VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Content Based Image and Video Retrieval

CSE-3018

J Component Report

Title: Traffic Sign Detection System using Convolutional Neural Network.

Faculty: Dr. GEETHA S

Team Members:

- **VISHWANATH B BHAT 19BCE1499**
- **LIKITH M D 19BCE1668**
- **GAUTAM SRINIVASAN 19BCE1643**

Contents:

Abstract:

In this Project, we are going to implement the Traffic Sign Detection System using Convolutional Neural Network model and Keras because CNN is best for image classification purposes. Using this model, we extract features of the traffic sign images and classify them into different categories. We are going to select an image from the database and the classified result depending on the image shall be shown on the GUI created using python programming language.

Introduction:

A Content-Based Image Retrieval system is an application of computer vision techniques which is used for searching digital images from large databases which would be difficult to search manually due to the large size of the databases. In CBIR, the features (shapes, textures, colors or other information) extracted from the images shall be used for search analysis rather than the metadata such as keywords, descriptions or tags associated with the image.

A Convolutional Neural Network (CNN) is a type of artificial neural network which uses deep learning algorithms for image and video classification. It takes an input image, assigns weight and biases to various aspects in the image and can differentiate one another. CNNs have neurons arranged in such a way like that of the frontal lobe of the humans which is responsible for the processing of visual stimuli. It uses a system like a multi-layer perceptron which is designed for reducing the processing. It consists of an input, output and a hidden layer which includes multiple convolutional layers, pooling layers, fully connected layers and normalization layers.

The below figure shows the architecture of the Traffic Sign Detection System with a Convolutional Neural Network.

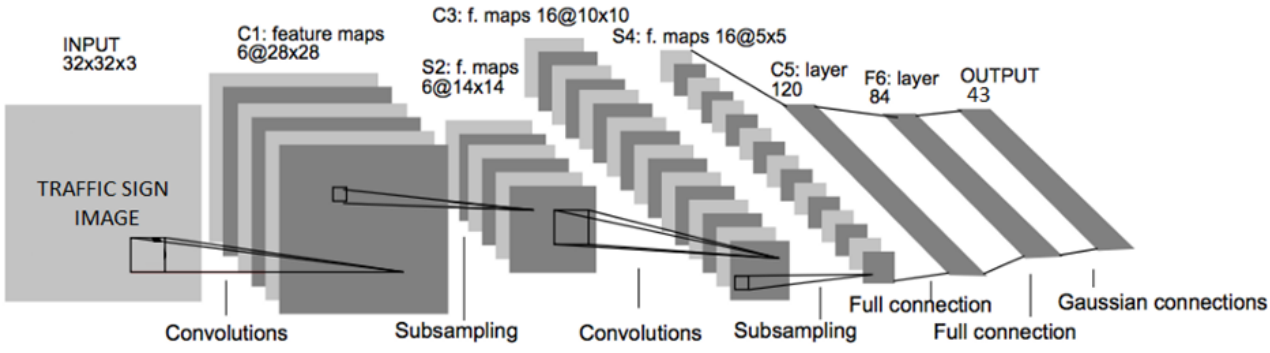


Figure: CNN Model for Traffic Sign Detection System.

Related Works:

Research Paper-1:

Real Time Detection and Reognition of Indian Traffic Signs using Matlab

There is an automatic in-vehicle computing device which receives video that has to be processed with the help of car cameras. Because of the motion of the vehicle image frames are blurred by Gaussian Noise which is this paper explains about the image enhancement is done using median filter and nonlinear Lucy-Richardson for de-convolution.The prototype proposed in this system can be summarized as:

1) Image extraction from the video. 2) Image enhancement for different lighting and whether conditions. 3) Selecting candidate objects based on colour. 4) Filtering the candidate objects based on shape. 5) Recognising the filtered object using neural network 6) Give visual image of the recognised sign and audio for the type of category of the traffic sign

Steps:

1. Image extraction and Pre-processing

2. Colour Based Segmentation (YCbCr color space is used here and the reason the paper explains for using this color space is to overcome the illumination sensitive characteristic of RGB space)

