Journal of Computer Science

Research Article

Person Re-Identification From Video Surveillance Systems Using Artificial Intelligence Methods

Revathi Lavanya Baggam and Vatsavayi Valli Kumari

Department of Computer Science & Systems Engineering, Andhra University College of Engineering, Waltair, India

Article history
Received: 05-09-2024
Revised: 21-03-2025
Accepted: 17-04-2025

Corresponding Author:
Revathi Lavanya Baggam
Department of Computer Science & Systems Engineering, Andhra University College of Engineering, Waltair, India
Email: revathilavanya.research@gmail.com

Abstract: The study explores use of deep learning models in person re-identification, leveraging the advancements made in face recognition however the abundance of model choices presents a challenge in selecting the optimal architecture. The study proposes a comprehensive framework for evaluating deep learning models on person re-identification tasks by considering various performance metrics, dataset preprocessing methods, model architectures, and evaluation techniques to enable a systematic comparison of different approaches through empirical analyses on standard person re-identification datasets. The proposed framework is worked-out in uncovering the strengths and limitations of diverse deep learning strategies. The primary objective is to utilize face recognition methodologies to achieve accurate person re-identification.

Keywords: Deep Learning (DL), Machine Learning (ML), Face Recognition (FR), Mathematical Model, Model Comparison, Performance Metrics, Dataset Preprocessing, Model Architecture, Evaluation Methodology

Introduction

Face recognition is a crucial component of many applications, including surveillance, security systems, and biometric authentication. The advent of deep learning, improved accuracy and efficiency of face recognition systems (Peng & Gopalakrishnan, 2019). However, the plethora of deep learning models poses a challenge in selecting the most appropriate model for a given task. This paper addresses the challenge by proposing a mathematical model for comparing deep learning models on face recognition datasets. The fundamental goal of face recognition is to automatically identify or verify individuals based on their facial characteristics. This introduction provides an overview of face recognition techniques, their evolution, challenges, and applications.

Face recognition techniques have evolved substantially over the dec-ades, driven by advancements in computer vision, ML and DL (Varanasi *et al.*, 2022). Early face recognition systems relied on handcrafted features such as Eigenfaces, which represented faces as vectors in a high-dimensional space (Jalalipour *et al.*, 2023). These techniques were limited by their reliance on shallow representations and struggled with variations in pose, illumination, and expression.

Advent of machine learning algorithms, particularly Support Vector Machines (SVM) and Neural Networks, face recognition systems began to leverage more sophisticated feature representations (Zhang *et al.*, 2021). Local Binary Patterns (LBP) and Histogram of Oriented Gradients (HOG) emerged as popular feature descriptors, enabling better robustness to variations in facial appearance.

Despite significant progress, face recognition still faces several challenges:

- Variability in Facial Appearance: Faces exhibit significant variations due to factors such as pose, illumination, expression, occlusion, and aging and an effective face recognition system must overcome these differences (Qi *et al.*, 2019).
- Privacy and Ethical Concerns: Ethical considerations surrounding consent and data protection are paramount (Liu *et al.*, 2017a).
- Bias and Fairness: Ensuring fairness and mitigating biases such as ethnicity, race, gender is essential for deploy-ing face recognition systems responsibly (Wang & El Saddik, 2023).
- Security Vulnerabilities: Face recognition systems are susceptible to attacks such as spoofing and crafted inputs designed to deceive the system. Robustness against such attacks is critical (Li & Chen, 2019).
- Security and Surveillance: FR is widely used in exploration systems, access control, and border security for identifying individuals of interest (Ren *et al.*, 2017).

SCIENCE Publications

- Biometric Authentication: Face recognition serves as a convenient and secure biometric modality for user authentication in smartphones, ATMs, and other devices (Wang *et al.*, 2018).
- Personalized Services: Face recognition enables personalized services in retail, entertainment, and advertising by tailoring experiences based on customer profiles (Huai *et al.*, 2020).
- Law Enforcement: Asist law enforcement agencies in developing forensic evidence (King *et al.*, 2022).

Face recognition techniques have witnessed remarkable advancements, fueled by the integration of ML and DL algorithms. Despite challenges related to variability, privacy, bias, and security, face recognition continues to play a pivotal role in various applications, offering both opportunities and ethical considerations in its deployment. This introduction sets the stage for exploring the technical aspects, methodologies, and advancements in face recognition technology.

Deep learning has emerged as a powerful paradigm within the field of artificial intelligence, enabling computers to learn complex representations of data through hierarchical layers of neural networks (Hou *et al.*, 2021).

The key architectures of DL Models include Feedforward Neural Networks (FNN) (Zhao *et al.*, 2023), Convolutional Neural Networks (CNN) (Fu *et al.*, 2019), Recurrent Neural Networks (RNN) (Modak *et al.*, 2022), Long Short-Term Memory Networks (LSTM) (Nair *et al.*, 2023) and Generative Adversarial Networks (GAN) (Fabarisov *et al.*, 2023).

Deep Learning Models have found applications across diverse domains, including Computer Vision, Natural Language Processing (Liang *et al.*, 2015), Speech Recognition and Autonomous Vehicles (Yuan *et al.*, 2022).

Deep learning models have transformed the landscape of artificial intelligence, offering powerful tools for solving complex problems across various domains. From image recognition and natural language understanding to speech recognition and autonomous systems (Swetha *et al.*, 2022), deep learning continues to drive innovation and reshape industries.

We anticipate a broad survey of human face recognition via machine in the current research graft, as well as a succinct study of related psychological reports. There were two forms of face recognition tasks (Hayat *et al.*, 2014) considered, one from video and the other from static images (Dai *et al.*, 2022). We have categorized all the techniques used and examined their benefits, drawbacks, and characteristics. We also compiled a list of current approaches, as well as the obstacles we face (Suri *et al.*, 2022). In real-time face recognition systems, two major issues have been identified, posture and illumination problems. We represent a review of our survey's findings as well as conclusions.

- Mechanical face recognition, image classification, image pro-cessing, computer vision, and neural networks are only a few ex-amples in an emerging and influential research area. Face biometric assisted ATM and control mechanisms, criminal detection (Chen *et al.*, 2023), and traffic management systems, among other commer-cial appliances, use it. While there are many powerful biometric methods available, such as fingerprint analysis and iris scans, face recognition has proven to be a popular form of personal identifica-tion due to its efficiency and convenience.
- In the literature, there are various image intensities-based face recognition approaches (Fang *et al.*, 2023). These strategies are extremely useful, but we also have drawbacks. The ap-proach is chosen, however, based on the specific requests of a particular application.
- In certain cases, videos are shot in an unchecked environment. Due to which, the videos are of poor quality, making it difficult to discern a face from a collection of images. Since humans are often shy or uninterested in recording, it can be difficult to recognize faces and obtain high-quality images. Face recognition structures (Bhatt *et al.*, 2023) based on multiple cues have shown excellent results in a controlled environment.
- Despite the numerous advanced face recognition techniques available, accurate face recognition is difficult to achieve. In a nutshell, the main problems are lighting, posture, and recognition in an open space. There are a number of approaches that can be used to address these problems in face recognition procedures. However, there are a few basic problems that need to be addressed, such as judging posture (Zhang & Bao, 2022), which is not difficult, but estimating a person's precise pose is difficult. Along with the above concerns, there are also several other issues, such as identifying a face from a photo taken several years ago.

Several approaches to face recognition have been established, but current approaches are used on face data. Video face recognition methods, which provide more information to enhance the protection system, have recently been implemented. In this research, a computer vision system is used to detect, track, and identify faces. To accomplish this goal, we first used a face detection and tracking method based on Kalman filtering. Following facial detection, extracted the input image's combined features and saved the trained data. The Bayesian learning technique is utilized for enhancing the learning process. A learning model based on RCNN (Lollett *et al.*, 2023) is also utilized to classify the detected faces using Joint Bayesian learning. The proposed approach, Background Subtraction Faster RCNN, incorporates these steps for video-based facial recognition.

Deepfakes often manipulate existing material by replacing one person with another or create entirely new content which is not actual.

Finally, a deep learning framework is introduced, encompassing various features such as image enhancement, development of a module for extracting features, aims to produce meaningful features through CNN training and testing.

Overall, designing a framework for face detection based on deep learning is crucial for unlocking the full potential of face detection technology and its applications across various industries and domains. Designing a framework for face detection based on deep learning offers numerous advantages like accuracy, robustness, flexibility, scalability, adaptability and integration capabilities, making it a preferred choice for various face detection applications.

Designing feature extraction architectures for deepfake detection is essential for developing effective, robust, and efficient detection systems capable of identifying and mitigating the proliferation of deepfake content across various digital platforms and applications.

Literature Review

Over the years, researchers have developed numerous techniques and algorithms to address the challenges inherent in face recognition, including variations in expression, occlusion, aging etc. This related work provides an overview of the state-of-the-art techniques in face recognition, highlighting advancements, challenges, and emerging trends.

Techniques such as transfer learning, that fine-tunes models on definite face recognition datasets, have facilitated the development of efficient models with reduced training data requirements. Additionally, attention mechanisms, recurrent neural networks (RNNs), and transformer-based architectures have been explored for temporal dependencies and capturing long-range contextual information in face images.

Liang *et al.* (2022) explained that face recognition from low-light exposures is complex due to the small number of photos available and the unavoidable noise, which is also spatially unevenly distributed, making the task even more difficult. The principle of exposure, which records various shots to achieve well-exposed photographs under challenging conditions, is a natural solution.

According to Zhou *et al.* (2020) in the field of security, the image taken by an outside surveillance camera, normally has distorted faces occluded in a variety of poses and tiny which is influenced by external factors such as camera pose and distance as well as weather conditions and so on. It is an issue of hard face recognition in natural photographs to put it that way. They suggested a deep convolutional neural network to solve the problem—two observations from contextual semantic knowledge and the process of scale face detection. Li *et al.* (2016) discussed face detection with end-to-end integration of convnet. Tao *et al.* (2016) explained CNN and SVM based robust face detection. Pham *et al.* (2016) narrates real time performance driven 3D face tracking. Ranganatha & Gowramma (2016) worked out fused algorithm for face tracking. Maleš *et al.* (2019) discussed multi agent dynamic system using expert systems. Ding & Tao (2015) discuss multi modal deep face representation and face recognition by very deep neural networks by Sun *et al.* (2015).

