Journal of Information Systems Engineering and Management
2025, 10(26s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

A Hybrid Facial Recognition System for Secure Driver's License Verification Using Deep Learning

[1]Bhoomireddy Venkata Haripratapreddy, [2]Dr.S.P. Vijayaragavan, [3]Prof. Dr.B.Karthik
[1]Ph.D. Scholar Department of Ece, Bharath Institute Of Higher Education And Research, haripratapreddy.b@gmail.com
[2]Professor Department of Eee, Bharath Institute Of Higher Education And Research, vijayaragavan.eee@bharathuniv.ac.in
[3]Professor Department of Eee, Bharath Institute Of Higher Education And Research, karthik.ece@bharathuniv.ac.in

ARTICLE INFO

Received: 01 Jan 2025

Revised: 21 Feb 2025

Accepted: 05 Mar 2025

ABSTRACT

Extracting driver's license information using facial recognition presents significant challenges, including handling diverse lighting conditions, face orientations, and occlusions while ensuring data security. Designing and implementing driver's license information using facial recognition systems that can handle poor image quality, excessive noise, and identification in real-time is difficult. In order to solve these problems, this study introduce a novel approach which is the hybridization of Eeigen faces algorithm and deepface algorithm ensures to deliver efficient accurate face detection as liveness would be ensured from the MobileNet perspective to rule out a true face from being photographically obtained or spoofed. The system is integrated with Firebase to store and retrieve license details like name, license number, date of birth, and address. Upon detection and verification of face, the system matches this face to the corresponding record available in the Firebase database and displays license information. This approach ensures access to the information provided, minimizing fraud and enhancing authentication accuracy.

Keywords: Face Recogniton, License plate recognition, Eeigenface algorithm, Deepface algorithm, MobileNet.

1. INTRODUCTION

The number of road vehicleshas expanded in the last few years. The use of vehicles in planned criminal activities has increased and transportation has frequently escaped from the criminal activities. By the way, the connectivity of roads was increased, because it is important to notice vehicles in present days to achieve the traffic, classify the stolen or blacklisted vehicles, and control transport access to these vehicles. In this case, there are many efforts taken to solve the problem easily. When we talk about vehicle detection, there are two essential elements located: tracking and the other one is criminal recognition, in the first thing, uses a person's license recognition to track their vehicle and the other thing is to recognize that person's face in that vehicle [12]. Biometric, iris, and fingerprint detection are some of the important methods for the human identification process. Biometric technology is lower than iris recognition and fingerprint recognition through the accuracy of facial recognition systems, due to its contactless process usually the techniques adopted. Thus, facial recognition is the greatest suitable method to recognize the identity of people. The Vision-based vehicle identification systems are capable of extracting a variety of information like the type, brand, model, color data, and license plates of the vehicles. Nevertheless, every vehicle has a license plate with unique information, which can be regarded as the identity of the vehicle. Thus, the camera system is used to process the image of the vehicle and also read its number plate to identify the vehicle. License plate recognition (LPR)or automatic number-plate recognition (ANPR) is a technology that uses optical character recognition (OCR) to read the license plate number of the vehicle with images [11].

The License plate recognition systems are attached and essential components of present-day Intelligent Transportation System (ITS). The difficulty of License plate recognition systems differs all over the universe [9]. Traffic control, self-driving, parking toll stations, etc. used the applications in license plate recognition systems and it included a large amount of data. These days license plate recognition systems face a challenging task but it is lower than the unlimited situations. For some dangerous situations like constriction, uneven illumination, rotation (large position), imprecision, etc. the troubles liein identifying the license plate characters accurately. This work provides some real-lifeconsequences; therefore, it is well-meaning for their further education. Consequently, most of the methods are more effective under exact circumstances [8]. During the current years, Automatic license plate recognition takes a variety of possible applications from security to traffic control, and entices significant research attention [6].Each vehicle has an exclusive license plate that can act as a primary key in the database and comprise the data about the vehicle without the need for other technologies [10]. In the recent ALPR system's exemplary presentation in measured surroundings; through deals with these complex scenes, the performance is diminished. Multinational ALPR system deals with some uncontrolled conditions such as irregular illumination, weather (snow, fog, rain, etc.), image distortion, image blurring, occlusions, etc. are the current challenges of ALPR. The issue of multinational ALPR provides some important challenges owed to the variations in license plate layouts across different countries [5].

Detecting and recognizing LPs are the most important methods and techniques used to solve problems easily. Of these methods, there have been traditional approaches such as edge detection, morphological operations, character-based approaches, texture-based techniques, and statistical analyses [2]. In present days, different types of research have been studied by machine learning (ML) methods and deep learning (DL) systems in perfect and challenging atmospheres [10]. ML is defined by the capability of the system to learn and produce from its understanding. These techniques have been completely overshadowed by Deep Learning (DL) techniques, which provide better results. DL is mostly used for decision-making purposes but can include other applications. This kind of knowledge is extremely valuable in the fields of image analysis, speech and facial recognition, translation of languages, and emotion recognition [7]. Despite the routine of traditional and DL-based methods, still, numerous complications and tasks are still not properly answered, particularly images in complex families with multi-orientations outstanding viewing point differences in cameras, and multi-language letterings [2].