3. Shape-based filtering.

Research Paper-2:

Real-time Traffic Sign Recognition System

This paper first explains about the lack of time for complex image processing techniques initially and later suggests some known startegies to solve these problems. Some of the general problems that the paper has mentioned is how to detect and to compensate for the change in ambient lighting conditions, including weather changes, daylight, and vehicle turns. Occlusions of traffic signs due to the presence of objects such as trees, buildings, vehicles, pedestrians, or another signs are also important factor to be considered.Ageing and vandalism also affect the image perception.Deterioration or intentional deformations.Architecture proposed for recognition of traffic signs in this paper is:

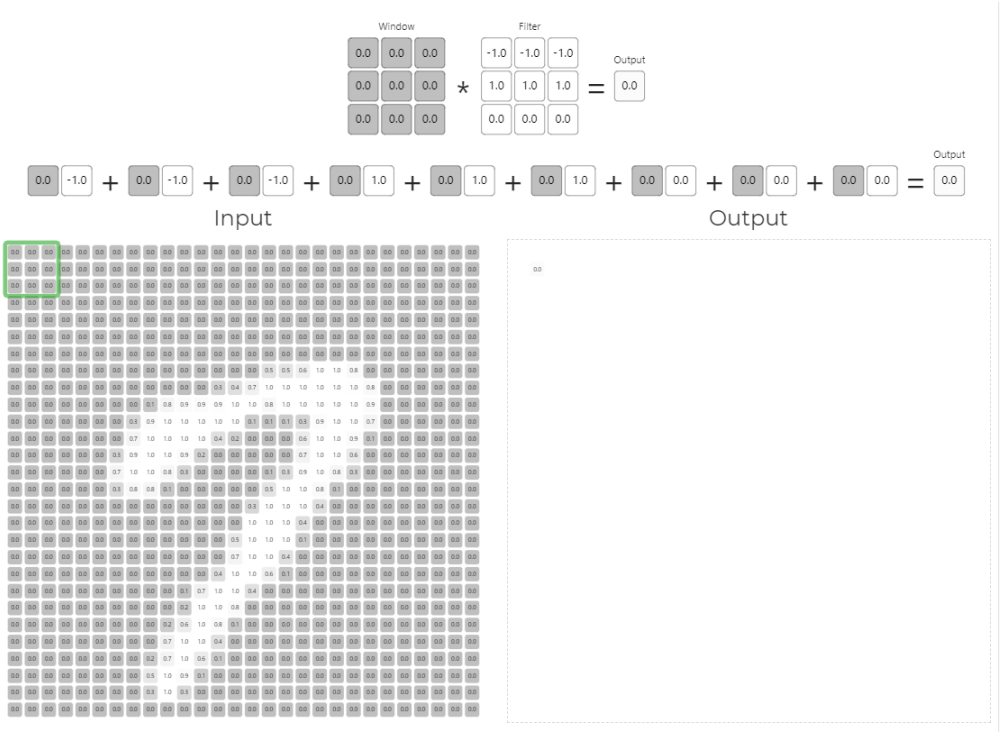
1) color segmentation or adjusment which includes video/image decoding and color space transformations. 2) selection of regions of interests (ROI) where traffic signs are present in the image. 3) identification of traffic signs, i.e. feature extraction techniques used with data mining classifiers. This paper says that the proposed generic system can be used in both personal and mobile computer environments based on a high quality web camera video stream. Finally the overview of this research paper can be concluded as The system integrates color, shape, and motion information. It is built on three main components, one is color acquisition framework,two is accelerated Hough-like transform based ROI identification and then country independent recognition module.

