WJCMS

Wasit Journal of Computer and Mathematics Science

Journal Homepage: https://wjcm.uowasit.edu.iq/index.php/WJCM
e-ISSN: 2788-5879 p-ISSN: 2788-5887

Categorization of Celebrity Photos using Deep Feature Extraction and Machine Learning Approaches

Nisreen Ryadh Hamza [1], Farah Jawad Al-Ghanim [1,*], Salwa Shakir Baawi [2]

[1]Department of Computer Science, College of Computer Science and Information Technology, University of Al-Qadisiyah, Al-Diwaniyah, IRAQ

[2]Department of Computer Information Systems, College of Computer Science and Information Technology, University of Al-Qadisiyah, Al-Diwaniyah, IRAQ

*Corresponding Author: Farah Jawad Al-Ghanim

DOI: https://doi.org/10.31185/wjcms.341
Received 02 January 2025; Accepted 27 February 2025; Available online 30 March 2025

ABSTRACT: In feature extraction, images or videos are analyzed to identify facial features before pinpointing the exact target. Significant progress has been made in the realm of feature extraction through deep learning. With deep learning technologies, developers have created algorithms for facial analysis and recognition that enhance accuracy and effectiveness. Consequently, programmers and developers have started to implement facial recognition technology in a wide range of applications to address these challenges. The identification process is particularly difficult due to the abundance of unstructured datasets. In real-world scenarios, identification continues to be quite challenging despite various methods proposed so far. This paper aims to analyze celebrity categorization based on deep features under different conditions. Most algorithms are influenced by these problematic factors, whether intentionally or unintentionally. Six datasets are utilized in this paper: the Open Famous People Dataset, Celeb Identification Dataset, Bollywood Celebrity Localized Dataset (170), Football Player's Dataset, Mini_LFW, and Nepali Celeb Localized Dataset. Most of these images feature celebrities including actors, singers, players, actresses, politicians, and social figures from diverse nationalities and countries. Therefore, this paper offers a combination of results from classification methods: Multilayer Perceptron Classifier (MLP), Decision Tree Classifier (DT), Bootstrap Aggregation Decision Tree Classifier (TreeBagger), Extreme Gradient Boosting Classifier (XGB), and Stochastic Gradient Descent Classification (SGD), alongside deep learning methods for feature extraction (VGGFace2 model pretraining and SENet-50 model architecture) across six different facial datasets. The results demonstrated that the highest accuracy (F-measure, Recall, and Precision) was achieved following the training and testing process in all datasets using the MLP and SGD classifiers.

Keywords: Keyword 1, keyword 2, number of keywords is usually 3-7, but more is allowed if deemed necessary

1. INTRODUCTION

A person can be verified or recognized using identification technology from a digital image or video. Images acquisition, faces representation, faces detection and matching make up a typical identification pipeline. identification technology is widely used in surveillance, border control, National security, law, e-commerce, banking and mobile phone user identification[1]. Humans are born with certain biological traits. Their exceptional traits and one-of-a-kindness, which are impossible to replicate. The biometric recognition-based personal ID system is attracting the greatest attention and has started to permeate many facets of our lives. In today's technologically advanced environment, image processing has many applications, including surveillance, teleconferencing, population & behaviour analysis, and identification via person recognition utilizing various biometrics[2].

Face recognition uses a variety of technologies including Neural Networks(NN), Image Processing(IP), and Machine Learning(ML). Face recognition is a key test used in the analysis of the multidimensional visual model and is a hot area

*Corresponding author: farah.jawad@qu.edu.iq

of examination. Since the human face displays shifting qualities including emotions, illumination, posture, disguise, and changes in hairstyle, among others, perceiving the human face is a challenging yet crucial talent to learn. The human face also experiences irreversible changes as we become older. These factors make face recognition non-trivial and challenging. Giving a computer system the ability to swiftly and reliably detect and recognize human faces in videos or photos is known as face recognition[3].