The objective of face recognition system is to securely display the individual license plate information and it deliver highly accurate face detection process. The facial recognition concept with license plate recognition is that criminals are known to either ditch their vehicle for a new one or change their license plate in order to lose their track, that's when facial recognition will help us detect the new vehicle they are travelling in. However, the rare data requires preprocessing before, it can be used for face recognition. This stage involves histogram equalization for preprocessing steps. After that, feature extraction converted raw data into suitable structure of license number plate. These features represent the furthermost related information for solve a specific problem. The extracted features were imported to our model to increase the accuracy of face recognition method with license number plate. The proposed method approach that integrates the eigenface algorithm with deep learning-based techniques influences the strengths of the both methods and achieving the balance between the computational efficiency.

Deep EigenNet: This network is the hybridization of eigenface algorithm with deep learning techniques that control the simplicity of eigen face for computational efficiency and the robustness of deepface for handling diverse data. This combination can recover the performance in specific set-ups. This model can handle large datasets efficiently by balancing computational cost and accuracy. Hence, it provides more accurate result.

This paper can be organized as follows. In section 2 briefs the literature of previous worksdone in the field. Section 3 details the proposed methodology. Section 4 discusses the experimental results of the datasets and the analysis of the models. Finally, we summarize the conclusion in Section 5.

2. LITERATURE REVIEW

In [1], the authors Jawaleet al.[1]. used the CNN method to achieve some well different conditions like low brightness, hazy or blurred images, and gradient license plate and it achieved an accuracy of 98.5% and a loss of 4.25%. Although the method gives some demerits, data shortage restricts the performance of CNN. So we will collect more data and use some advanced techniques to produce the results. Selmiet al.[2], proposed the Delp-Dar model to evaluate some datasets such as Caltech, AOLP, and PKU as well as widely explained about new Tunisian dataset, and these models give the result of AC, LP, and RP in percentage 97.8%,97.4%, and 96.3% respectively. However, the framework provided the speed of accidental occlusion which does not consider the weakness of the performance of the used equipment and addressed the various problems like low resolution, terrible illumination, etc. So, in this method, the result is not satisfied. The authors Huang et al.[3] designed the framework of a Single Neural Network for Mixed

Style License Plate Detection and Recognition used to the method CNN, and this convolutional framework solved the LP detection and recognition method without any use of RNNs. This framework achieved an accurate result of 98.21%. This method gives a highly sensitive result but it has some degraded performance because of small changes put in the input data(e.g., slight image rotations, scaling, or color changes).So, the output is not perfect in this case. In [4], the researchers Pattanaik et al. used the GAN method to complete the study of the Enhancement of license platerecognition performanceusing Xception with Mish activation function. This study addressed two difficulties: image deblurring and super-resolution, which is very useful for clearing objects and limiting block noise. Also, this method reconstructs blurred images intohigh-resolution images. These models achieve the results 0.1211 and 0.1224 of training and validation losses respectively. However, this study did not deliver great accuracy and real-time performance. In [5], the authors Henry et al. used the method YOLOv3 to prepare the summary of Multinational License Plate Recognition using Generalized Character Sequence Detection. This is the popular deep learning method for object detection, awareof its accuracy and speed. This method gives a 98.93% accurate result; nevertheless, it struggles with detecting small substances or overlapping objects. Li et al.[6] proposed a method to detect and recognize car license plates used as deep CNNs and RNNs, both methods provide high productivity and accuracy, and without the use of intermediate processing such as image cropping or character separation, the network-trained approximately. The achievement of this trained network is 99.73% (average detection ratio), but in the future, the use of NMS to extend the network and the whole processing time will take half of the time of the network. Kaur et al. [7] used the CNN method to recognize automatic license plates for vehicles, it enhanced the image quality and achieved a high recognition rate of 98.13%. However, it cannot find the difference between the grills and the plate area of motor vehicles.

3. RECOGNIZE FACES FROM LICENSE PLATES USING INTEGRATION OF DEEP LEARNING AND EIGEN FACE ALGORITHM

Face recognition is one of the important issues in object recognition and computer vision. The human face is a vital component that must be detected for a variety of purposes, including forensic analysis and security. Proper face identification and recognition techniques are necessary, given the challenges of ageing, occlusion, position fluctuations, facial emotions, and resolution, whether in the frame of a stationary item or in video sequencing photos. To overcome these issues this paper proposed a integrated method such as deep EigenNet for face recognition. The input images used to train the model is collected from firebase dataset [16] and histogram equalization is applied to the image for image normalization. After preprocessing the most relevant features are retrieved using Adam optimizer for efficient recognition. Then the extracted features are fed to the proposed deep EigenNet for train the model. Figure 1 shows the block diagram of face recognition.

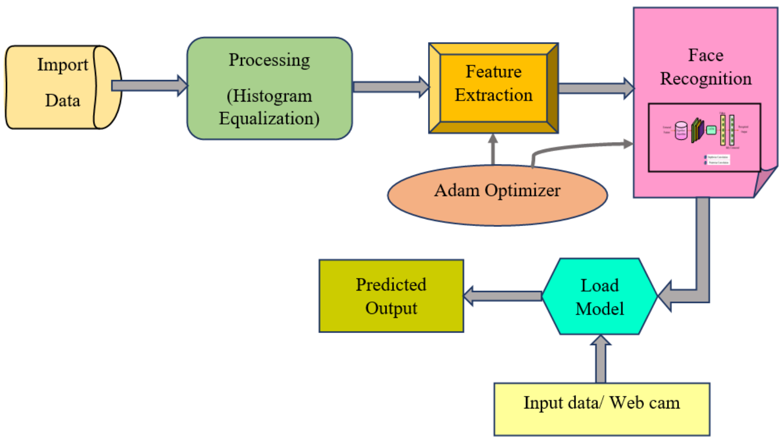


Figure 1: Block diagram of proposed methodology

3.1 Input Data

The input data are collected from the Firebase network source and the images are obtained by the given form