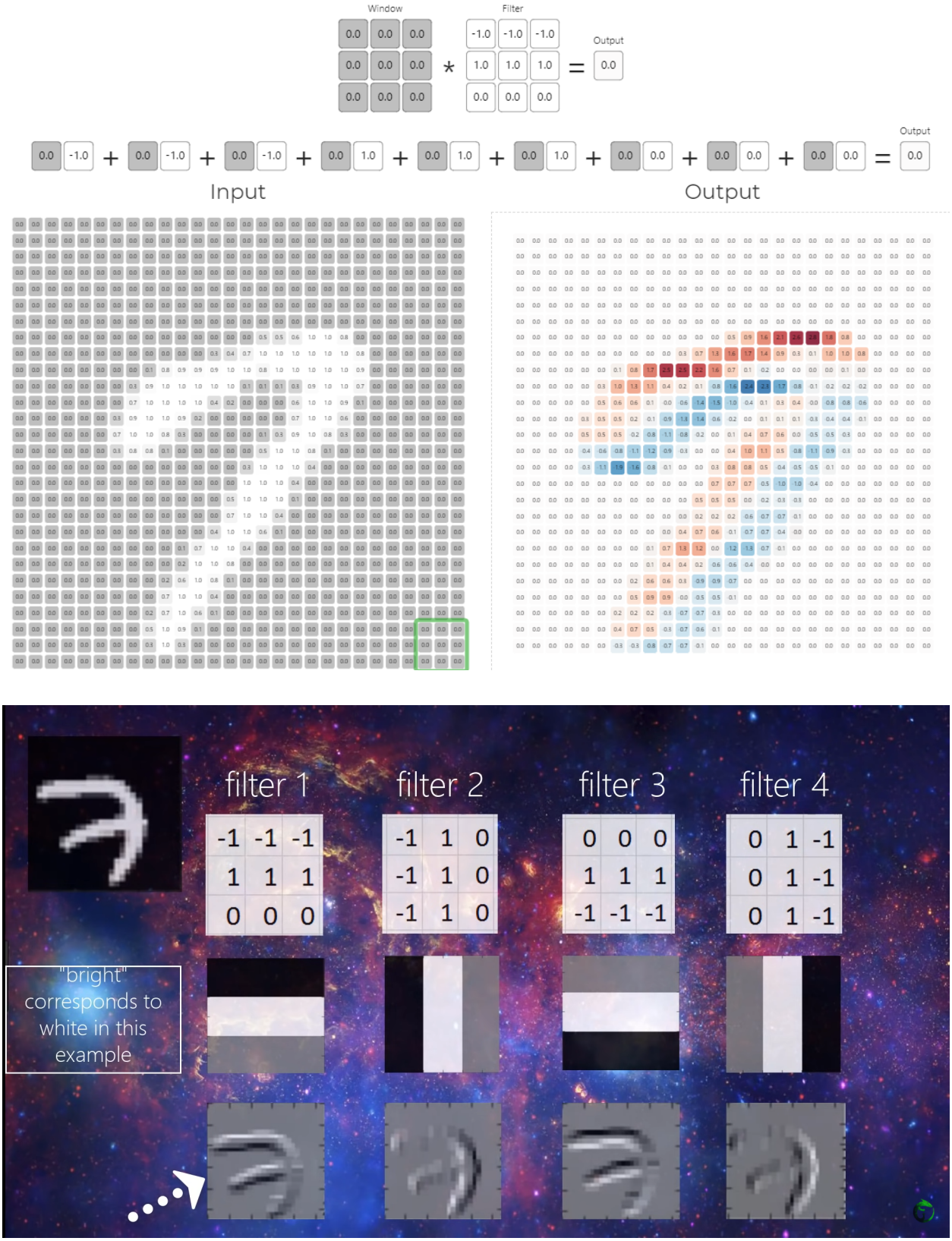
Proposed work:

Based on the research papers we went through and other online resources we narrowed down our options to proceed further by using CNN . The other alternatives were using Random forest classification machine learning algorithm as suggested, But it takes only numerical input so we had to convert every image into statistical feature and then fed it to the model . The achieved accuracy was very poor below 48%.Then we tried using Local Binary pattern to feed as input but still we were not able to cross 63% accuracy.Hence after some suggestions from people we decided to go ahead with CNN since it is known to provide good result for images.