The primary problem with the face recognition system's feature representation strategy is that it extracts features for a specific biometric characteristic using a subpar method of representation. One of the important steps in image classification is features extraction. Numerous features extraction methods have been proposed such as Principal Component Analysis(PCA), Independent Component Analysis(ICA), and Local Binary Patterns(LBP). Convolution neural networks(CNN), in particular, have emerged as the most popular feature extraction technique in recent years[4]. There are numerous ways to use CNN. The first step is to fully learn the model Where using the pre-trained model. The second approach uses transfer-learning with CNN features that was previously trained. In the end, using CNNs through learning of transfer by preserving the initial basis and using the outputs to feed the classifier. When the problem is similar to one to be classified or the database is small, the pre_trained model is used[5].

In this research, we combined the power of deep learning with the flexibility of traditional machine learning algorithms by utilizing a pre-trained model for feature extraction and employing these features to train traditional models. This methodology introduces significant contributions and innovations, Using the VGGFace2 pre-trained model, which is highly capable of extracting rich representations from images due to its training on large-scale datasets. This approach eliminates the need for training a new deep learning model, saving time and computational resources while avoiding the risk of overfitting when working with limited datasets. Integrating deep Features with Traditional Machine Learning Algorithms, Instead of relying solely on a deep learning model, we used the extracted features as input to traditional machine learning models, such as Decision Tree, MLP, Bagging, Boosting, and SGD. This integration provides a simple yet effective solution for data analysis, leveraging the flexibility and interpretability of traditional models. Addressing Data and Resource Constraints. This approach tackles the challenge of limited data by utilizing high-quality, general-purpose features extracted from the pre-trained model, which enhances the performance of traditional models even with small datasets. Additionally, it reduces the need for high-end computing devices, as traditional machine learning algorithms are less computationally demanding. Performance, Results showed that models like MLP and SGD achieved high performance in all datasets when paired with the extracted features, demonstrating the effectiveness of this approach in achieving accurate and reliable image classification. The primary innovation lies in combining deep learning and traditional machine learning to strike a balance between high performance and computational efficiency. The research demonstrates how leveraging features extracted from pre-trained models can simplify the classification process and enhance performance while reducing complexity. This approach provides a practical solution to the challenges faced by researchers working with complex datasets or limited resources, making it a valuable contribution to the field of image classification using machine learning.

The structure of this article explains as follow: the previous works are described in Section 2. In Section 3, the datasets, approach, and system architecture are explained. The features extraction approach and classification methods are described in Sections 4 and 5. Results and comparisons of the approach are described in Section 6. Finally, conclusion in Section 7.

2. LITERATURE SURVEY

Artificial intelligence has made great strides recently in resolving a number of image processing problems for face recognition applications. Numerous studies had been conducted in the field. For example, Guo et al. [6] demonstrated a facial recognition system that extracts feature from images using CNN and uses SVM as a classifier. Using optimization techniques, they enhanced CNN's performance. The model enhances its rate of recognition while requiring less training time. The experiments in the study were based on the (ORL and FERET) datasets. According to the study's findings, the system had a high recognition rate and needed minimal training.

Abdalah and Alkazaz [6], in this work, random movie images were utilized to identify actors and determine their ages using machine-learning techniques. There are 574 photos from different movies in the Arab Actors Dataset-AAD, which includes both color and black-and-white images. The images depict complete scenes or fragments thereof, Multiple models were employed for feature extraction (like Inception v3, SqueezeNet, VGG-16, and VGG-19), and diverse machine learning algorithms were utilized during the classification actor based on face data (SVM, NB, Constant, SGD, KNN, DT, AdaBoost, CN2 Rule Induction, Gradient Boosting, Logistic Regression, RF, and ANN) determine the most effective algorithm for handling such image types. The study's AUC, accuracy, CA, and f1-score values of 99%, 86%, 85.5%, and 84.2%, respectively, showed that the Logistic Regression model was successful and performed the best during the training phase when compared to other models.

Akter *et al.*[7], To more accurately identify children with ASD in their early phases, this work proposes an enhanced transfer-learning-based autism face recognition system. As a result, we gathered facial images of kids with ASD from the Kaggle data base. When compared to the other classifiers and pre-trained models, we found that our enhanced MobileNet-V1 model displays the best accuracy of 90.67%. Additionally, this classifier uses the k-means clustering technique to identify distinct ASD groups based solely on autistic image data. For k = 2 autism subtypes, the enhanced MobileNet-V1 model therefore demonstrated the maximum accuracy (92.10%).