$$I_i = \sum G_i, \text{where} i = 1,2, \dots, n. \quad (1)$$

3.2 Data preprocessing [13]

Apply the preprocessing technique to the collected image tocategorize the spoofed and actual images and improve the image quality for more processing. This step involves three stages. They are resizing the image, eliminating noise, and normalization. The most common method used for preprocessing is histogram equalization. The idea of matching histograms is to bounce and rearrange the original histogram in the image processing method in a way that an enhancement of image contrast is achieved using the entire range of discrete levels of the image. Wherever unique efforts are taken to change the image histogram into a histogram, overall illumination values are constant. Whenever all the values are equal, they correspond to the brightness values.The histogram is defined by the image $A(a,b)$ with discrete $l$gray values and the probability of occurrence level of gray $j$is given by:

$$q(j) = \frac{c_j}{N} \tag{2}$$

Where $j \in 0,1,\dots,l-1$ grey level and N is total number of pixels in the image.

Alteration to a new strength value is defined by:

$$O_{pre} = \sum_{j=0}^{l-1} \frac{c_j}{N} = \sum_{j=0}^{l-1} q(j) \tag{3}$$

The productivity values are from the closed intervals [0,1].To get pixel values into the original interval, it must be rescaled by $[0, l-1]$. Then the output of the preprocessing is denoted by I_i^*

3.3 Feature Extraction

Feature extraction is the method of converting rare information like text, images and audio into established of computable, useful structures that capture the fundamental structure of the data. It also solves some specific problems such as sorting or deterioration. Naturally, evaluating and removing is the process of suitable material from the image or the text of a license plate. Also, it classifies the fake news from the license plate. Automatic Number Plate Recognition (ANPR) or other vehicle identification technologies are part of this system. Here the Adam optimizeris used to extract features from the pre-processed image, and targeting to simplify the compound and well extract features from the image. The extracted structures were introduced to raise the recognition accuracy of the model, and the output of this stage is expressed as I_i^{**}.

3.3.1 Adam optimizer

The adaptive moment estimation (Adam) algorithm [14] is a very important algorithm used for various fields, it is a highly efficient adaptive optimization method for training deep neural network models. It is a broadly used and multipurpose optimization process for its effectiveness and efficiency. The role of this algorithm is to estimate the instants and develop them for function optimization. It helps update the weights and biases more efficiently and extends the stochastic gradient descent. Exactly, the root mean square (RMS) prop algorithm is the union of gradient descent with the moment algorithm. This algorithm uses historical gradient information and adaptive learning rates. Each parameter of ADAM regulates the learning rate, avoiding alternations and supporting quicker convergence. Likewise, it is in contradiction of flexible and loud slopes with different neural network tasks.

The benefits of this algorithm are, that needs a tiny memory, is computationally more effective, and is suitable for problems having large information. Iterative Formulations used in Adam optimizer are assumed by way of:

$$f = (g_\sigma(u-v))u^r \tag{4}$$

$$o_s = \alpha_1 o_{s-1} + (1-\alpha_1) * f \tag{5}$$

$$p_s = \alpha_2 p_{s-1} + (1-\alpha_2) * f^2 \tag{6}$$

$$o_s^- = \frac{o_s}{1-\alpha_1^s} \tag{7}$$

$$p_s^- = \frac{p_s}{1-\alpha_2^s} \tag{8}$$

$$\sigma_t = \sigma_{t-1} - o_s^- * \frac{\beta}{\sqrt{p_s^-}+\varepsilon} \tag{9}$$

where $f$denotes the calculated gradient, $o_s$is the first moment of gradient$f$,$p_s$stands for the second moment of gradient f, $\alpha_1$denotes the first-order moment reduction coefficient, α_2 stands for the second - moment reduction coefficient, σrepresents the parameter that needs to be solved, $o_s^-$and p_s^-definebalanceimprovement of $o_s$and $p_s$respectively.

3.4 Integrated DeepNet Eigen Face Model

This paper introduces an integrated model for face recognition, it is acombination of eigenface, LSTM, and MobileNet. The eigenface algorithm easily understands the problem, compares it to the deeper learning methods, and transforms the face image into a smaller one. It works very quickly because it reduces the dimensionality of the facial data after that it transforms from a high-dimensional pixel image and reduces its computational load. Hence, lower dimensional pixels allow for faster face recognition. The eigenface algorithm works very well if the camera directly focuses on the face. However, when the person's face is rotated, the algorithm gives lower performance and results in poor recognition accuracy for any considerable change in pose. Therefore, MobileNet andLSTM is integrated with the eigenface algorithm to recognize the face data at various poses. The LSTM learns the temporal relationships between different frames or sequences of face data.MobileNet is a lightweight neural network architecture designed for efficient image classification on mobile devices. It uses depthwise separable convolutions to reduce the number of parameters and computations.

The Eigen face algorithm is based on the Principle Component Analysis (PCA) and thisis used toextracts appropriate data from a face image after that converts it into a face code. This is called the eigenvector. The face code is compared to the face database. Eigenvector is alsostated as facial features; therefore, this algorithm is called as eigenface algorithm. To each face is characterized in a direct eigenface combination.The Eigenface calculation consists of twostages, namely the training stage and the face recognition stage.