In their study, Vishwakarma & Dalal (2020) introduced a new technique for handling face recognition in varying illumination conditions caused by changes in the angle of light projection. This technique applies adaptive illumination normalization and dditionally, a non-linear modifier is utilized to modify DCT coefficients.

Face recognition techniques for FR have recently been enhanced and to account for nonlinear changes attributable to posture, a double sparse representation model is utilized (Mokhayeri and Granger, 2020).

According to Yang *et al.* (2015), the MDML-DCPs (Multi-Directional Multi-Level Dual-Cross Patterns) scheme, used the first derivative of the Gaussian operator to reduce the impact of illumination differences. Chen *et al.* (2015) proposed PCANet to learn multistage filter banks that demonstrated impressive performance in various image classification tasks. However, it should be noted that PCANet is data-dependent and lacks flexibility. A reinforcement learning algorithm that combines face detection and target tracking technology for an improved facial and target detection accuracy.

Gao *et al.* (2015) proposed a supervised auto-encoder, that supervises auto-encoder in extract features and making face identification easier.

Ren *et al.* (2021) describe a four-phase process for developing a multi-camera face-tracking system. The first phase involves dividing the LAN into four layers. The first three layers for face data acquisition and transmission and the fourth layer is dedicated to recognition and tracking. The YOLOv3 and WIDER FACE datasets are used to collect facial data, specifically targeting small faces and the CW clustering technique is employed to cluster face data from the same camera. Siam16, built on VGG16's Double Triplet Networks, achieves multi-camera tracking. In another study by Khan *et al.* (2017), the authors highlight the challenges of automatic gender classification, particularly in unconstrained situations where facial photo variations are significant. In another study, it is highlighted the importance of automatically determining gender and facial expression in various applications.

In response to the problem that emerges when the present level set approaches segment images with intricate backgrounds or intensity inhomogeneity, Wu *et* *al.* (2017) suggested a method based on prior form. To learn a previous shape that satisfied both global and local deformations, the technique used a Deep Boltzmann Machine.

Despite the significant progress in face recognition techniques, several challenges persist. In addition to primary challenges like pose, expression concerns related to privacy, bias, fairness, and security have prompted researchers to explore ethical considerations and mitigation strategies. Future research directions include investigating multi-modal approaches that integrate additional biometric modalities such as iris recognition and fingerprint recognition, as well as exploring generative models for face synthesis and augmenta-tion.

In conclusion, face recognition techniques have evolved from traditional handcrafted feature-based methods to sophisticated deep learning approaches. The advent of deep learning has significantly improved the accuracy and robustness paving the way for applications in varied domains.

Materials and Methods

This section deals with model design, face detection and tracking, algorithm development in detail.

Model Designing

Designing a mathematical model for deep learning involves defining the architecture, loss function, optimization process, training procedure, evaluation metrics, hyper parameter tuning, cross-validation, model selection, interpretability, and deployment strategies. By systematically designing and evaluating deep learning models, we can build robust and efficient models for various applications like person re-identification shown in Fig. 1.

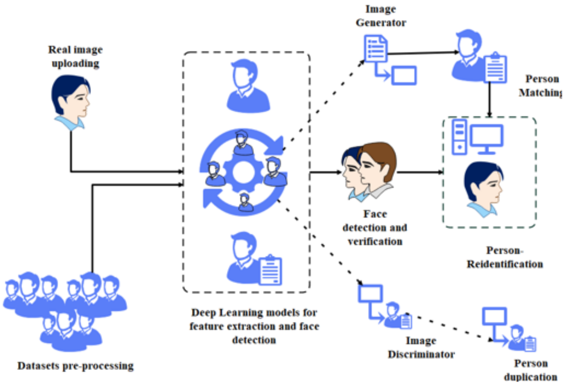


Fig. 1: Structured Framework for Person Re-Identification using face detection

Face Detection and Tracking

In the Bayes filtering, A state transition model that describes the state transition from the previous state is needed for prediction $x_{t-1}$to the current state x_t and Measurement and model to update the measurement. Mathematical relationship between the current state x_t and current measurement z_t. Transition model and the Measurement model for the Linear model (Julier & Uhlmann, 1997) which can be written as:

$$Transition\ model\colon x_t = Ax_{t-1} + Bu_t + \epsilon_t \tag{1}$$

$$Measurement\ model\colon z_t = Cx_t + \delta_t \tag{2}$$

where, ϵ_t and δ_t are Gaussian noises: $\epsilon_t \sim \mathcal{N}(0, R_t)$ and $\delta_t \sim \mathcal{N}(0, Q_t)$ and A, B and C are matrices.

The state transition probability $p(x_t|x_{t-1}, u_t)$ and the measurement probability $p(z_t|x_t)$ can be computed respectively:

$$\begin{aligned} &p(x_t|x_{t-1}, u_t) = \mathcal{N}\left(Ax_{(t-1)} + Bu_t, R_t\right) \\ &p(z_t|x_t) = \mathcal{N}(Cx_t, Q_t) \end{aligned} \tag{3}$$

Then the predicted posterior $\hat{prob}(x_t)$ can be calculated as

$$\begin{aligned} &prediction\colon \hat{prob}(x_t) = p(x_t|z_{1:\,t-1}, u_{1:\,t}) \\ &= \int p(x_t, x_{t-1}|z_{1:\,t-1}, u_{1:\,t})\, dx_{t-1} \\ &= \int p(x_t|x_{t-1}, z_{1:\,t-1}, u_{1:\,t})\, p(x_{t-1}|z_{1:\,t-1}, u_{1:\,t})\, dx_{t-1} \\ &= \int p(x_t|x_{t-1}, u_t)\, prob(x_{t-1})\, dx_{t-1} \end{aligned} \tag{4}$$

Algorithm 1 illustrates the entire Linear model.

```
Algorithm 1. Linear model

Initialization
    $\mu_0 = E(x_0), P_0 = E\left((x_0 - \mu_0)(x_0 - \mu_0)^T\right)$
For $t = 1, \ldots, \infty$:
1. Prediction:
    $\hat{\mu}_t = A\mu_{t-1} + Bu_t$
    $\hat{P}_t = AP_{t-1}A^T + R_t$
2. Measurement update
    $K_t = \hat{P}_t C^T \left(C\hat{P}_t C^T + Q_t\right)^{-1}$
    $\mu_t = \hat{\mu}_t + K_t(z_t - C\hat{\mu}_t)$
    $P_t = (1 - K_t C)\hat{P}_t$                                  (4)
```

In certain situations, however, this is not the case. The Extended Linear model is a nonlinear version that we introduce in this section. The following are the definitions of the nonlinear transition probability model and measurement model:

$$\begin{aligned} &TransistionModel\colon x_t = g(x_{t-1}, u_t) + \epsilon_t \\ &MeasurementModel\colon z_t = h(x_t) + \delta_t \end{aligned} \tag{5}$$

where g and h are nonlinear functions.

The extended Linear model key concept is to use first order Taylor expansion to linearize nonlinear functions:

$$\begin{aligned} &g(x_{t-1}, u_t) \approx g(\mu_{t-1}, u_t) + G_t(x_{t-1} - \mu_{t-1}) \\ &h(x_{t-1}) \approx h(\hat{\mu}_t) + H_t(x - \hat{\mu}_t) \end{aligned} \tag{6}$$

Where $G_t = \frac{\partial g(x_{t-1},\mu_t)}{\partial x_{t-1}} | \left(x_{t-1} = \mu_{t-1}\right)$ and $H_t = \frac{\partial h(x_t)}{\partial x_t} | \left(x_t = \hat{\mu}_t\right)$ denote the first order of derivatives, algorithm 2 presents the process of Extended Linear algorithm.

Algorithm 2. Extended Linear model

```
Initialization: $\mu_0 = E\left(\left(x_0 - \mu_0\right)\left(x_0 - \mu_0\right)^T\right)$
For $t = 1, \ldots, \infty$:
1. Prediction:
    $\hat{\mu}_t = g\left(\mu_{t-1}, u_t\right)$
    $P_t = G_t P_{t-1} G_t^T + R_t$
2. Measurement:
    $K_t = \hat{P}_t H_t^T \left(H_t \hat{P}_t H_t^T + Q_t\right)^{-1}$
    $\mu_t = \hat{\mu}_t + K_t\left(z_t - h\left(\hat{\mu}_t\right)\right)$
    $P_t = \left(I - K_t H_t\right)\hat{P}_t$                    (7)
```

The Extended Linear model employs the first-order Taylor expansion to linearize nonlinear functions, but this approach may result in significant estimation errors when dealing with highly nonlinear systems. To overcome this limitation, introduced the Unscented Filter (Shankaranarayanan *et al.*, 2016) by incorporating the unscented transform into the filter scheme. Unlike the Taylor series, the unscented transform can linearize nonlinear functions up to the second order, thereby enabling the Unscented Filter to generate more accurate outcomes than the first-order Extended Linear model. This section introduces the unscented transform and subsequently presents the derivation of the Unscented Filter algorithm.