CNN in Images:

CNN is able to identify the patterns[shapes,texture etc] in the image base on multiplying a 3*3 matrix filter with 3*3 pixel box . There is [interactive website] which helps us visualizing this along with different filters . So the more layer we use the more complex pattern will be recorded which will make our model more precise . Below are some of the snapshot on how it works

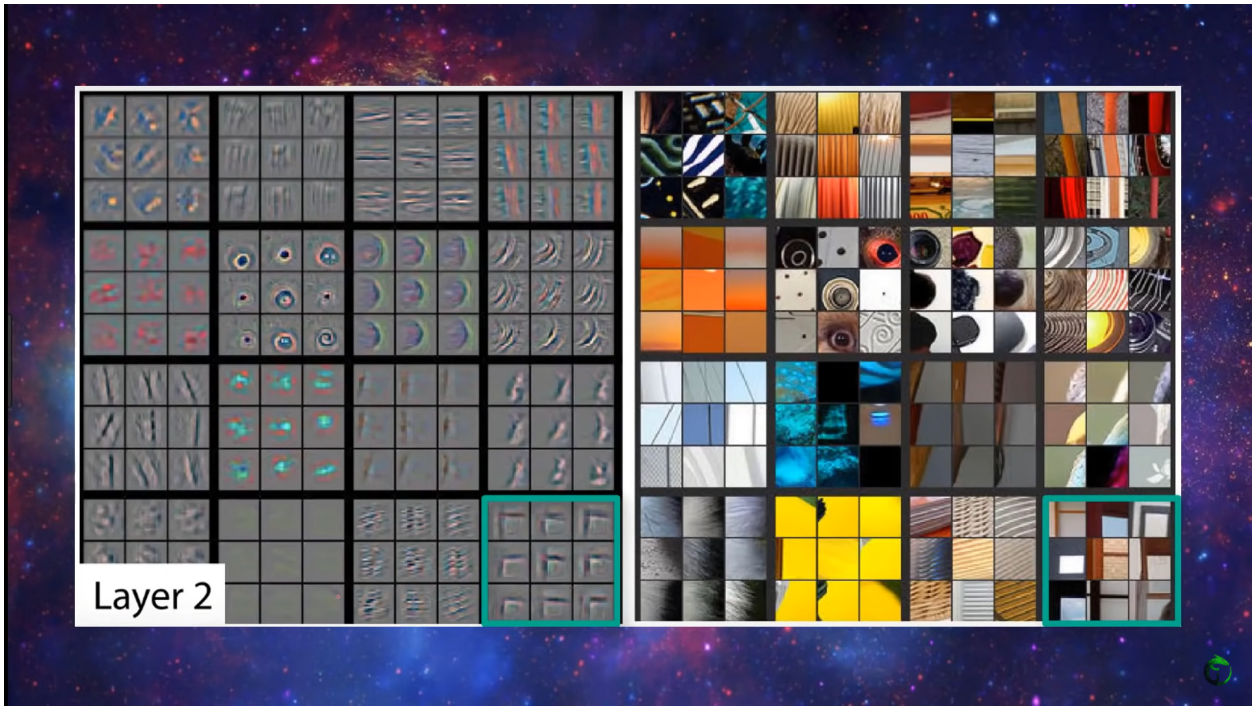




Here in the above diagram we can see different filter gives us different pattern which in put all together gives us a better result. Hence it is always good to use

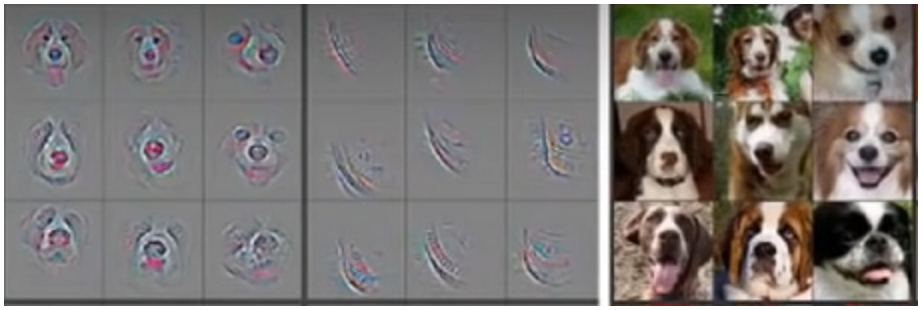
more layer which gives us more patterns of the same image. Just make sure when the accuracy starts to flatten out.