Calculating the average or mean value(Φ). I_i^{**}represents the extracted features from the training image. I_i refers to total amount of training image.

$$\Phi = \frac{1}{I_i}\sum_{i=1}^{n} I_i^{**} \tag{10}$$

Calculating the difference (Ψ) between $I_i^{**}$with mean value (Φ).

$$\Psi = I_i^{**} - \Phi \tag{11}$$

Calculating the value of the covariance matrix (M),

$$M = \frac{1}{I_i}\sum_{i=1}^{n} \quad \Psi\psi_i = BB^T \tag{12}$$

$$F = B^T B = \psi_i \psi^T \tag{13}$$

B is the matrix that consists of differences between each I_i^{**}withmean value.

$$B = \{\psi_1, \psi_2, \dots, \psi_n\} \tag{14}$$

Calculating the eigenvalue (γ) , and eigenvector $(e)$from the covariance matrix (M).

$$M \times e_i = \gamma_i \times e_i \tag{15}$$

After the eigenvector (e) is obtained, the eigenface (η)can be calculated by the below equation.

$$\eta_i = \sum_{m=1}^{n} \quad e_{im}\psi_m \tag{16}$$

The obtained eigenface (η) features are fed to the LSTM network to find the pose variations of the training image. There are several architectures of LSTM units which is shown in Figure 2. That consists of a cell and three "regulators", usually called gates, of the flow of knowledge inside the LSTM unit: an input gate, an output gate, and a forget gate. Approximate variations of the LSTM unit don't have one or more of those gates or even produce other gates.

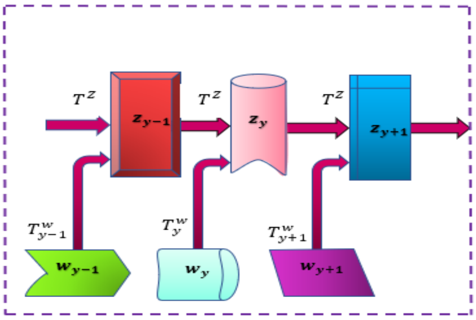


Figure 2: Architecture of LSTM

An ordinary face recognition system void of concrete and deliberate anti-spoofing measures (as the case is with many deployed face recognition systems) is prone to failure if there be spoof attack attempts on such a system. Hence, anti-spoofing measures is used to extract features from facial images in real-time, and these features can be analyzed to distinguish between real faces and spoofed ones.Face anti-spoofing approaches based their cues onthe physiological behaviors of the human face such as theblinking of the eyes, head rotation, facial expressions, andlips movement. Generally, MobileNet is used as ideal for real-time applications due to its small size and efficiency. When combined with other techniques such LSTM, it can help in detecting the authenticity of biometric samples in scenarios like facial recognition systems.

Figure 3 presents the overall architecture of theMobileNet V2 with the LSTM model with a combination of set of convolutions and max pooling layers and the LST component that is attached to the flattening layer of the model. The fully connected layer that performs the correlation of the identified features with the pre-existing data through training. Finally, the softmax layer that determines the ID is fake or not.

$$B_y = \alpha_x(T_B w_y + S_B z_{y-1} + V_B z'_{y-1} + h_B) \tag{17}$$

$$C_y = \alpha_x(T_C w_y + S_C z_{y-1} + V_C z'_{y-1} + h_C) \tag{18}$$

$$D_y = \alpha_x(T_D w_y + S_D z_{y-1} + V_D z'_{y-1} + h_D) \tag{19}$$

$$\tilde{E}_y = \alpha_k(T_E w_y + S_E z_{y-1} + V_E z'_{y-1} + h_E) \tag{20}$$

$$E_y = B_y \circ E_{y-1} + C_y \circ \tilde{E}_y \tag{21}$$

$$H_y = D_y \circ \alpha_k(E_y) \tag{22}$$

Where $T_B, T_C, T_D, and\ T_E$ represents the weight metrics, $h_B, h_C, h_D, and\ h_E$represents the bias vectors,$V_B, V_C, V_D, and\ V_E$represents the weights associated with the hidden recurrent layer,α represents the sigmoid function, and '$\circ$' represents the element-wise -product.

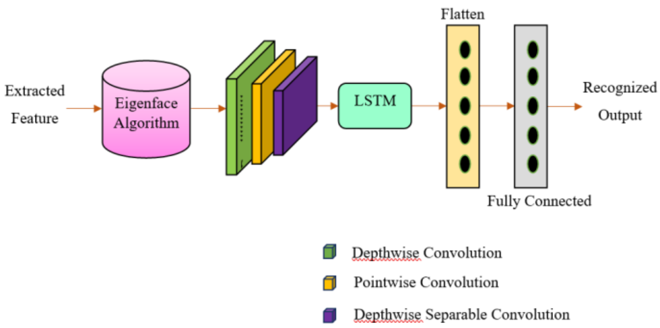


Figure 3: Architecture of eigenNet-LSTM

The performance of deep learning techniques is highly dependent on the choice of hyperparameters, and optimization involves searching for the best combination. It reduces the complexity and processing time and improves the accuracy of the model. This model utilizes the Adam optimizer for selecting hyperparameter such as weight and bias.

