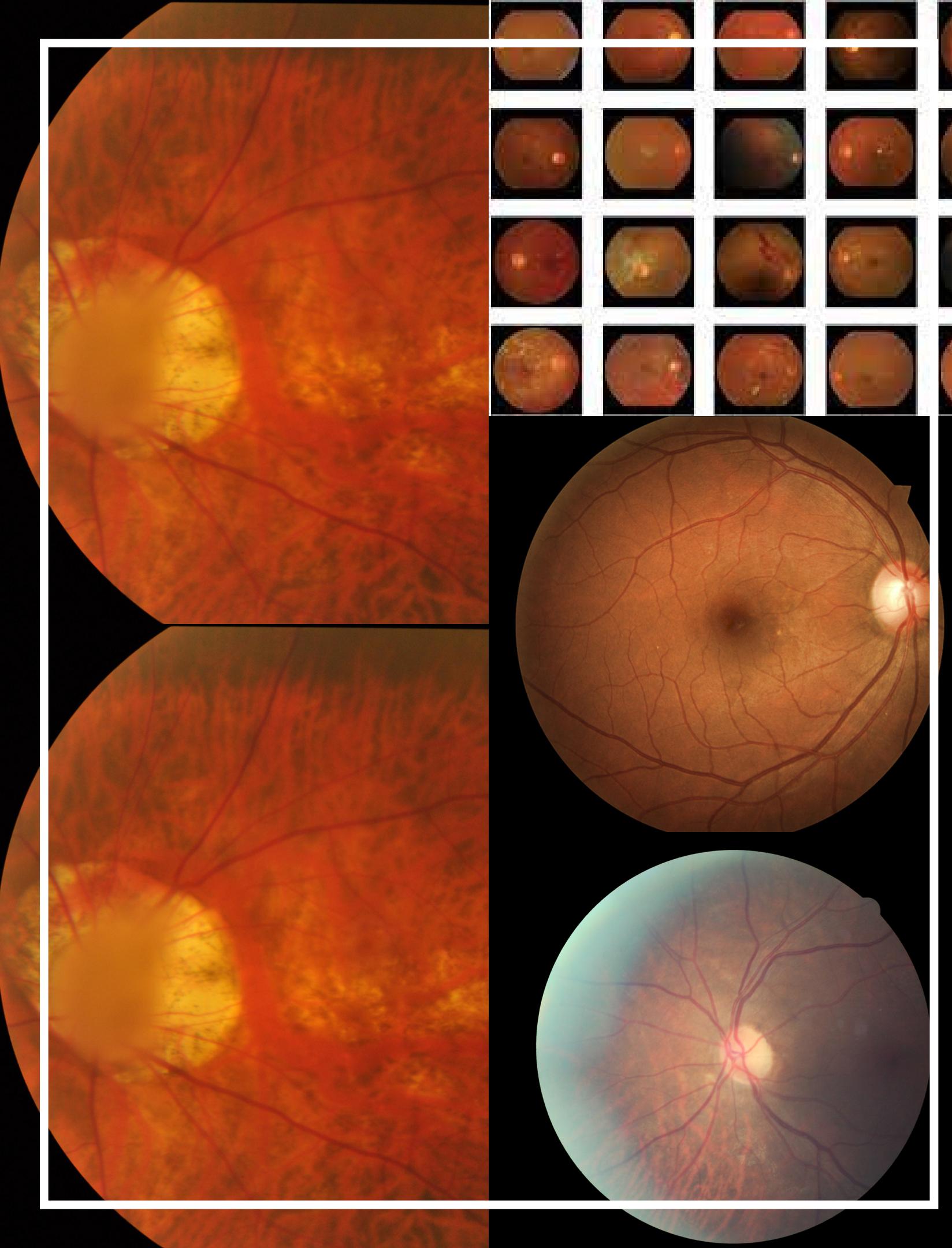
REFERENCES

1. R. Taylor, D. Batey **Handbook of retinal screening in diabetes:diagnosis and management (seconded.)**, John Wiley & Sons, Ltd Wiley-Blackwell (2012)
2. International diabetes federation - what is diabetes [Online] .<https://www.idf.org/aboutdiabetes/what-is-diabetes.html>
3. American academy of ophthalmology-what is diabetic retinopathy? [Online]. <https://www.aoa.org/eye-health/diseases/what-is-diabetic-retinopathy>
4. R.R. Bourne, et al. Causes of vision loss worldwide, 1990-2010: a systematic analysis Lancet Global Health, 1 (6) (2013), pp. 339-349
5. C.A. Harper, J.E. Keeffe **Diabetic retinopathy management guidelines** Expert Rev Ophthalmol, 7 (5) (2012), pp.
6. E. T. D. R. S. R. GROUP **Grading diabetic retinopathy from stereoscopic color fundus photographs- an extension of the modified Airlie House classification** Ophthalmology, 98 (5) (1991), pp. 786-806
7. M. Dubow, et al. **Classification of human retinal microaneurysms using adaptive optics scanning light ophthalmoscope fluorescein angiography** Invest Ophthalmol Vis Sci, 55 (3) (2014), pp. 1299-1309
CrossRefView Record in Scopus
8. F. Bandello, M.A. Zarbin, R. Lattanzio, I. Zucchiatti **Clinical strategies in the management of diabetic retinopathy (second ed.)**, Springer (2019)



TEAM-13

Questions?



**THANK
YOU**