Aghdam, et al. [8] , authors investigated the elements that could help low resolution face recognition perform more accurately in inconsistent settings. They found that, when compared to pre-built models (models e and g), that is trained by using the MS-Celeb-1M dataset. While models f and h fine-tuned trained by using the VGGFace2 dataset considerably enhance Rank-1 IR for very low resolution probing face images (d1 of SCFace). This can be attributed to the fact that 20% of the face photos in VGGFace2 have a resolution of less than 50 pixels, which supported the model to acquire robust features for faces with low resolution. According to the study's results, cropped faces with more details and probing and gallery sets with similar resolutions perform much better in terms of Rank-1 IR. On 130 subjects from the SCFace benchmark, our model h obtained state-of-the-art Rank-1 IR results, saved scores of 75.08%, 97.69%, and 99.69% for d1, d2, and d3, respectively. With the model c, obtained scores 84.22% Rank-1 IR on the ICBRW benchmark, also it greatly enhanced the Rank-1 IR.

Swal, Vivek et al. [9], focused on employing a single camera for two approaches of masked face detection and recognition: First stage: a one-step YOLOv3-face model that has been pre-trained by using a set of known individuals; second stage: a two-step process using the pre-trained one-stage feature detector network RetinaFace for finding masked-faces and VGGFace2 that generates face image features vectors for effective mask-facial verification. The dataset utilized that involving seven videos of actual people in various lighting, occlusion, and orientation situations. RetinaFace and VGGFace2 can be executed one-to-one mask facial verification on a bespoke dataset by state-of-the-art results of (92%) whole performance and (94%) face verification accuracy, according to the results of the trial.

Prasad et al. [10], authors employed two deep learning models which are separate widely used: Lightened CNN VGG- Face for extraction of face representation under various situations such as upper and lower face occlusions, misalignment, various head positions, shifting illuminations, and inaccurate localization of facial features. As a result, the models demonstrated the deep learning is robust over various kinds of misalignment and can be withstand intraocular distance localization errors.

Mahmoud Ali et al. [11] studied facial recognition using deep learning techniques like Inception and Squeeze_Net, Random Forest, Logistic Regression, KNN, and SVM. They used the Pins-Face dataset. The Logistic Regression using the Inception model achieved the highest accuracy of 94.8%. This improved face recognition accuracy has spaC rked interest in various fields.

Usgan, M. et.al [12], Authors used a photo ID as a data set to identify a person. The primary for electronic identity cards and the tertiary IvS datasets were utilized to evaluate the suggested method. Age progression criteria for face recognition systems are provided in photo ID datasets. We only use one data—a photo ID that can currently only be used to recognize faces—for the training process. In order to refine the data during training, we apply the pretrained VGGFace2 model and (AM-Soft max loss) as a loss function. SVM is then used to perform the classification. Our findings have a higher accuracy score of 0.93%.

Al Daoud, Essam et.al [13] authors presented a novel face recognition technique based on combining "Gabor" and "VGGFace2" characteristics. The suggested methodology made up three stages: feature extraction, concatenation, and recognition. The VGGFace2 model implemented for the feature extraction, and PCA to minimize the features of Gabor. Besides, full connected neural networks are utilized to implement the recognition step. On the IJB-A dataset and the Labeled Faces in the Wild (LFW) dataset, the suggested technique was assessed. Results indicate that it can recognize objects with an accuracy of more than 99%.

Hemmatiyan-Larki, et.al [14] authors provided a useful system for face expression identification, this study evaluated various feature extraction and classification techniques. And it looked at a number of feature extraction techniques, including face encoding, features extracted and the Oriented Gradients Histogram by VGG16 Network. These features are used to evaluate a variety of traditional classifiers for classification, such Logistic Regression, SVM, and AdaBoost. Additionally, they modified a model ResNet50 that was pre-trained on the "VGGFace2" dataset. Finally, by concentrating on various sections of face photos, and also suggested a part-based ensemble classifier. The finding demonstrated that the tuned ResNet50 model with a complete image of face outperforms the alternative approaches based on FER-2013 dataset.