4. RESULT AND DISCUSSIONS

Our proposed system achieves notable improvements in facial recognition accuracy and reliability, even under challenging conditions such as varying illumination and occlusions. The test is run on a machine equipped with an Intel (R) core (TM) i5 4570s CPU @ 2.90 GHz, 8GB RAM, and the computer name SSM107.smg.local running Windows 64-bit. Acer is the system manufacturer using the PYTHON tool. Our experimental configuration includes two data centers with four hosts and a total RAM of 8 GB. The host has a bandwidth of 2800 Mbps.

4.1. Dataset Description

Kaggle presents artificial images of European driver licenses found in the Synthetic EU Driver's Licenses dataset. Its main objective is to facilitate computer vision research particularly within document detection together with OCR (Optical Character Recognition) and fraud detection fields. The data collection features diverse license images endorsed with different layouts, fonts and security characteristics which imitates genuine license documents while preserving privacy measures. Each license image contains artificial data which includes truthful yet false personal information including names and addresses combined with dates of birth and license numbers. Model training receives structured information through metadata files which accompany the image documents. The fake document dataset serves perfectly for modeling document verification procedures combined with information extraction tasks while avoiding exposure of actual sensitive information.

4.2. Evaluation Metrics

It has chosen many metrics to gauge how well predict the face recognition for secure driver's license. They have selected accuracy, precision, recall, and f-measure for our investigation. The confusion matrix was primarily used to determine the true positive, true negative, false positive, and false negative for the majority of the measurements. To evaluate these findings, compute the precision, recall, accuracy, and F1-score, FPR, FNR, MCC and NPV indicators.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{15}$$

$$recall = \frac{TP}{TP + FN} \tag{16}$$

$$precision = \frac{TP}{TP + FP} \tag{17}$$

$$F1 - score = \frac{2TP}{2TP + FP + FN} \tag{18}$$

TP signifies the true positive, FP the false positive, TN the true negative, and FN the false negative.

4.3 Experimental results

The experimental results, including accuracy, precision, recall, F-measure, and accuracy vs. loss value, show the performance of the suggested method.

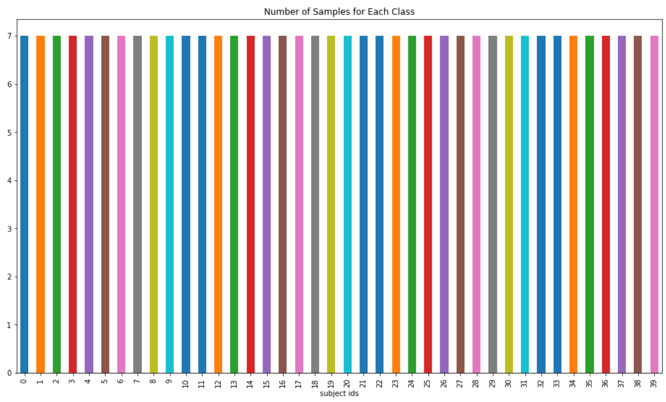


Figure 4: Distribution of Samples across Classes

The Figure 4 displays an organized bar chart which shows the sample count distribution regarding subject-specific identification classes. Each vertical bar in this bar chart shows how many samples belong to a given subject therefore illustrating one bar for each entry. Each class maintains an equivalent number of samples according to the bar chart and shows no preference for any particular class distribution. Employed bar colors enable clear distinction and reading of the various subjects. Such equal distribution of samples remains vital to guarantee the reliability of any model derived from this dataset.

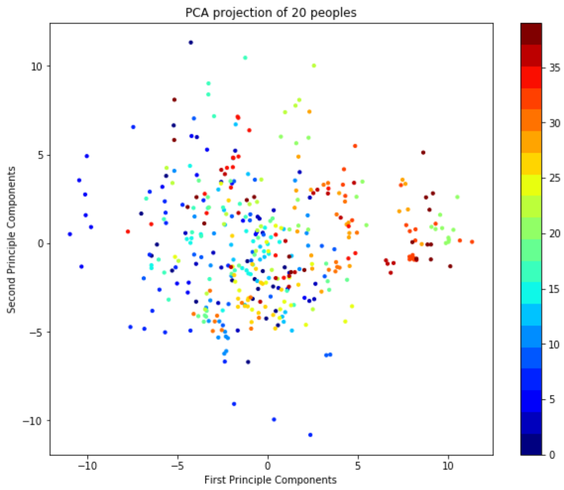


Figure 5: PCA Projection of 10 People

A Figure 5 presentation of a ten-person dataset appears in this scatter plot because of the PCA transformation procedure. The first primary component makes up the x-axis which corresponds to the second primary component shown on the y-axis. Different samples from ten individuals appear as points on the plot where colour representation indicates the assignments of these subjects to defined categories. Points that spread across the principal space provide an overview of the sample variations which capture principal data variations. Several distinct color clusters indicate that particular individuals possess common traits. This PCA representation produces valuable understanding about data organization together with inter-individual relationships.