For a nonlinear model $y = g(x)$ where x is a L dimensional variable and meets $x \sim \mathcal{N}\left(\mu, P_x\right)$, a set of 2L + 1. The following rule is used to choose weighted sigma points algorithmically:

$$\mathcal{X}_i = \begin{cases} \mu, i = 0 \\ \mu + \left(\sqrt{\left(\lambda + L\right) P_x}\right)_i, i = 1, \ldots L \\ \mu - \left(\sqrt{\left(\lambda + L\right) P_x}\right)_i, i = L + 1, \ldots 2L \end{cases} \tag{8}$$

Where $\lambda = \alpha^2\left(L + k\right) - L$, α and k are scaling parameters that specify the distribution of sigma points from the mean μ. These sigma points $\mathcal{X}_i$ after the nonlinear machine function g propagates:

$$y_i = g\left(\mathcal{X}_i\right) \tag{9}$$

The mean μ_y, covariance P_y and cross-covariance P_{xy} of the variable y are extracted from the output sigma points Y_i as follows:

$$\begin{aligned} \mu_y &= \textstyle\sum_{i=0}^{2L} w_i^m y_i \\ P_y &= \textstyle\sum_{i=0}^{2L} w_i^c \left(y_i - \mu_y\right)\left(y_i - \mu_y\right)^T \\ P_{xy} &= \textstyle\sum_{i=0}^{2L} w_i^c \left(\mathcal{X}_i - \mu\right)\left(y_i - \mu_y\right)^T \end{aligned} \tag{10}$$

Finally, the Unscented Filter is calculated on the state transformation using the unscented transform function g.

Algorithm 3. Unscented Filter

```
Initialization:
    $\mu_0 = E\left(x_0\right), P_0 = E\left(\left(x_0 - \mu_0\right)\left(x_0 - \mu_0\right)^T\right)$
    at $t = 1, \ldots, \infty$:
1. Create sigma points for prediction:
    $\mathcal{X}_{t-1} = \left[\mu_{t-1}\mu_{t-1} + \gamma\sqrt{P_{t-1}}\mu_{t-1} - \gamma\sqrt{P_{t-1}}\right]$
2. Prediction:
    $\mathcal{X}_{t|t-1} = g\left(u_t, \mathcal{X}_{t-1}\right)$
    $\hat{\mu}_t = \sum_{i=0}^{2L} w_i^m \mathcal{X}_{i,t|t-1}$
    $\hat{P}_t = \sum_{i=0}^{2L} w_i^c \left(\mathcal{X}_{i,t|t-1} - \hat{\mu}_t\right)\left(\mathcal{X}_{i,t|t-1} - \hat{\mu}_t\right)^T +$
    $R_t$
3. Create sigma points for measurement update:
    $\hat{\mathcal{X}}_t = \left[\hat{\mu}_t, \hat{\mu}_t + \gamma\sqrt{\hat{P}_t}\hat{\mu}_t - \gamma\sqrt{\hat{P}_t}\right]$
4. Measurement update equations:
    $\hat{\mathcal{Z}}_t = h\left(\hat{\mathcal{X}}_t\right)$
    $\hat{P}_{z_t} = \sum_{i=0}^{2L} w_i^c \left(\hat{\mathcal{Z}}_{i,t} - \hat{z}_t\right)\left(\hat{\mathcal{Z}}_{i,t} - \hat{z}_t\right)^T + Q_t$
    $\hat{P}_{x_{tz_t}} = \sum_{i=0}^{2L} w_i^c \left(\mathcal{X}_{i,t} - \hat{\mu}_t\right)\left(\hat{\mathcal{Z}}_{i,t} - \hat{z}_t\right)^T$
    $K_t = \hat{P}_{x_{tz_t}}\hat{P}_{z_t}^{-1}$
    $\mu_t = \hat{\mu}_t + K_t\left(z_t - \hat{z}_t\right)$
    $P_t = \hat{P}_t - K_t\hat{P}_{z_t}K_t^T$                    (11)
```

Color space models for display and computer graphics, such as RBG and HSV, have been widely used in a range of online and offline applications. YCbCr colour spaces are commonly used in video coding and storage, and they may also be utilised to extract information about skin colour. However, for high-complexity and occlusion situations, these techniques fall short. As a result, we provide a face identification and tracking approach. This pattern is used for video clips. The prediction can be expressed as using the Kalman filter:

$$X\left(k\right) = A\left(k-1\right)X\left(k-1\right) + W\left(k\right) \tag{12}$$

$$Z\left(k\right) = H\left(k\right)X\left(k\right) + V\left(k\right) \tag{13}$$

Where *X(k)* denotes the state vector, *Z(k)* is the observation vector at the time *t(k)*. Similarly, *A(k-1)* is the state transition observation given as *H(k)*.

Person Re-identification using Face Recognition Model: Deep Learning

The input picture component representation and posture fluctuations must be extensively investigated and to develop a stable model this manuscript proposes a path based approach to extract information using average pooling.

For an eight-pixel neighborhood, the operator is applied by comparing the intensity of the center pixel with the intensity of its eight neighbors. If a pixel's value is greater than or equal to that of the central point in an 8-bit neighborhood, "1" is applied to the current row that represents the neighborhood texture in relation to the central point. Otherwise "0" is applied to the corresponding bit. Instead of eight neighborhoods, the LBP operator can be expanded to any number of them. The LBP binary code is obtained by adding these weighted bits together as:

$$LBP_{N,R}\left(C\right) = \sum_{n=0}^{N-1} s\left(I_n - I_c\right).2^n \tag{14}$$

All codes produced by rotating a particular code are mapped to a specific reference code in this way. This mapping is seen as:

$$LBP_{P,R}^{ri} = \min\left\{ROR\left(LBP_{P,R}^{ri}\right) | i = 0, 1, 2, \ldots, P-1\right\} \tag{15}$$

function ROR (x, i) is described as:

$$ROR\left(x, i\right) = \begin{cases} \sum_{k=1}^{P-1} 2^{k-1} a_k + \sum_{k=0}^{i-1} 2^{P-i+k} a_k i > 0 \\ x, i = 0 \\ ROR\left(x, P+i\right), i < 0 \end{cases} \tag{16}$$

The appearance of homogeneous patterns is achieved by the use of image histograms.

$$\mathcal{H}_i = \sum_{x,y} I\left(f\left(x, y\right) = i\right), i = 0, \ldots n-1 \tag{17}$$

The option of a non-linear activation function causes the non-linearity. An FFN's aim is to approximate a nonlinear function g. A feed-forward network (FFN) describes a mapping *y=h(x; W)* and learns the value of the parameters *W* called weights, as a result of which the best function approximation is obtained.

Consider the simple network, which is given an input vector $x=(x_1,x_2)^T$, the input units activations are set to:

$$\begin{aligned} a_1 &= x_1 \\ a_2 &= x_2 \end{aligned} \tag{18}$$

The hidden units are given by formula:

$$a_3 = f\left(w_{1,3}a_1 + w_{2,3}a_1\right) \tag{19}$$

$$a_4 = f\left(w_{1,4}a_1 + w_{2,4}a_2\right) \tag{20}$$

The output units are given as:

$$a_5 = f\left(w_{3,5}a_3 + w_{4,5}a_4\right) \tag{21}$$

$$a_6 = f\left(w_{3,6}a_3 + w_{4,6}a_4\right) \tag{22}$$

In vector notation, we can rewrite the equations. The activation function f has to be a vector function that is applied element-by-element to vector member.

Equations 23 and 24 are:

$$f\left(\begin{pmatrix} w_{1,3} & w_{2,3} \\ w_{1,4} & w_{2,4} \end{pmatrix} \binom{a_1}{a_2}\right) = \binom{f(w_{1,3}a_1 + w_{2,3}a_2)}{f(w_{1,4}a_1 + w_{2,4}a_2)} = \binom{a_3}{a_4} \tag{23}$$

$$f\left(\begin{pmatrix} w_{3,5} & w_{4,5} \\ w_{3,6} & w_{4,6} \end{pmatrix} \binom{a_3}{a_4}\right) = \binom{f(w_{3,5}a_3 + w_{4,6}a_4)}{f(w_{3,6}a_3 + w_{4,6}a_4)} = \binom{a_5}{a_6} \tag{24}$$

Denote W_1 a matrix of weights of the layer 1 to the layer 2, W_3 a matrix of weights of the layer 2 to the layer 3 and x a vector of input features:

$$W_1 = \begin{pmatrix} w_{3,5} & w_{4,5} \\ w_{3,6} & w_{4,6} \end{pmatrix} \tag{25}$$

$$W_2 = \begin{pmatrix} w_{1,3} & w_{2,3} \\ w_{1,4} & w_{2,4} \end{pmatrix} \tag{26}$$

$$x = \binom{x_1}{x_2} \tag{27}$$

Finally, in the compact form:

$$H(x; W_2, W_1) = f(W_2 f(W_1 x)) \tag{28}$$

The pooling function replaces rectangular input areas with their summaries. It's a kind of non-linear down sampling technique. Let $[a_{i,j}] = A \in \mathbb{R}^{n\times m}$be a real matrix that represents a feature map area and is passed through a pooling function. The widely used pooling techniques are as follows:

Max pooling, which takes the maximum value of the input region and substitutes it:

$$p\left(A\right) = \max\left(A\right) = \max\left(\{a_{i,j} | i \in 1, \ldots., n, j \in 1, \ldots, m\}\right) \tag{29}$$

Average pooling and weighted average pooling calculated an average or a weighted amount based on the distance from the area's center to aggregate the input field:

$$p\left(A\right) = \frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} a_{i,j} \tag{30}$$

$L2$ process of pooling calculates the $L2$ the norm of the vector formed by "unfolding" the input area:

$$p\left(A\right) = \sum_{i=1}^{n} \sum_{j=1}^{m} a_{i,j^2} \tag{31}$$

The performance of the layer until it is normalized by the batch normalization layer. Let $X \subseteq \mathbb{D}$ be a batch of inputs, then:

Algorithm 4. Computation of output y of the batch normalization error

```
$\mu \leftarrow \frac{1}{|x|} \sum_{x \in x} x$
$\sigma^2 \leftarrow \frac{1}{|x|} \sum_{x \in x} \left(x - \mu\right)^2$
$\mathfrak{x} \leftarrow \frac{x_1 - \mu}{\sqrt{\sigma^2 + \varepsilon}}$
$y = \gamma x + \beta$                                        (32)
```