As said earlier in the below diagram we are able to see on how different pattern is being recognised and the recognised pattern are simple in layer 2



But in layer 4 as shown below it is able to detect facial pattern of dogs

LAYER 4



Project Output:

As said earlier we used CNN model to train our model and below is the model summary

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 26, 26, 32)        2432

 conv2d_1 (Conv2D)           (None, 22, 22, 32)        25632

 dropout_1 (Dropout)         (None, 3, 3, 64)          0

 flatten (Flatten)           (None, 576)               0

 dense (Dense)               (None, 256)               147712

 dropout_2 (Dropout)         (None, 256)               0

 dense_1 (Dense)             (None, 43)                11051

=================================================================
Total params: 242,251
Trainable params: 242,251
Non-trainable params: 0
_________________________________________________________________
None
```

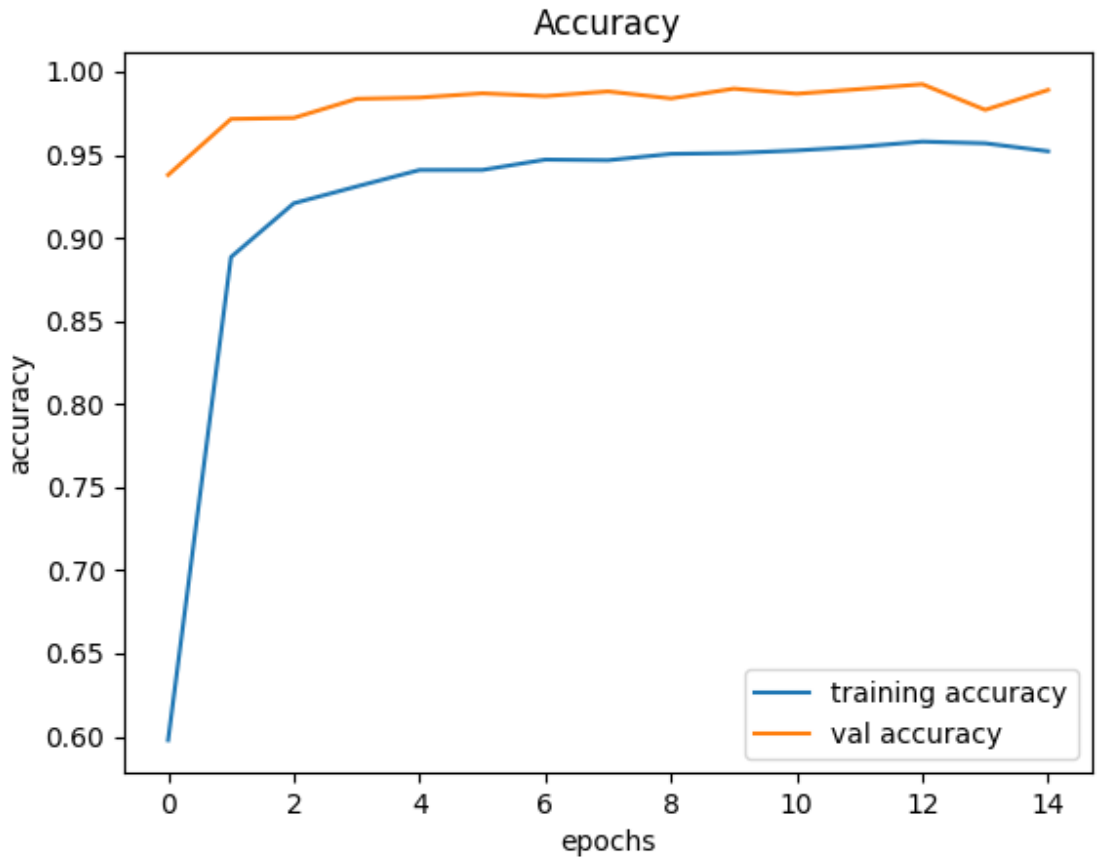
It took us around 18 minutes to train the model on gtx 1650(4gb) system for 15 epochs.