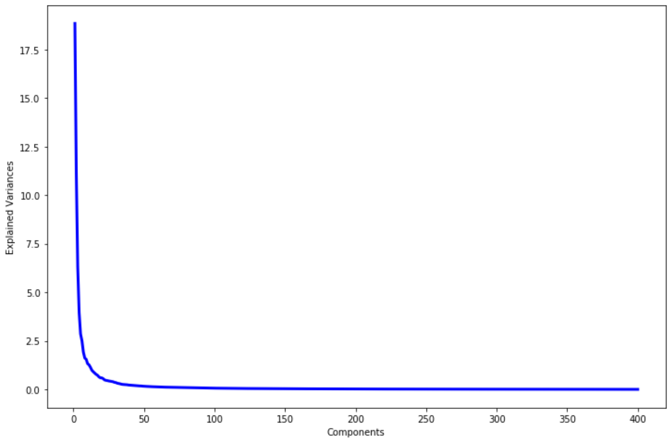


Figure 6: Explained Variance vs. Principal Components

The Figure 6 presents the connection between primary component numbers and the amount of dataset information explained. The analysis displays two axes where the x-axis shows principal component number and the y-axis demonstrates the explained variance value. The explained variance experiences a rapid reduction when the first few components are selected since they extract substantial portions of total data variation. The explained variance becomes stable after the 50th component addition since extra components no longer significantly contribute to the reduction in observed variance. The data structure contains enough information to be effectively captured by only a small number of principal components. The visualization serves as a critical step to identify which number of components will provide the most suitable data for additional investigation.

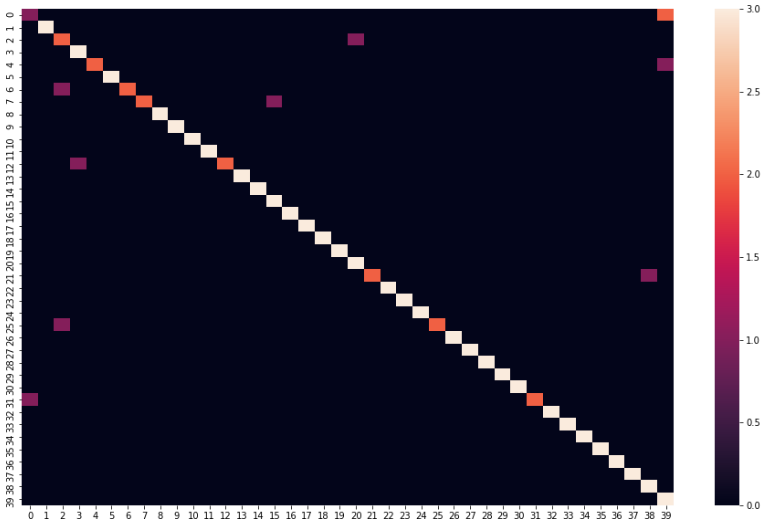


Figure 7: Heatmap of Correlation Matrix

The Figure 7 represents the correlation matrix between 40 variables and reveals different inter-variable connections in the dataset. The color gradient extends from dark purple to bright orange to show correlation strength with light colors representing higher values. The presented matrix shows only the upper right section because correlation exists as a symmetrical relationship. A value between 0 and ±3 on the color gradient demonstrates the strength of relationship between variables. The color gradient of absent squares across the matrix helps researchers determine which variable pairs lack correlation significance during their analysis of key variables. The heatmap serves as an indispensable instrument that helps both identify suitable variables and reveal data relationships between them.

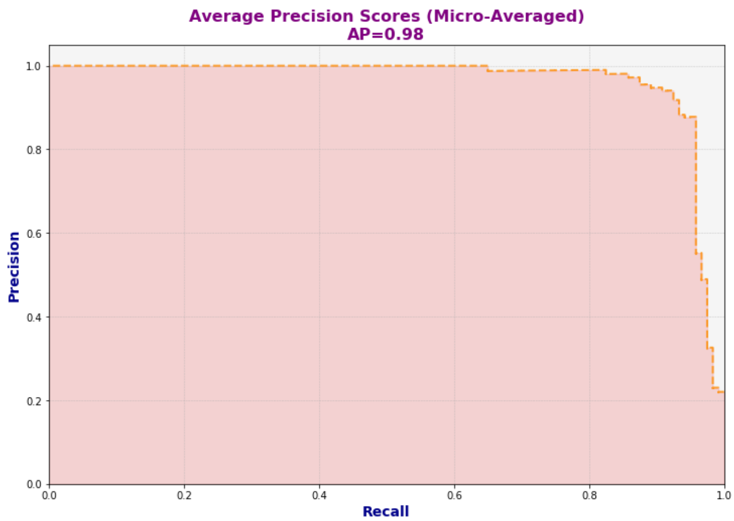


Figure 8: Precision-Recall Curve

A Figure 8 shows precision-recall curve shows micro-averaged performance for every class in this display and reaches an Average Precision (AP) score of 0.97. The recall of predictions appears on the x-axis while precision values reside on the y-axis to demonstrate their relationship. The plot depicts high precision when near 1 throughout most of the values along the curve particularly at the start of the recall range. The model precision experiences a rapid decline in performance areas when achieving higher recall levels thus confirming the prediction of less accurate positive results. This curve aids model performance evaluation and imbalanced datasets assessment because it shows how well the classification method detects actual positive events. The obtained AP score indicates positive outcomes regarding the model's performance.

(a)

(b)

Figure 9: sample images from ids

The Figure 9 presents examples of driving licenses (IDs) that are utilized during identification processes and verification functions. Multiple driving license formats present in the images feature necessary elements that include photo identification along with name as well as date of birth and license number. The data samples include different design variations from different areas and serve to ensure the system operates with diverse formats. Deep learning models use these images for training purposes to achieve accurate information extraction and authentication. The inclusion of visual dataset enhances the model's capability to detect forgery along with weaknesses in document information. The implementation of image-based processing makes automated ID verification systems run more reliably.