False positive bounding boxes for identification can be caused by the dynamic and constantly changing context. As a result, context removal is a promising task for improving detection accuracy. We can use any video or still image to complete this mission. Let's assume that the video-frame is represented as one-column vector, denoted $f \in \mathbb{F}^N$ where $N = R \times C$. The video frames described below are as follows: as$\{f_j\}, j = 1, 2, \ldots K$ .Vertically, this is expressed as:

$$y = [x_1, x_2, \dots x_k]^T \in \mathbb{F}^{N \times k} \tag{33}$$

Since these frames were taken from the same film, they are linked together. The sum of the two common components can be seen as the interpretation of each frame (background and foreground) components as well as:

$$f_j = z^c + z_j^i \tag{34}$$

z^c denotes as background part and z_j^i shown as the foreground element. Let $\gamma \in \mathbb{F}^{N \times N}$ is an orthonormal basis matrix that is used to represent the input image in dense form, and the coefficients $\alpha_j = \gamma f_j \in \mathbb{F}^N$ of the given input alert f_j as:

$$\alpha_j = \beta^c + \beta_j^i = \gamma z^c + \gamma z_i^c \tag{35}$$

β^c a similar context is present in the picture, the result is unchanged and β_j^i is Continually changing in response to the $j = 1, 2, \dots k$. As shown in (2), the common component is :

$$w = \left[\theta^c \theta_1^i \theta_2^i \dots \theta_k^i\right]^T \in \mathbb{F}^{N(k+1)} \tag{36}$$

The main aim here is to collect as much information as possible about the foreground data and describe it as accurately as$w$ in a dense manner of $\{f_j\}$, thus theframes can be interpreted in a dense manner in the following way:

$$y = \overline{\gamma} w \tag{37}$$

Where $\overline{\gamma} = [I_1 I_2]$ where I_1 is denoted as $\left[\gamma^T \gamma^T \dots \gamma^T\right]^T \in \mathbb{F}^{(kN) \times N}$ and I_2 is denoted as $diag\,(I_1) \in \mathbb{F}^{(kN) \times kN}$. The best solution for the foreground/common portion can be found by using l_1 minimization problem. This can be described as:

$$min_w \tfrac{1}{2} \left\|y - \overline{\gamma} w\right\|_2^2 + \lambda \left\|w\right\|_1 \tag{38}$$

By using the inverse transform, the common and creative components can be recovered at this point as:

$$z = \lambda w \tag{39}$$

Where $\lambda = diag\left(\left[\gamma^T \gamma^T \dots \gamma^T\right]\right) \in \mathbb{F}^{k(N+1) \times k(N+1)}$. The problem of minimization described above is solved using the Separable Surrogate method and z can be used to achieve the desired result.

To discriminate between face and non-facial models, the learning process is viewed as a two-class issue. Let's take that into account for x_i, a cross entropy loss as:

$$L_i^{det} = -\left(\left(1 - y_i^{det}\right)\left(1 - \log\left(p_i\right)\right) + \left(y_i^{det} \log\left(p_i\right)\right)\right) \tag{40}$$

Where p_i is likelihood of sample as face image.

We concentrate on bounding box regression after defining the face. We consider a candidate part in this step and predict bounding box are identical to or close to the ground truth labels. For each sample, the problem in the context of a regression problem and the Euclidean loss function is used. This loss function is written as follows:

$$L_i^{box} = \left\|\hat{y}_i^{box} - y_i^{box}\right\|_2^2 \tag{41}$$

$\hat{y}_i^{box}$ represents the coordinated part and y_i^{box} represents the truth coordinates of face. Likewise, a recognition procedure is used in which a regression problem is formulated and the Euclidean loss is maintained as follows:

$$L_i^{box} = \left\|\hat{y}_i^{landmark} - y_i^{landmark}\right\|_2^2 \tag{42}$$

We allocate different tasks of the CNN in this phase and in this learning process, there are various types of training images. The ultimate learning goal:

$$\min \textstyle\sum_{l=1}^{N} \sum_{j \in \{det, box\}} \alpha_j \beta_i^j L_i^j \tag{43}$$

Where N is the total amount of training samples.

The face recognition CNN model is presented in this segment, which includes deep feature learning. The function learning is done on the landmark points that have been defined. The input layer, according to the proposed architecture, is dimensioned as 100x100x1 or any other picture you have as an input. We used ten convolutional layers, one completely connected layer and five pooling layers in this network, with every convolutional layer receiving a Rectified Linear Unit (ReLU) excluding the final convolutional layer. To enhance efficiency, we provide parametric ReLU as well as additional convolutional layers, Conv-12 and Conv-22, that help to minimize the effects of illumination variations. We use a 3x3 filter in this study, with the first pooling layers using the peak pooling operator and the last three layers using average pooling. These dimensional features provide strong discriminative details of testing face images, which are collected from the face recognition module, in order to obtain successful classification. Before processing feature learning. Since there are multiple frames in a video sequence, the average function of the pool5 features is used to reflect the feature of the whole sequence. To learn the function, we employ a Bayesian learning procedure i^{th} and j^{th} images are Gaussian distribution was used to model the data directly as a joint distribution. Consider the following representation of the combined distribution of these images as $P\left(x_i, x_j | H_I\right) \sim N\left(0, \Sigma_I\right)$ in the event that the input images are x_i and x_j If the images are from the same class and the Gaussian distribution is described as $P\left(x_i, x_j | H_E\right) \sim N\left(0, \Sigma_E\right)$.The log-likelihood ratio between inter and intra groups is calculated using a Gaussian distribution as:

$$r\left(x_i, x_j\right) = \log \frac{P(x_i, x_j | H_I)}{P(x_i, x_j | H_E)} = x_i^T G x_i + x_i^T G x_i - 2 x_i^T R x_i \tag{44}$$

Where G and R and the two semi-definite matrices. The face vector can be modeled as in this learning method $x = \mu + \xi$ where μ denotes the identified vector and ξ denotes the pose, and illumination variation. These two parameters are added together to form a Gaussian distribution with a zero mean as $N(0, S_\mu)$ and $N(0, S_\xi)$. Here, the estimation of S_μ and S_ξ by maximizing the distance as:

$$\underset{G,B,b}{\text{argmin}} \sum_{i,j} \max \left[1 - y_{i,j}\left(b - \left(x_i - x_j\right)^T G\left(x_i - x_j\right) + 2x_i^T B x_j\right), 0\right] \tag{45}$$

$y_{i,j}$ is the pair such as $y_{i,j} = 1$ if x_i and x_j are the same individual, otherwise $y_{i,j} = -1$.

In this study, we employed a NN approach utilizing convolution layers to identify, map, and recognize faces. Our proposed method begins with a context removal model that facilitates feature extraction and learning. In addition, we have developed a CNN-based structure for identifying the facial area and a tool for bounding box regression. We employed a CNN-based model for feature learning to train the features, for inter and intra-features. Through a comprehensive experiment, we demonstrated proposed solution with superior performance.

Feature Extraction Architectures

Real-time face detection and verification using a CNN-based scheme is described. The introduction of AlexNet brought about a significant breakthrough in image classification, leading to the widespread adoption of Convolutional Neural Networks (CNNs) as an effective machine learning technique. This architectural design represents a remarkable advancement in deep learning for addressing image classification tasks. The initial convolutional layer applies an 11x11 filter scale with a stride of 4, while the subsequent convolutional layers use a 3x3 filter size. The architecture employs max-pooling as the subsampling layer.

AlexNet integrates overlapping pooling, which means that the strides of the pooling layer are smaller than the pooling filters. By adopting this approach, the top-1 and top-5 error rates are reduced by 0.4% and 0.3%, compared to the use of local pooling, where the stride size matches the pooling size. VGG conducted a research on the depth of convolutional networks in order to investigate the influence of network depth on the accuracy and precision of large-scale image recognition and categorization. VGG employs a 3x3 convolution kernel for all layers to increase the number of network layers without introducing excessive parameters.

The VGG network requires RGB images of size 224x244 as input. To create the input, the RGB values of all images in the training dataset are combined into a single image. This aggregated image is then fed into the VGG convolutional network. The convolution stage of the network can be configured with either a 3x3 or 1x1 filter. VGG11 consists of at least eight convolutional layers and three fully connected layers.

VGG19 can have up to 16 convolutional layers. For more complex problems, additional layers are often added to the VGG network. These extra layers allow the network to learn more complex features over time, improving precision and performance. However, training DNNs with many layers can be challenging. To address this challenge, residual networks (ResNets) have been introduced. ResNets are designed to make training intense networks easier.

We initially observe a direct connection that bypasses multiple intermediate layers (the specific number of layers may differ based on the model). The skip connection, also referred to as the center of residual blocks, establishes a relationship. The presence of this skip connection impacts the performance of the layer. Without the skip connection, the input 'x' undergoes multiplication with the layer weights, followed by adding a bias term.

The activation function is then applied to this concept, *f(x)* and we get our output as *H(x)*:

$$H(x) = f(wx + b) \tag{46}$$

The performance has increased since the implementation of the skip relation:

$$H(x) = f(x) + x \tag{47}$$

A potential issue arises when the structure of the input and output differs, particularly in the case of convolutional and pooling layers. To address this, we can employ one of two techniques. The first involves padding the skip relation with additional zero entries to increase its dimensions. The second technique, the projection method, adds 11 convolutional layers to the data to match the dimensions. In this scenario, the outcome is as follows:

$$H(x) = f(x) + w_1.x \tag{48}$$

GoogLeNet differs significantly from AlexNet and ZF-Net as it were previously advanced architectures. It uses a series of methods with 11 convolution and global average pooling, to produce deeper architecture.