Keshava, M. Chenna et.al [15], presented an innovative approach that is used to find missing people. A common webpage allows users to contribute information and pictures of sketchy characters. The supplied image is promptly compared to the registered missing person images. The input person image is categorized after which the missing person database is queried for the image that best matches the input image. To complete the selection process, the deep learning type is trained to correctly identify the missing person image. A powerful deep learning technique and the convolutional NN are used to recognize faces. Face descriptors are generated using VGG-face deep architecture from images. The retrieved descriptors were classified by using the SVM and KNN classifiers and match nearest missing person testing photographs. Unlike other deep learning techniques, the convolutional algorithm is used as a high-level feature extractor. However, the results are independent of light, noise, contrast, occultation, the age subject, and position. Therefore, the results are quite precisely obtained.

Adithama, et.al [16], authors employed the VGGFace2 model, transfer learning and SENet 50 model for conducting facial recognition. The dataset was gathered via single-sample per-individual sampling, often known as one-shot learning.

Patient registration and verification are the two unique results of applying to the identification system. The patient verification focused on a minimum distance of (0.28) and matches only the relevant face, whereas registration utilized a minimum distance of (0.35) and matches data with the entire database. As a results for patients registered, the accuracy ranged from (90 % to 100%). However, the accuracy was 100 percent at the time of patient verification.

Singh et al. [17], authors introduced a real-time system to detect and identify indiviuals in live or recorded surveillance feeds using face recognition algorithms like CNN and deep learning. The database of real-time suggested integrated system is depended on the VGGFace deep learning neural architecture. The expreimental results demonstrated that the suggested approach achieved a highest degree of recognition accuracy by accurately recognizing for each 26 indiviuals with a confidence llevel ranging from (78.54 to 100)%, and the mean average of 96 percent on the real-time inputs.

Butuner, Resul et.al [18], researchers used machine learning algorithms to categorize lentil photos. SqueezeNet, InceptionV3, DeepLoc, and VGG16 were used to extract features from the captured images. In order to create models for the classification of lentil photos, Artificial Neural Network (ANN), Naive Bayes (NB), Random Forest (RF), Adaptive Boosting (AB), and Decision Tree (DT) algorithms were used. The success of classifying the developed machine learning models was calculated, and the outcomes were examined. The ANN algorithm had the maximum classification success with the deep features derived from the SqueezeNet model, 99.80%.

Chang *et al.* [19], the asymmetric face recognition problem is the primary topic of this research. In this study, an asymmetric face recognition mechanism—abbreviated AFRM—is proposed. The suggested AFRM first uses the support vector machine (SVM) and histogram of oriented gradients (HOG) to identify and extract every face from images. The convolution feature map (Conv_FF) is then used by AFRM to extract the features from each face, and the features are then adopted to divide the faces into various classes. The AFRM uses the K-nearest neighbors (KNN) to represent each face's features in order to rapidly identify a face. The proposed AFRM can use KNN to derive features after extracting one face's features during the new meeting. The suggested approach yields 97% accuracy without one-to-one name and face labeling.

Grd, Tomičić and Barčić [20], using a pre-trained EfficientNetV2S neural network and the potential of transfer learning, this research proposes a multi-step approach for face shape classification. Pre-processing, augmentation, training, and testing are important stages in our methodology that guarantee a thorough and dependable outcome. Our approach makes use of a publicly accessible dataset of female celebrities that includes five different face shape classes: square, round, oval, oblong, and heart. Using pre-trained weights, the EfficientNetV2S neural network maximizes accuracy, training time, and parameter size. As a result, the model's overall accuracy of 96.32% is efficient.

3. PROPOSED SYSTEM

The proposed system combines feature extraction to face identification and classification by leveraging the power of Machine Learning (ML) and Deep Learning (DL). Fig.1 illustrates the proposed system along with a detailed description of the work and methods.