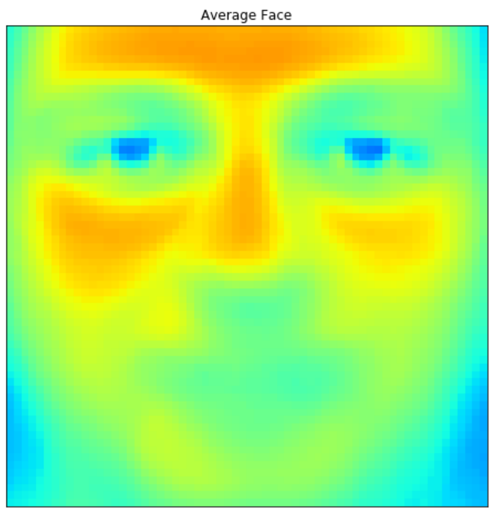


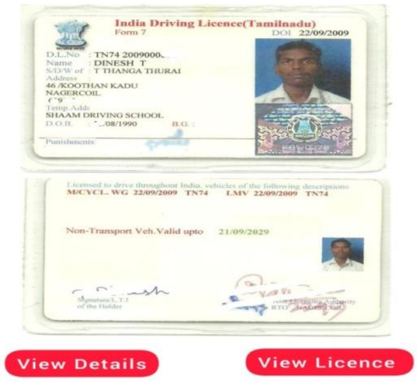
Figure 10: show average face

Figure 10 shows the average face representation obtained by combining facial features from different samples. The composite face points out typical patterns, such as the general location of eyes, nose, and mouth, averaging out individual differences. The heatmap effect demonstrates the consistency of feature alignment, with darker colors representing high-consistency areas. Such representations are useful in training facial recognition models for learning generalized facial structures. The mean face helps mitigate bias by smoothing out distinctive facial features between samples. This method increases the accuracy and stability of the model for real-world identity authentication tasks.

Figure 11: Eigen faces

Figure 11 shows a collection of eigenfaces, the major components extracted from a collection of facial images by Principal Component Analysis (PCA). The eigenfaces capture the most important features that change across the dataset, including the outlines of the eyes, nose, and mouth. The top rows are the most prominent features, and lower rows depict finer, less influential details. Eigenfaces facilitate dimensionality reduction of facial data, making face recognition efficient through the identification of important facial variations. The union of these elements facilitates reconstruction and identification of faces accurately. The method improves model performance by discarding redundant information.

(a) (b)



(c)

Figure 12: User-friendly Window for License Verification

Figure 12 showcases a user-friendly window for license verification that presents a successful verification outcome, wherein the system effectively identifies a match for the uploaded driver image within the existing database. The interface highlights key details such as the driver's name and license validity, reinforcing user confidence in the accuracy of the verification process. This streamlined presentation not only simplifies the user's interaction with the system but also emphasizes the efficiency of the matching algorithms in delivering rapid results. Overall, the design serves to enhance user satisfaction by facilitating quick and clear outcomes during the verification process.

Figure 13: License Verification Failed

Figure 13 illustrates a scenario where the license verification process has failed, indicating that the system could not find a match for the uploaded driver image in the database. The prominent error message, Verification Failed! User not found, clearly communicates the outcome to the user, ensuring transparency in the verification process. This feedback is crucial for users, as it prompts them to check the accuracy of the uploaded image or consider re-uploading a different one. Overall, this interface design emphasizes the importance of clear communication in guiding users through potential issues during the verification process.

```
Accuracy score: 0.98

Classification Results:

              precision    recall  f1-score   support
           0       0.96      0.99      0.97        15
           1       0.97      1.00      0.98        27
           2       0.96      0.99      0.97        20
           3       0.98      1.00      0.99        18
           4       1.00      0.97      0.98        26
           5       0.98      0.99      0.98        26
           6       0.99      0.96      0.97        41
           7       0.99      0.97      0.98        30
           8       0.96      0.98      0.97        16
           9       0.98      0.98      0.98        46
          10       1.00      1.00      1.00        24
          11       0.98      0.97      0.97        42
          12       0.97      0.97      0.97        48
          13       0.98      0.97      0.97        27
          14       1.00      0.96      0.98        19
          15       1.00      0.99      0.99        21
          16       0.98      0.98      0.98        15
          17       0.96      0.98      0.97        37
          18       0.98      0.99      0.98        20
          19       0.97      0.99      0.98        37
...

   micro avg       0.98      0.98      0.98      1198
   macro avg       0.98      0.98      0.98      1198
weighted avg       0.98      0.98      0.98      1198
```

Figure 14: classification report

The model's overall class performance appears in Figure 14 within a comprehensive classification report. The report shows precision and recall measurements alongside F1-score for each class in addition to providing support statistics which refer to the true sample counts per class. The model reaches an outstanding accuracy rating of 0.98. The micro average method assesses complete model performance by treating all observations as equally important but the macro average derives an overall metric by averaging all class measurements using uniform class weightings. The weighted average technique gives more importance to each class according to its prevalence through support measurement. The model exhibits strong prediction capabilities according to its consistently high F1-scores which demonstrate effective precision and recall balance.