Various image processing techniques such as histogram modification, wang filter, linear contrast correction, and Contrast Limited Adaptive Histogram Equalization (CLAHE) can be used. Additionally, video frames can be affected by issues like motion blur, out-of-focus blur, jitter, and occlusions. To create a face representation for the simple template, a straightforward approach is to aggregate the features obtained from the extracted frames. This aggregation allows for the classification of faces from video frames. Face embedding's, calculated using a facial representation model, play a crucial role in this process. This model is denoted as:

$$p\left(f|\mathbb{I}^*, \mathbb{F}^*\right) = \int p\left(f|i, \mathbb{I}^*, \mathbb{F}^*\right) p\left(f|i, \mathbb{I}^*, \mathbb{F}^*\right) di \tag{49}$$

The selection of photos for training, denoted by 𝕀*= *{i1,i2, iM }*, and the corresponding training data features, denoted by *F*={f1,f1, fM}*, are also referred to as noisy training data embeddings. The expression *p(f I, I*, F*)* represents the uncertainty in estimating the function embedding, while *p(f I*, F*)* represents the probability density of the face image in the noiseless embedding. In this context, we use function φ to map face images to the appropriate embedding. Let's consider a template *T= {i1,i2, iN}* that contains *N* images representing a single identity. The number of *N* images allows us to approximate the noiseless attribute embedding using an expectation function, it can be very high $\hat{E}(F^T)$:

$$\varphi \approx \hat{E}\left(f^T\right) = \sum_{i=1}^{N} p\left(f|\mathbb{I}^*, \mathbb{F}^*\right) f_i \tag{50}$$

$$r^T = \sum_{i=1}^{N} g\left(f_i\right) f_i \tag{51}$$

Where r^T introduced the templates and $g(f_i)$ indicates the expected weights for the attributes of i^{th} image in the template under consideration *T*. This method aids in the reduction of attribute noise.

Consider the case where we have *n* number of faces from video data as $\left(\mathcal{X}^i, y_i\right)_{i=1}^{n}$ where $\mathcal{X}^i$ denotes the video sequence with varying number of images K_i as $\mathcal{X}^i = \left\{x_1^i, x_2^i, \ldots x_{K_i}^i\right\}$ where x_k^i represents the kth frame in the current video and y_i denotes the label. We start by extracting the function for each frame x_i^k from the function embedding module, which is denoted as f_k^i . The main goal here is to create a collection of linear weights $\{a_k\}_{k=1}^{K}$ Features extracted from the feature embedding module are used for each video. As a result, the combined features can be written as:

$$r = \sum_k a_k f_k \tag{52}$$

This aids in the creation of a feature vector that is roughly the same size as a single face picture.

$$\begin{gathered} s_k = q^T f_k \\ a_k = \frac{exp(s_k)}{\sum_j exp(s_j)} \end{gathered} \tag{53}$$

The aggregation module is also improved by cascading two attention blocks. Consider the following: q^1 denotes the kernel of first attention block and r^0 denotes the aggregated features. Here, we calculate the q^2 using a transfer layer and aggregated features, the kernel of the secondary attention block is formed r^0. This can be denoted as:

$$q^2 = tanh\left(r^0 w + b\right) \tag{54}$$

Where W is weight matrix and b is the bias vector of neurons and $\tanh\left(x\right) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$. A hyperbolic tangent nonlinearity is used. We share the weights and calculate the loss using these functions. The main goal of network training is to reduce loss as:

$$l_{i,j} = y_{i,j}\|ri1 - rj1\|_2^2 + (1 - y_{i,j})\max(0, m - \|ri1 - rj1\|_2^2) \tag{55}$$

Where $y_{i,j} = 1$ and *m* is a constant with a value 2. During the training phase, we will train these models in both an end-to-end and one-by-one manner. In this study, we have opted to train the network on individual images, one by one, and subsequently train the aggregation module. Before ex-tracting features, face recognition and alignment are per-formed. The input image has a resolution of 224x224 pixels. Face detection and recognition are crucial components in vis-ual surveillance systems. This paper introduces a novel tech-nique for real-time facial detection and recognition in videos recorded by CCTV surveillance cameras and the features are extracted.

Results and Discussion

The introduced models for face detection, monitoring, and verification are evaluated. To begin, we'll go over the specifics of the dataset used in these techniques. In terms of different performance metrics, the observed performance using proposed techniques is compared to cutting-edge approaches.

Dataset Comparison

We present a detailed experimental setup in this section of review based on the proposed method. We used open-source video face recognition databases such as the IARPA Janus Benchmark A (IJB-A), the YouTube Face dataset, and the Celebrity-1000 dataset to test the output of the proposed method.

Table 1: Comparative analysis for IHB-A using Bayesian Learning Model

Method	FAR = 0.01 (%)	FPR = 0.1 (%)
LSFS (Wang *et al.*, 2015)	0.73	0.89
Triplet Similarity (Sankaranarayanan *et al.*, 2016)	0.79	0.94
Deep Multi-Pose (Liang *et al.*, 2022)	0.87	.95
DCNNfusion (Chen *et al.*, 2015)	0.83	0.96
Triplet Embedding (Sankaranarayanan *et al.*, 2016)	0.90	0.96
Proposed Model	0.92	0.97

Similarly, we assess the efficiency of the proposed solution in the 1: N case, as shown in Table 1, by comparing it to existing techniques.

Table 2: Comparative Analysis in terms of FPIR and FPR Bayesian Learning Model

Method	FPIR = 0.01 (%)	FPR = 0.1 (%)
LSFS (Wang *et al.*, 2015)	0.38	0.61
Triplet Similarity (Sankaranarayanan *et al.*, 2016)	0.55	0.75
Deep Multi-Pose (AbdAlmageed *et al.*, 2016)	0.52	0.75
DCNNfusion (Chen *et al.*, 2015)	0.57	0.79
Triplet Embedding (Chen *et al.*, 2016)	0.75	0.86
Proposed Model	0.78	0.89

The provided comparison in Tables 2 and 3 clearly illustrate that the proposed method outperforms various existing approaches including LSFS, DCNNmanual+metric, Triplet Similarity, Deep Milti-Pose, DCNNfusion, and Triplet Embedding. In this section, the performance of the YouTube face dataset, which is specifically designed for video face recognition, is evaluated. Table 3 presents a comparison of the face detection results obtained for the YouTube dataset.

Table 3: Comparative performance for Youtube dataset

Method	Accuracy (%)	AUC (%)
LM3L (Hu *et al.*, 2015)	81	89
DDML (combined) (Hu *et al.*, 2014)	82	90
DeepFace Single (Taigman *et al.*, 2014)	91	96
DeepID2+ (Sun *et al.*, 2015)	93	-
Wen *et al.* (Wen *et al.*, 2016)	94	-
CNN+Max. L2	91	97
CNN+Min. L2	94	98
CNN+MaxPool	88	95
Proposed Model	95	98

As shown in Table 4, regarding facial recognition precision and AUC, the suggested solution outperforms the competition. Compared to the CNN+MaxPool model, the average accuracy of the proposed model is 95.22, which is a 6.06% improvement. The Celebrity-1000 dataset is primarily concerned with the issue of video-based face recognition. This data comprises 159726 video sequences with 1000 human subjects and 2.4 million frames. With the results, this dataset offers two test protocols: open-set and close-set. Table 5 illustrates the performance of the suggested approach for close-set data, and it is compared to existing techniques on LTF and YTF datasets. To assess results, we looked at a variety of subjects and calculated rank-1 frequency.

Table 4: Comparative performance for Celebrity-1000 dataset

Method	100	200	500	1000
MTJSR (Yuan *et al.*, 2012)	50 %	40%	35%	30%
CNN+Mean L2	85%	77%	74%	67%
CNN+AvePool - VideoAggr	86%	82%	80%	74%
CNN+AvePool - SubjectAggr	84%	78%	77%	73%
Proposed Model	91%	86%	85%	82%

Table 5: Face Detection accuracy values on YTF and LFW Datasets

Method	Trainset	LFW	YTF
FaceNet (Tang *et al.*, 2020)	200M	99.7%	95.1%
DeepID2+ (Tang *et al.*, 2020)	0.2M	99.4%	93.2%
Center face (Wen *et al.*, 2016)	0.7M	99.2%	94.9%
A-softmax (Liu *et al.*, 2017b)	0.46M	99.4%	95%
Alignment learning (Identical) (Tang *et al.*, 2020)	0.46M	99.5%	96.2%
Alignment learning (Similarity) (Tang *et al.*, 2020)	0.46M	98.0%	93.4%
Alignment learning (Affine) (Tang *et al.*, 2020)	0.46M	98.8%	94.2%
Proposed model	0.46M	99.8%	98.6%

This study also aims to assess the efficiency of the end-to-end detection process by using CASIA-Web-Face as the training dataset. The main focus is on end-to-end learning and examining the effect of face alignment on recognition outcomes with various geometric transformation configurations. To accomplish this, alignment networks are trained using CASIA-Web-Face, and the model's performance is evaluated using a single patch function for recognition. Fig. 2 gives comparative performance of Celeb-1000 dataset.

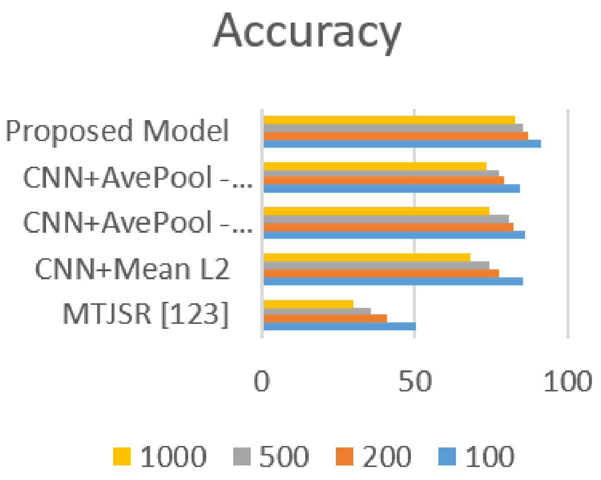


Fig. 2: Comparative performance of Celeb-1000 dataset

Table 6 illustrates Celebrity dataset recognition accuracy comparison in which proposed model has reported highest accuracy.