```
Epoch 1/15
981/981 [==============================] - 43s 43ms/step - loss: 2.1841 - accuracy: 0.4322 - val_loss: 0.4827 - val_accuracy: 0.8676
Epoch 2/15
981/981 [==============================] - 38s 39ms/step - loss: 0.6564 - accuracy: 0.8075 - val_loss: 0.1434 - val_accuracy: 0.9630
Epoch 3/15
981/981 [==============================] - 39s 39ms/step - loss: 0.4059 - accuracy: 0.8826 - val_loss: 0.0980 - val_accuracy: 0.9750
Epoch 4/15
981/981 [==============================] - 39s 39ms/step - loss: 0.3189 - accuracy: 0.9097 - val_loss: 0.1359 - val_accuracy: 0.9602
Epoch 5/15
981/981 [==============================] - 38s 39ms/step - loss: 0.2766 - accuracy: 0.9222 - val_loss: 0.0903 - val_accuracy: 0.9746
Epoch 6/15
981/981 [==============================] - 38s 39ms/step - loss: 0.2379 - accuracy: 0.9340 - val_loss: 0.0991 - val_accuracy: 0.9745
Epoch 7/15
981/981 [==============================] - 38s 39ms/step - loss: 0.2444 - accuracy: 0.9347 - val_loss: 0.0536 - val_accuracy: 0.9855
Epoch 8/15
981/981 [==============================] - 39s 40ms/step - loss: 0.2274 - accuracy: 0.9398 - val_loss: 0.0472 - val_accuracy: 0.9871
Epoch 9/15
```

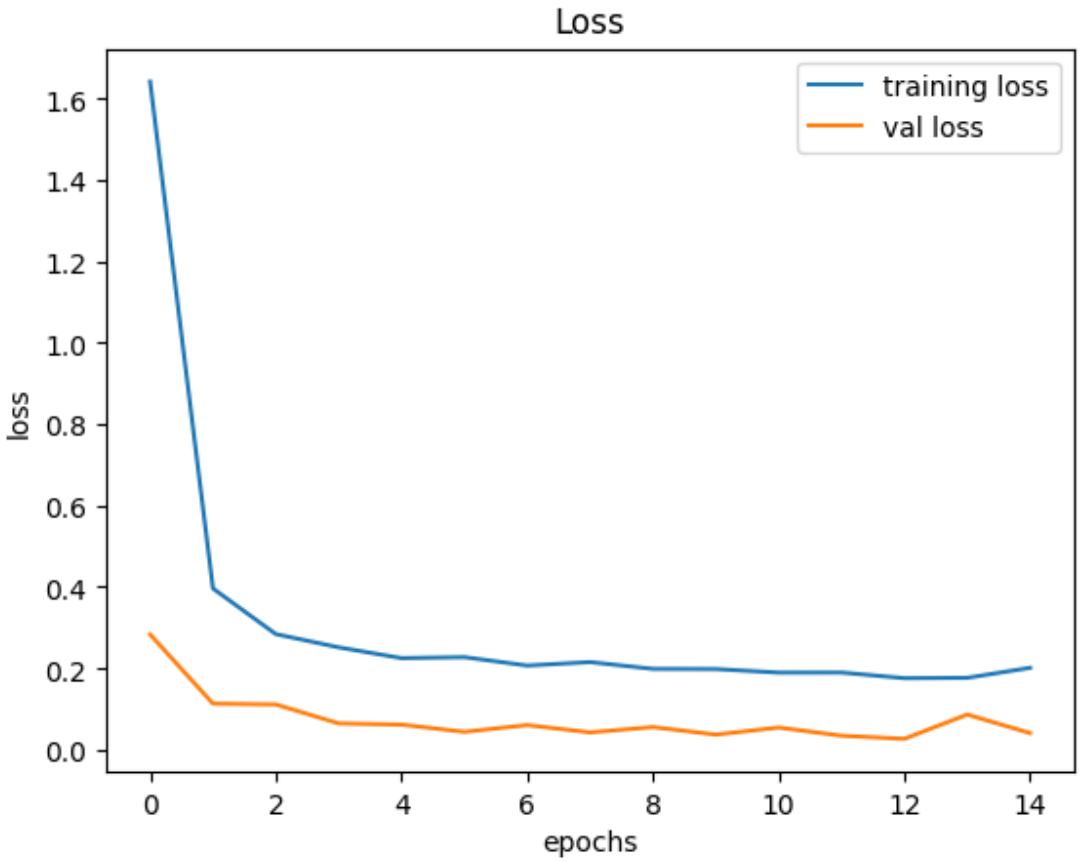
```
Epoch 12/15
981/981 [==============================] - 38s 39ms/step - loss: 0.1552 - accuracy: 0.9583 - val_loss: 0.0415 - val_accuracy: 0.9889
Epoch 13/15
981/981 [==============================] - 38s 39ms/step - loss: 0.1828 - accuracy: 0.9516 - val_loss: 0.0739 - val_accuracy: 0.9797
Epoch 14/15
981/981 [==============================] - 38s 39ms/step - loss: 0.1989 - accuracy: 0.9506 - val_loss: 0.0909 - val_accuracy: 0.9750
Epoch 15/15
981/981 [==============================] - 38s 39ms/step - loss: 0.1494 - accuracy: 0.9610 - val_loss: 0.0526 - val_accuracy: 0.9881
0.9584323040380047