Table 1: Comparison table

Methods	Accuracy	Precision	Recall	F-measure
LSTM	97.4	97.4	97.4	97.4
RNN	98.04	97	96	97
CNN	97.19	97.9	97.9	97.9
proposed	99.2	99.3	99.4	99.9

The table 1 given below compares the performance of different deep learning models—LSTM, RNN, CNN, and a proposed approach—on metrics such as Accuracy, Precision, Recall, and F-measure. The proposed approach performs better than the rest, with an Accuracy of 99.2% and a F-measure of 99.9%. Long Short-Term Memory (LSTM) networks are good at capturing long-term dependencies in sequential data, which makes them perform well in tasks such as sentiment analysis. Recurrent Neural Networks (RNNs) also deal with sequential data but can experience difficulty with long-term dependencies and therefore might suffer slightly lower performance than LSTM. Convolutional Neural Networks (CNNs) are good at spatial feature extraction and have also been used with text data to great success, although they could perhaps not recognize temporal dependencies quite as well as LSTM or RNN. The better performance of the new method indicates that it includes enhancements to classic architectures, perhaps by harnessing strengths from several models or implementing new mechanisms to more effectively learn data patterns. For example, hybrid models such as CNN-LSTM have shown better accuracy through the use of CNNs for feature extraction and LSTMs for sequence prediction. In one COVID-19 detection study, a CNN-LSTM model found 98.9% accuracy, a better result than a single CNN model's 96.73% accuracy.

5. CONCLUSION

In this paper, a novel deep EigenNet model is proposed to recognize the license plate through facial recognition. The data is collected from the Firebase database and the raw data cannot be used directly for the recognition system. Hence a pre-processing step is performed, which is used to handle the datanormalization. After that feature extraction is performed on the data that helps to reduce the amount of redundant data from the data set. Then the extracted features are applied to proposed deep EigenNetmodel which is used to efficiently train the model for recognize face data. Using these trained data, the model precisely recognizes the license plate, resulting which enhanced accuracy.

REFERENCES

[1] Jawale, M. A., P. William, A. B. Pawar, and Nikhil Marriwala. "Implementation of number plate detection system for vehicle registration using IOT and recognition using CNN." Measurement: Sensors 27 (2023): 100761.
[2] Selmi, Zied, Mohamed Ben Halima, Umapada Pal, and M. Adel Alimi. "DELP-DAR system for license plate detection and recognition." Pattern Recognition Letters 129 (2020): 213-223.
[3] Huang, Qiuying, Zhanchuan Cai, and Ting Lan. "A single neural network for mixed style license plate detection and recognition." IEEE Access 9 (2021): 21777-21785.
[4] Pattanaik, Anmol, and Rakesh Chandra Balabantaray. "Enhancement of license plate recognition performance using Xception with Mish activation function." Multimedia tools and applications 82, no. 11 (2023): 16793-16815.
[5] Henry, Chris, Sung Yoon Ahn, and Sang-Woong Lee. "Multinational license plate recognition using generalized character sequence detection." Ieee Access 8 (2020): 35185-35199.
[6] Li, Hui, Peng Wang, and Chunhua Shen. "Toward end-to-end car license plate detection and recognition with deep neural networks." IEEE Transactions on Intelligent Transportation Systems 20, no. 3 (2018): 1126-1136.

[7] Kaur, Parneet, Yogesh Kumar, Shakeel Ahmed, AbdulazizAlhumam, Ruchi Singla, and Muhammad Fazal Ijaz. "Automatic License Plate Recognition System for Vehicles Using a CNN." Computers, Materials & Continua 71, no. 1 (2022).

[8] Zou, Yongjie, Yongjun Zhang, Jun Yan, Xiaoxu Jiang, Tengjie Huang, Haisheng Fan, and Zhongwei Cui. "License plate detection and recognition based on YOLOv3 and ILPRNET." Signal, Image and Video Processing 16, no. 2 (2022): 473-480.

[9] Kulkarni, Parag, Amit Khandebharad, Dattatray Khope, and Pramod U. Chavan. "License plate recognition: a review." In 2012 Fourth International Conference on Advanced Computing (ICoAC), pp. 1-8. IEEE, 2012.

[10] Quraishi, Abdul Awal, Farid Feyzi, and AsadollahShahbahrami. "Detection and recognition of vehicle licence plates using deep learning in challenging conditions: a systematic review." International Journal of Intelligent Systems Technologies and Applications 22, no. 2 (2024): 105-150.

[11] Alim, Fikret, EnesKavakli, SefaBurakOkcu, Ertugrul Dogan, and CevahirCigla. "Simultaneous license plate recognition and face detection application at the edge." In AI and Optical Data Sciences IV, vol. 12438, pp. 232-243. SPIE, 2023.

[12] Agarwal, Vrinda, Aaron George Pichappa, ManideepRamisetty, and BalaMurugan MS. "Suspicious Vehicle Detection Using Licence Plate Detection And Facial Feature Recognition." arXiv preprint arXiv:2304.14507 (2023).

[13] Anila, Satish, and N. Devarajan. "Preprocessing technique for face recognition applications under varying illumination conditions." Global Journal of Computer Science and Technology 12, no. 11 (2012): 13-18.

[14] Kohli, Himani, Jyoti Agarwal, and Manoj Kumar. "An improved method for text detection using Adam optimization algorithm." Global Transitions Proceedings 3, no. 1 (2022): 230-234.

[15] Murthy, G. S. N., Shanmukha Rao Allu, Bhargavi Andhavarapu, Mounika Bagadi, and Mounika Belusonti. "Text based sentiment analysis using LSTM." Int. J. Eng. Res. Tech. Res 9, no. 05 (2020).