Table 6: Youtube celebrity dataset recognition accuracy comparison

Method	Accuracy (%)
Hidden Markov Model (HMM) (Kim *et al.*, 2008)	72
MDA (Wang & Chen, 2009)	68
SANP (Hu *et al.*, 2011)	69
COV+PLS (Wolf *et al.*, 2011)	72
UISA (Zhou *et al.*, 2020)	76
MSSSRC (Ortiz *et al.*, 2013)	84
Proposed model	93

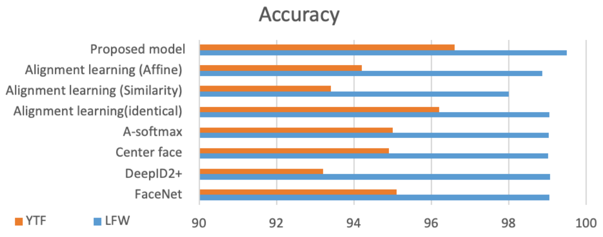


Fig. 3: Face detection accuracy on YTF and LFW datasets

This shows that the proposed model realizes better results than the competitors in facial recognition, identification, and monitoring. Figures 3 and 4 show the face detection and person recognition comparison on datasets.

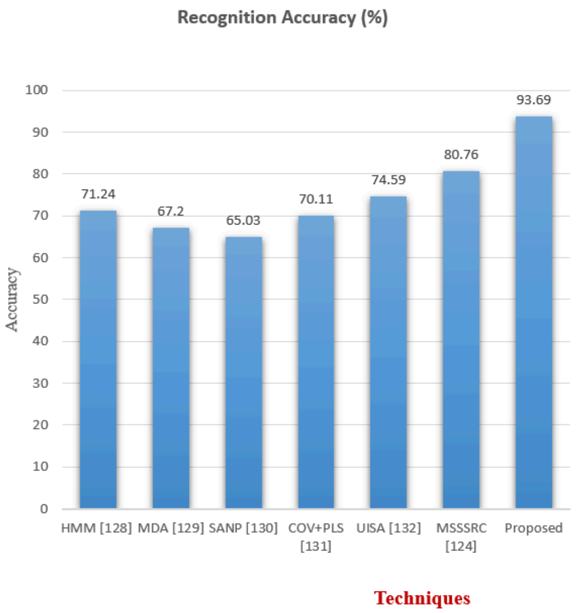


Fig. 4: Person Recognition Accuracy comparison for YouTube celebrity dataset

Table 7: Person detection and recognition in terms of FAR

Method	1:1 Verification TAR		
	FAR=0.001(%)	FAR=0.01(%)	FAR=0.1(%)
LSFS (Wang *et al.*, 2015)	0.51	0.73	0.89
DCNN (Chen *et al.*, 2015)	-	0.78	0.94
Triplet Similarity (Shankaranarayanan *et al.*, 2016)	0.59	0.79	0.94
Pose-aware models (Masi *et al.*, 2016)	0.65	0.82	-
Triplet Embedding (Masi *et al.*, 2016)	0.81	0.90	0.96
Template Adaption (Crosswhite *et al.*, 2018)	0.83	0.93	0.97
CNN + Max L2 (Yang *et al.*, 2017)	0.20	0.34	0.60
CNN + Min L2 (Yang *et al.*, 2017)	0.03	0.14	0.97
CNN + Mean L2 (Yang *et al.*, 2017)	0.68	0.89	0.97
CNN + Soft Min L2 (Yang *et al.*, 2017)	0.69	0.90	0.97
CNN + Max Pool (Yang *et al.*, 2017)	0.20	0.34	0.60
CNN + Avg Pool (Yang *et al.*, 2017)	0.77	0.91	0.97
NAN (Yang *et al.*, 2017)	0.88	0.94	0.97
Proposed Model	0.92	0.97	0.98

Here we present a study on the recognition of faces using the IJB-A dataset. The IJB-A dataset is composed of videos and face images that have been captured in different environments. Table 7 provides the results of the proposed method where True Acceptance Rate (TAR) & False Acceptance Rate (FAR) are compared with the existing models.

As compared to current approaches, the proposed solution produces better results, as seen in the table above. We used some of the comparative techniques described by Wang & Chen (2009), in which the experimental research is expanded by integrating L2 distance measurements with CNN architecture, such as CNN + Max L2, CNN + Min L2, CNN+Mean L2 and CNN+SoftMin L2 along with max and average pooling such as CNN+MaxPool and CNN+AvePool. As a result of these methods, efficiency is improved as 0.978±0.004 however, the suggested aggregation module aids in the reduction of noisy functions, thus improving the system's accuracy.

In this segment, we show the results of an experiment on the YouTube face database. There are 3425 videos in this data set, representing 2684 different individuals. The number of frames in these videos ranges between 59 and 7080.

Table 8: Person recognition performance for Youtube dataset

Method	Accuracy (%)	AUC (%)
LM3L (Hu *et al.*, 2015)	81	89
DDML (Hu *et al.*, 2014)	82	90
Eigen PEP (Li *et al.*, 2015)	84	92
Deep Face-single (Taigman *et al.*, 2014)	91	96
CNN + Max L2 (Yang *et al.*, 2017)	91	-
CNN + Min L2 (Yang *et al.*, 2017)	94	98
CNN + Mean L2 (Yang *et al.*, 2017)	95	98
CNN + Soft Min L2 (Yang *et al.*, 2017)	95	98
CNN + Max Pool (Yang *et al.*, 2017)	88	95
CNN + Avg Pool (Yang *et al.*, 2017)	95	98
NAN (Yang *et al.*, 2017)	95	98
Proposed Model	98	99

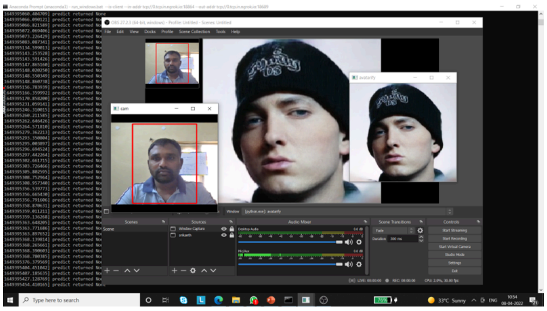
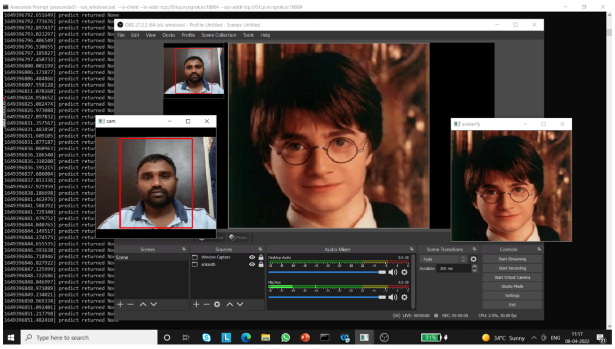


Fig. 5: Real-time person identification

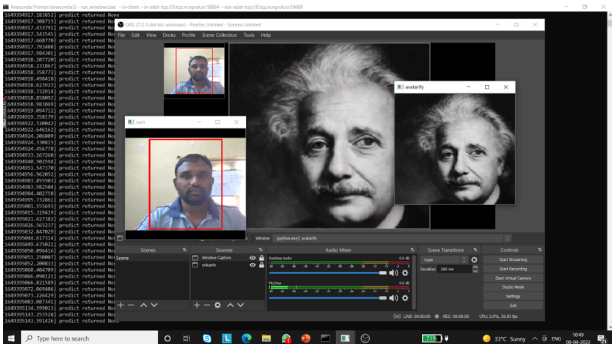
Prior to conducting facial recognition on the video, a feature vector is created with extraction of the features. In Table 8, a comparison of the efficiency of various methods for detecting faces in videos in terms of recognition accuracy and the area under the curve are presented (Yang *et al.*, 2017). Based on comparative analysis, the proposed method achieves a precision rate of 98.23%, demonstrating a significant improvement over existing technique.

From Figures 5, 6 and 7, our model is able to detect multiple persons on a single execution process and generating the results. And also, it is most suitable on both real-time and existing datasets, the major advantage is, it is also suitable on moving objects.



```
1649396792.651649] predict returned None
1649396792.773676] predict returned No
1649396792.897437] predict returned No
1649396793.023297] predict returned No
1649396796.406549] predict returned No
1649396796.530655] predict returned No
1649396797.185827] predict returned No
1649396797.450732] predict returned No
1649396800.001199] predict returned No
1649396806.171877] predict returned No
1649396806.484866] predict returned No
1649396807.558128] predict returned No
1649396811.070260] predict returned No
1649396824.958652] predict returned No
1649396825.082474] predict returned No
1649396826.973088] predict returned No
1649396827.097832] predict retur
1649396831.357567] predict retur
1649396831.483850] predict retur
1649396831.609105] predict retur
1649396831.877187] predict retur
1649396836.060963] predict retur
1649396836.186540] predict retur
1649396836.310280] predict retur
1649396836.591215] predict retur
1649396837.686084] predict retur
1649396837.811336] predict retur
1649396837.921959] predict retur
1649396838.186698] predict retur
1649396841.462976] predict retur
1649396841.588392] predict retur
1649396841.729340] predict retur
1649396841.979752] predict retur
1649396844.040765] predict retur
1649396844.149517] predict retur
1649396844.274575] predict retur
1649396844.655535] predict returned No
1649396846.593638] predict returned No
1649396846.718946] predict returned No
1649396846.827922] predict returned No
1649396847.125999] predict returned No
1649396848.722686] predict returned No
1649396848.846997] predict returned No
1649396848.971089] predict returned No
1649396849.234021] predict returned No
1649396850.969334] predict returned No
1649396851.092401] predict returned No
1649396851.217798] predict returned No
1649396851.482410] predict returned None
```