(traffic_sign) D:\Projects\5th sem\cbir2\Traffic sign classification>
```

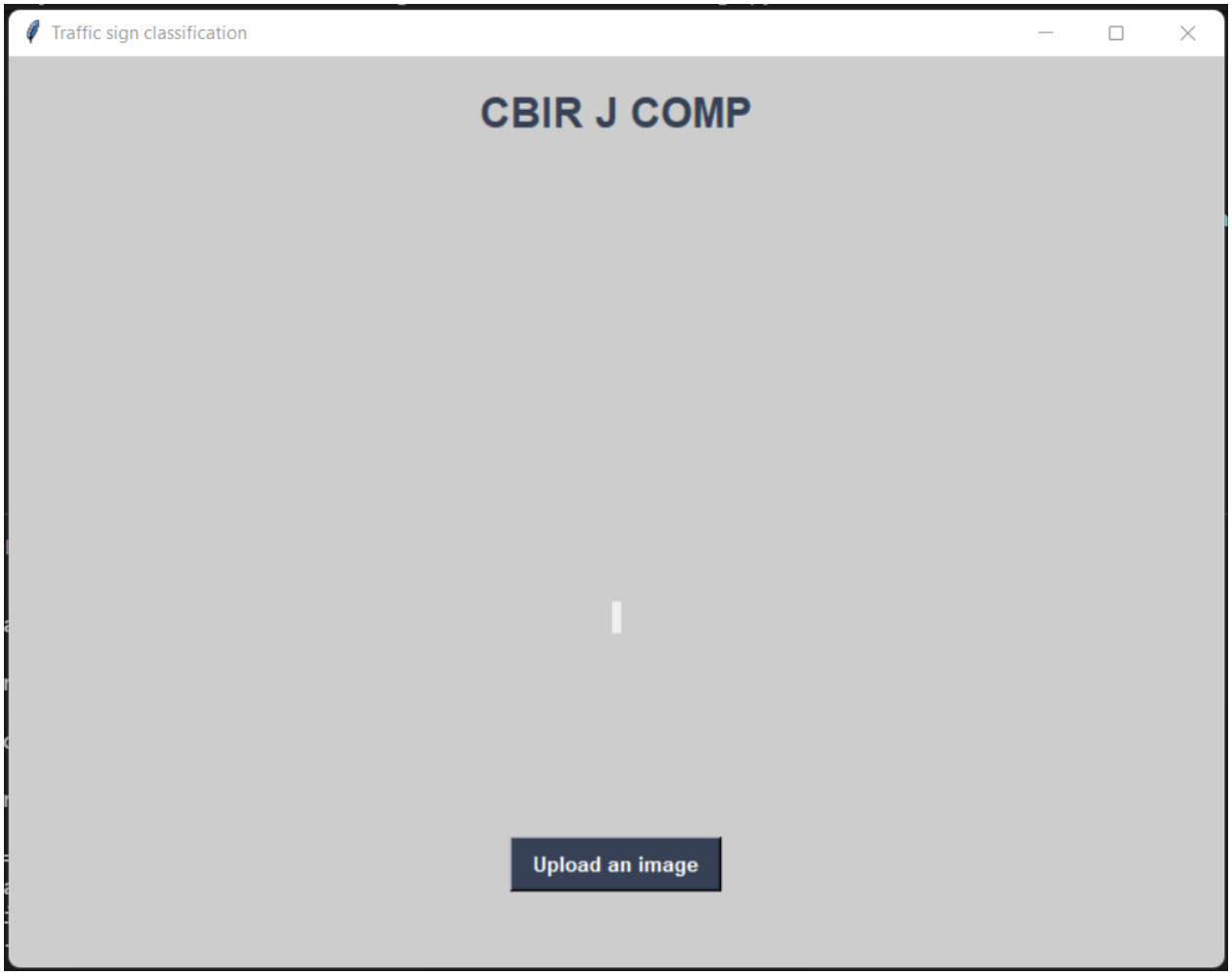
and got around 96% training accuracy and 98% testing accuracy



and had a minimal loss



and we used this model to build a simple GUI model using python's tkinter. below is how we get the initial interface



and when we give the image it accurately gives the output

Conclusion and Future work:

Hence we were able to construct a deep learning model with the help of CBIR concepts to accurately recognise the road/traffic sign meaning .The project helped us understand and work on new technologies .

Future work: with the help of opencv and object detection in video processing we can build a real time traffic sign recognition system just like how it is used in car automation

References:

[1] Alexander Shustanov and Pavel Yakimov “A Method for Traffic Sign Recognition with CNN using GPU”

[2] Saurabh Dubey, Omkar Kadam, Vandana Singh “Traffic Sign Detection and Recognition using Convolution Neural Network(CNN)”

[3] XIE BANGQUAN AND WENG XIAO XIONG “Real-Time Embedded Traffic Sign Recognition Using Efficient Convolutional Neural Network”

[4] Sanam Narejo, Shahnawaz Talpur, Madeha Memon, and Amna Rahoo “AN AUTOMATED SYSTEM FOR TRAFFIC SIGN RECOGNITION USING CONVOLUTIONAL NEURAL NETWORK”

[5] Sepehr Foroughi Shafiei,Hafiza Rimsha Rauf& Yashdeep Singh”TRAFFIC SIGN DETECTION USING DEEP LEARNING”

[6] Ayoub ELLAHYANI,Mohamed EL ANSAR,Ilyas EL JAAFARI ” Traffic Sign Detection and Recognition using Features Combination and Random Forests ”

[7] G. Bharath Kumar,N. Anupama Rani,CH. Sanath Kumar,G. Dinesh “ TRAFFIC SIGN DETECTION USING CONVOLUTION NEURAL NETWORK ”

[8] Prof Manju Khanna,Himanshu Upadhyay,Koshal Bothra,Jonah Jacob Oommen “ Traffic Sign Recognition using CNN”

[9] Ameur Zaibi, Anis Ladgham,Anis Sakly “ A Lightweight Model for Traffic Sign Classification Based on Enhanced LeNet-5 Network”

[10] Wael Farag “Recognition of traffic signs by convolutional neural nets for self-driving vehicles”