Fig. 6: Real-time face detection and person re-identification from playing videos (motion detection) (wearing goggles) and compared with datasets



```
1649394917.183852] predict returned None
1649394917.308715] predict returned No
1649394917.433791] predict returned No
1649394917.543545] predict returned No
1649394917.668770] predict returned No
1649394917.793408] predict returned No
1649394917.984301] predict returned No
1649394918.107720] predict returned No
1649394918.231867] predict returned No
1649394918.358772] predict returned No
1649394918.498418] predict returned No
1649394918.623923] predict returned No
1649394918.732914] predict returned No
1649394918.858092] predict returned No
1649394918.983869] predict returned No
1649394919.094712] predict returned No
1649394919.358179] predict returned No
1649394922.520042] predict returned No
1649394922.646162] predict returned No
1649394924.206809] predict returned No
1649394924.330815] predict returned No
1649394924.456778] predict returned No
1649394933.167260] predict returned No
1649394940.502594] predict returned No
1649394951.547370] predict returned No
1649394956.962052] predict returned No
1649394983.855503] predict returned No
1649394983.982504] predict returned No
1649394984.402758] predict returned No
1649394995.732861] predict returned No
1649395001.555693] predict returned No
1649395015.319419] predict returned No
1649395021.427382] predict returned No
1649395026.565237] predict returned No
1649395032.847029] predict returned No
1649395044.617314] predict returned No
1649395049.675021] predict returned No
1649395050.096416] predict returned No
1649395051.250007] predict returned No
1649395052.200833] predict returned No
1649395060.404709] predict returned No
1649395066.090121] predict returned No
1649395066.821589] predict returned No
1649395072.069406] predict returned No
1649395073.226429] predict returned No
1649395083.087341] predict returned No
1649395134.599013] predict returned No
1649395143.253528] predict returned No
1649395143.591426] predict returned None
```

Fig. 7: Real-time face detection from the image datasets - Old images (art images) and person re-identification

Conclusion

This study focuses on improving the efficiency of a face detection system by developing new methods. While current methods mainly concentrate on facial recognition in still images, the increasing demand for security applications requires the development of surveillance systems that can identify, monitor, and recognize multiple faces. To accomplish this, a technique based on deep learning is used to detect and track faces. A model for extracting combined features is then created to generate a database that has been trained. Additionally, a model for extracting TLBP features is employed to improve face detection accuracy. A scheme for Bayesian learning is developed for face recognition. A model for feature learning based on CNN is used to learn the features, utilizing the calculation of log-likelihood ratios between inter and intra-features. The GoogleNet architecture is utilized to develop the CNN. The softmax operator is employed to incorporate feature attention and aggregation models, processing substantial features. The network is designed using a one-by-one training process. For future work, integrating person re-identification techniques into the system can facilitate identification even in cases of occlusion. Additionally, implementing a face reconstruction approach based on 3D face reconstruction can improve accuracy for low-quality images.

Acknowledgment

Authors thank Principal, College of Engineering, Andhra University for supporting our work.

Funding Information

No funding was received for conducting this study.

Author's Contributions

Revathi Lavanya Baggam: Responsible for data collection, writing the original draft, and generating algorithms.

V. Valli Kumari: Contributed to conceptualization, methodology design, formal analysis, reviewing, and overall supervision.

References

AbdAlmageed, W., Wu, Y., Rawls, S., Harel, S., Hassner, T., Masi, I., Choi, J., Lekust, J., Kim, J., Natarajan, P., Nevatia, R., & Medioni, G. (2016). Face recognition using deep multi-pose representations. 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Placid, NY, USA. https://doi.org/10.1109/wacv.2016.7477555

Bhatt, R., Malik, S., Arora, R., Agarwal, G., Sharma, S., & Dhablia, A. (2023). Recognition of Criminal Faces From Wild VideosSurveillance System Using VGG-16 Architecture. 2023 International Conference on Data Science and Network Security (ICDSNS), Tiptur, India. https://doi.org/10.1109/icdsns58469.2023.10245450

Chen, H., Li, W., Gao, X., & Xiao, B. (2023). Novel Multi-Feature Fusion Facial Aesthetic Analysis Framework. *IEEE Transactions on Big Data*, *9*(5), 1302-1320. https://doi.org/10.1109/tbdata.2023.3255582

Chen, J.-C., Patel, V. M., & Chellappa, R. (2016). Unconstrained face verification using deep CNN features. 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Placid, NY, USA. https://doi.org/10.1109/wacv.2016.7477557

Chen, J.-C., Ranjan, R., Kumar, A., Chen, C.-H., Patel, V. M., & Chellappa, R. (2015). An End-to-End System for Unconstrained Face Verification with Deep Convolutional Neural Networks. 2015 IEEE International Conference on Computer Vision Workshop (ICCVW), Santiago, Chile. https://doi.org/10.1109/iccvw.2015.55

Crosswhite, N., Byrne, J., Stauffer, C., Parkhi, O., Cao, Q., & Zisserman, A. (2018). Template adaptation for face verification and identification. *Image and Vision Computing*, *79*, 35-48. https://doi.org/10.1016/j.imavis.2018.09.002

Dai, W., Wang, J., Ren, T., & Zhu, Z. (2022). Face Mask Recognition Based on YOLOv3-tiny. 2022 3rd International Conference on Electronic Communication and Artificial Intelligence (IWECAI), Zhuhai, China. https://doi.org/10.1109/iwecai55315.2022.00104

Ding, C., & Tao, D. (2015). Robust Face Recognition via Multimodal Deep Face Representation. *IEEE Transactions on Multimedia*, *17*(11), 2049-2058. https://doi.org/10.1109/tmm.2015.2477042

Fabarisov, T., Naik, V. G., Attar, A. A., & Morozov, A. (2023). Remedy: Automated Design and Deployment of Hybrid Deep Learning-based Error Detectors. *IECON 2023- 49th Annual Conference of the IEEE Industrial Electronics Society*, 1-8. https://doi.org/10.1109/iecon51785.2023.10312506

Fang, X., Duan, Y., Du, Q., Tao, X., & Li, F. (2023). Sketch Assisted Face Image Coding for Human and Machine Vision: A Joint Training Approach. *IEEE Transactions on Circuits and Systems for Video Technology*, *33*(10), 6086-6100. https://doi.org/10.1109/tcsvt.2023.3262251

Fu, R., Wu, T., Luo, Z., Duan, F., Qiao, X., & Guo, P. (2019). Learning Behavior Analysis in Classroom Based on Deep Learning. 2019 Tenth International Conference on Intelligent Control and Information Processing (ICICIP), Marrakesh, Morocco. https://doi.org/10.1109/icicip47338.2019.9012177

Gao, S., Zhang, Y., Jia, K., Lu, J., & Zhang, Yingying. (2015). Single Sample Face Recognition via Learning Deep Supervised Autoencoders. *IEEE Transactions on Information Forensics and Security*, *10*(10), 2108-2118. https://doi.org/10.1109/tifs.2015.2446438

Hayat, M., Bennamoun, M., & An, S. (2014). Learning Non-linear Reconstruction Models for Image Set Classification. 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA. https://doi.org/10.1109/cvpr.2014.246

Hou, X., & Zhang, F. (2021). The Improved CenterNet for Ship Detection in Scale-Varying Images. 2021 3rd International Conference on Industrial Artificial Intelligence (IAI), Shenyang, China. https://doi.org/10.1109/iai53119.2021.9619209

Hu, J., Lu, J., & Tan, Y.-P. (2014). Discriminative Deep Metric Learning for Face Verification in the Wild. 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA. https://doi.org/10.1109/cvpr.2014.242

Hu, J., Lu, J., Yuan, J., & Tan, Y.-P. (2015). *Large Margin Multi-metric Learning for Face and Kinship Verification in the Wild*. 252-267. https://doi.org/10.1007/978-3-319-16811-1_17

Hu, Y., Mian, A. S., & Owens, R. (2011). Sparse approximated nearest points for image set classification. *CVPR 2011*. 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Colorado Springs, CO, USA. https://doi.org/10.1109/cvpr.2011.5995500

Huai, W., & Zhuo, H. (2020). An Improved AlexNet Model with Multi-channel Input Images Processing for Human Face Feature Points Detection. *12th International Conference on Communication Software and Networks (ICCSN)*, 246-251,. https://doi.org/10.1109/ICCSN49894.2020.9139075

Jalalipour, S., Ayyalasomayjula, S., Damrah, H., Lin, J., Rekabdar, B., & Li, R. (2023). Deep Learning-Based Spatial Detection of Drainage Structures using Advanced Object Detection Methods. *2023 Fifth International Conference on Transdisciplinary AI (TransAI)*. https://doi.org/10.1109/transai60598.2023.00007

Julier, S. J., & Uhlmann, J. K. (1997). New extension of the Kalman filter to nonlinear systems. *SPIE Proceedings*. AeroSense '97, Orlando, FL, USA. https://doi.org/10.1117/12.280797

Khan, K., Mauro, M., Migliorati, P., & Leonardi, R. (2017). *Gender and Expression Analysis Based on Semantic Face Segmentation*. 37-47. https://doi.org/10.1007/978-3-319-68548-9_4

Kim, M., Kumar, S., Pavlovic, V., & Rowley, H. (2008). Face tracking and recognition with visual constraints in real-world videos. 2008 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Anchorage, AK, USA. https://doi.org/10.1109/cvpr.2008.4587572

King, C.-H., Wang, Y.-L., Lin, W.-Y., & Tsai, C.-L. (2022). Automatic Cephalometric Landmark Detection on X-Ray Images Using Object Detection. 2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI), Kolkata, India. https://doi.org/10.1109/isbi52829.2022.9761506

Li, H., Hua, G., Shen, X., Lin, Z., & Brandt, J. (2015). *Eigen-PEP for Video Face Recognition*. 17-33. https://doi.org/10.1007/978-3-319-16811-1_2

Li, K., & Chen, H. (2019). Implementation of Automatic Face Detection System Based on ARM. *IEEE International Conference on Signal, Information and Data Processing (ICSIDP)*, 1-4,. https://doi.org/10.1109/ICSIDP47821.2019.9173162

Li, Y., Sun, B., Wu, T., & Wang, Y. (2016). *Face Detection with End-to-End Integration of a ConvNet and a 3D Model*. 420-436. https://doi.org/10.1007/978-3-319-46487-9_26

Liang, A., Pathirage, C. S. N., Wang, C., Liu, W., Li, L., & Duan, J. (2015). Face Recognition Despite Wearing Glasses. 2015 International Conference on Digital Image Computing: Techniques and Applications (DICTA), Adelaide, Australia. https://doi.org/10.1109/dicta.2015.7371260

Liang, J., Wang, J., Quan, Y., Chen, T., Liu, J., Ling, H., & Xu, Y. (2022). Recurrent Exposure Generation for Low-Light Face Detection. *IEEE Transactions on Multimedia*, *24*, 1609-1621. https://doi.org/10.1109/tmm.2021.3068840

Liu, H., Lu, J., Feng, J., & Zhou, J. (2017a). Learning Deep Sharable and Structural Detectors for Face Alignment. *IEEE Transactions on Image Processing*, *26*(4), 1666-1678. https://doi.org/10.1109/tip.2017.2657118

Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., & Song, L. (2017b). SphereFace: Deep Hypersphere Embedding for Face Recognition. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI. https://doi.org/10.1109/cvpr.2017.713

Lollett, C., Kamezaki, M., & Sugano, S. (2023). Normalized Facial Features-Based DNN for a Driver's Gaze Zone Classifier Using a Single Camera Robust to Various Highly Challenging Driving Scenarios. 2023 IEEE Intelligent Vehicles Symposium (IV), Anchorage, AK, USA. https://doi.org/10.1109/iv55152.2023.10186697

Maleš, L., Marčetić, D., & Ribarić, S. (2019). A multi-agent dynamic system for robust multi-face tracking. *Expert Systems with Applications*, *126*, 246-264. https://doi.org/10.1016/j.eswa.2019.02.008

Masi, I., Rawls, S., Medioni, G., & Natarajan, P. (2016). Pose-Aware Face Recognition in the Wild. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA. https://doi.org/10.1109/cvpr.2016.523

Modak, G., Das, S. S., Miraj, M. A. I., & Morol, M. K. (2022). A Deep Learning Framework to Reconstruct Face under Mask. *7th International Conference on Data Science and Machine Learning Applications (CDMA), Riyadh, Saudi Arabia*, 200-205. https://doi.org/10.1109/CDMA54072.2022.00038

Mokhayeri, F., & Granger, E. (2020). A paired sparse representation model for robust face recognition from a single sample. *Pattern Recognition*, *100*, 107129. https://doi.org/10.1016/j.patcog.2019.107129

Nair, A. R., Charan, R., Krishna S, H., & Rohith, G. (2023). A Two-level authentication for Attendance Management System using deep learning techniques. 2023 International Conference on Signal Processing, Computation, Electronics, Power and Telecommunication (IConSCEPT), Karaikal, India. https://doi.org/10.1109/iconscept57958.2023.10170617

Ortiz, E. G., Wright, A., & Shah, M. (2013). Face Recognition in Movie Trailers via Mean Sequence Sparse Representation-Based Classification. 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA. https://doi.org/10.1109/cvpr.2013.453

Peng, B., & Gopalakrishnan, A. K. (2019). A Face Detection Framework Based on Deep Cascaded Full Convolutional Neural Networks. 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS), Singapore. https://doi.org/10.1109/ccoms.2019.8821692

Pham, H. X., Pavlovic, V., Cai, J., & Cham, T. (2016). Robust real-time performance-driven 3D face tracking. 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun. https://doi.org/10.1109/icpr.2016.7899906

Qi, R., Jia, R.-S., Mao, Q.-C., Sun, H.-M., & Zuo, L.-Q. (2019). Face Detection Method Based on Cascaded Convolutional Networks. *IEEE Access*, *7*, 110740-110748. https://doi.org/10.1109/access.2019.2934563

Ranganatha, S., & Gowramma, Y. P. (2016). A novel fused algorithm for human face tracking in video sequences. 2016 International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS), Bengaluru, India. https://doi.org/10.1109/csitss.2016.7779430

Ren, G., Lu, X., & Li, Y. (2021). A Cross-Camera Multi-Face Tracking System Based on Double Triplet Networks. *IEEE Access*, *9*, 43759-43774. https://doi.org/10.1109/access.2021.3061572

Ren, Z., Yang, S., Zou, F., Yang, F., Luan, C., & Li, K. (2017). A face tracking framework based on convolutional neural networks and Kalman filter. 2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS), Beijing. https://doi.org/10.1109/icsess.2017.8342943

Sankaranarayanan, S., Alavi, A., Castillo, C. D., & Chellappa, R. (2016). Triplet probabilistic embedding for face verification and clustering. 2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS), Niagara Falls, NY, USA. https://doi.org/10.1109/btas.2016.7791205

Sun, Y., Liang, D., Wang, X., & Tang, X. (2015). Deepid3: Face recognition with very deep neural networks. *ArXiv:1502.00873*.

Sun, Y., Wang, X., & Tang, X. (2015). Deeply learned face representations are sparse, selective, and robust. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA. https://doi.org/10.1109/cvpr.2015.7298907

Suri, A., Bhadauria, R. V. S., & Bansal, L. K. (2022). Survey on Methods of Face Mask Detection System. *International Mobile and Embedded Technology Conference (MECON)*, 576-581. https://doi.org/10.1109/MECON53876.2022.9751815

Swetha, L., Praiscia, A., & Juliet, S. (2022). Pain Assessment Model using Facial Recognition. 2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India. https://doi.org/10.1109/iciccs53718.2022.9788265

Taigman, Y., Yang, M., Ranzato, M., & Wolf, L. (2014). DeepFace: Closing the Gap to Human-Level Performance in Face Verification. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 1701-1708. https://doi.org/10.1109/cvpr.2014.220

Tang, F., Wu, X., Zhu, Z., Wan, Z., Chang, Y., Du, Z., & Gu, L. (2020). An end-to-end face recognition method with alignment learning. *Optik*, *205*, 164238. https://doi.org/10.1016/j.ijleo.2020.164238

Tao, Q.-Q., Zhan, S., Li, X.-H., & Kurihara, T. (2016). Robust face detection using local CNN and SVM based on kernel combination. *Neurocomputing*, *211*, 98-105. https://doi.org/10.1016/j.neucom.2015.10.139

Varanasi, L. V. S. K. B. K., & Dasari, C. M. (2022). A Novel Deep Learning Framework for Diabetic Retinopathy Detection. *IEEE 6th Conference on Information and Communication Technology (CICT)*, 1-5. https://doi.org/10.1109/CICT56698.2022.9997826

Vishwakarma, V. P., & Dalal, S. (2020). A novel non-linear modifier for adaptive illumination normalization for robust face recognition. *Multimedia Tools and Applications*, *79*(17-18), 11503-11529. https://doi.org/10.1007/s11042-019-08537-6

Wang, D., Otto, C., & Jain, A. K. (2015). Face search at scale: 80 million gallery. *ArXiv:1507.07242*.

Wang, P., Qiao, M., Li, X., Wang, H., & Li, K. (2018). Facial Expression Recognition Algorithm Using Shallow Residual Network for Individual Soldier System. 2018 Chinese Automation Congress (CAC), Xi'an, China. https://doi.org/10.1109/cac.2018.8623130

Wang, R., & Chen, X. (2009). Manifold Discriminant Analysis. *CVPR*.

Wang, Z. Q., & El Saddik, A. (2023). DTITD: An Intelligent Insider Threat Detection Framework Based on Digital Twin and Self-Attention Based Deep Learning Models. *IEEE Access*, *11*, 114013-114030. https://doi.org/10.1109/access.2023.3324371

Wen, Y., Zhang, K., Li, Z., & Qiao, Y. (2016). *A Discriminative Feature Learning Approach for Deep Face Recognition*. 499-515. https://doi.org/10.1007/978-3-319-46478-7_31

Wolf, L., Hassner, T., & Taigman, Y. (2011). Effective Unconstrained Face Recognition by Combining Multiple Descriptors and Learned Background Statistics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *33*(10), 1978-1990. https://doi.org/10.1109/tpami.2010.230

Wu, X., Zhao, J., & Wang, H. (2017). Face segmentation based on level set and deep learning prior shape. 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Shanghai. https://doi.org/10.1109/cisp-bmei.2017.8301981

Yang, J., Ren, P., Zhang, D., Chen, D., Wen, F., Li, H., & Hua, G. (2017). Neural Aggregation Network for Video Face Recognition. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI. https://doi.org/10.1109/cvpr.2017.554

Yang, M., Zhu, P., Liu, F., & Shen, L. (2015). Joint representation and pattern learning for robust face recognition. *Neurocomputing*, *168*, 70-80. https://doi.org/10.1016/j.neucom.2015.06.013

Yuan, X.-T., Liu, X., & Yan, S. (2012). Visual Classification With Multitask Joint Sparse Representation. *IEEE Transactions on Image Processing*, *21*(10), 4349-4360. https://doi.org/10.1109/tip.2012.2205006

Yuan, Y., Fu, X., Wang, G., Li, Q., & Li, X. (2022). Forgery-Domain-Supervised Deepfake Detection With Non-Negative Constraint. *IEEE Signal Processing Letters*, *29*, 2512-2516. https://doi.org/10.1109/lsp.2022.3193590

Zhang, B., & Bao, Y. (2022). Cross-Dataset Learning for Age Estimation. *IEEE Access*, *10*, 24048-24055. https://doi.org/10.1109/access.2022.3154403

Zhang, L., Wang, H., & Chen, Z. (2021). A Multi-task Cascaded Algorithm with Optimized Convolution Neural Network for Face Detection. *2021 Asia-Pacific Conference on Communications Technology and Computer Science (ACCTCS)*. https://doi.org/10.1109/acctcs52002.2021.00054

Zhao, Z., Zhang, H., Wang, L., & Huang, H. (2023). A Multimodel Edge Computing Offloading Framework for Deep-Learning Application Based on Bayesian Optimization. *IEEE Internet of Things Journal*, *10*(20), 18387-18399. https://doi.org/10.1109/jiot.2023.3280162

Zhou, Z., He, Z., Jia, Y., Du, J., Wang, L., & Chen, Z. (2020). Context prior-based with residual learning for face detection: A deep convolutional encoder-decoder network. *Signal Processing: Image Communication*, *88*, 115948. https://doi.org/10.1016/j.image.2020.115948