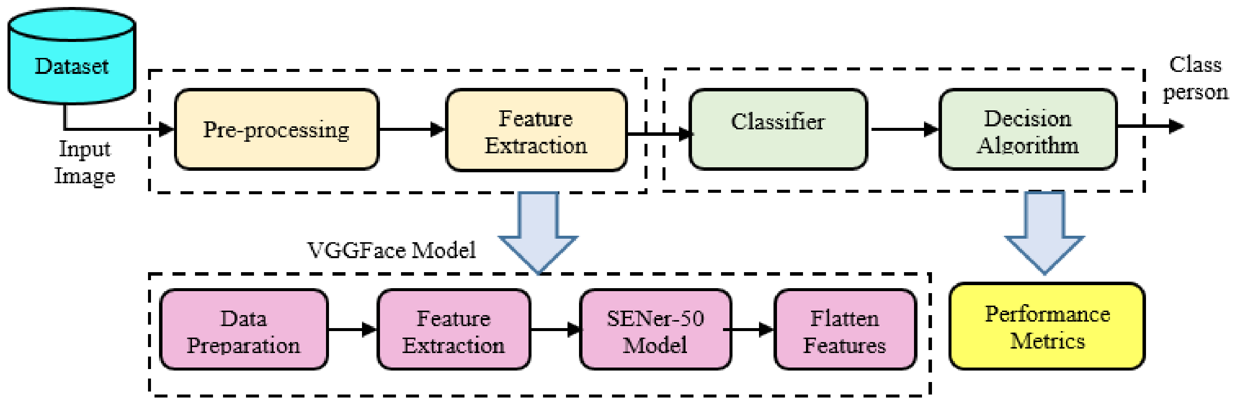

FIGURE 1. Proposed system

3.1 DATA PREPARATION

The first stage starts by getting the images and then resizing them to (row: 224, column: 224). All datasets are organized into subfolders, each representing a different person (class).

3.2 FEATURE EXTRACTION

In the second stage, the pre-trained VGGFace2 with SENet-50 weights was utilized in this study to produce face features from the original face images and then employed in the classification. The extracted features, which are in the form of high-dimensional arrays, are flattened into one-dimensional vectors. The extracted features (feature vectors) were stored along with their corresponding labels (classes). This transformation makes the data suitable for the classification step.

3.3 FEATURE FLATTING

The extracted features, which are in the form of high-dimensional arrays, are flattened into one-dimensional vectors. This transformation makes the data suitable for further processing and classification.

3.4 DATA SPLITTING

The dataset is divided into two parts: a training set and a testing set (70% of the data is allocated for training, while the remaining 30% is used for testing).

3.5 CLASSIFICATION

Traditional machine learning methods, such as Decision Tree, MLP, Bagging, Boosting, and SGD, were used to classify the extracted features. The features used to distinguish between numerous classes were used to train these machine learning models. Performance was assessed, showing that deep learning-derived features can be used by Traditional classifiers to provide reliable multi-class classification.

3.6 EVALUATION

After training the classifiers, then generate predictions for the faces in the test set. To assess the model's performance, it calculates various classification metrics, such as accuracy, precision, recall, and F1-score, for each famous class. These metrics provide insights into how well the model can correctly classify faces.

This proposed system blends the simplicity of use and adaptability of traditional machine learning techniques with the power of pre-trained deep learning models for feature extraction. It provides a strong framework for classifying facial images while striking a compromise between computational economy, interpretability, and performance.

4. DATASETS

In this study, six types of face databases were used to cover the largest number of famous faces (actors, singers, players, politicians, actresses, athletes, and social figures) from different countries and nationalities. The details for each database are shown as follows:

4.1 OPEN FAMOUS PEOPLE FACES

The authors of this dataset worked very hard to gather as many photos as they could for each class in the dataset. Eye position alignment was used to align the faces, and then landmarks were used to crop out the region of interest. There are 258 classes in the Open Famous People Faces dataset, and each class has at least five images. Images come in a variety of sizes; some are small and of low quality, while others are large and of great quality. The same individual appears in pictures taken at various ages [21].

FIGURE 2. Illustrates an example images of open famous dataset

4.2 MINI_LFW

This dataset (a cleaner version of the LFW dataset) is created by carefully clipping the area of interest from the images and then scaling them to 128x128 size. Additionally, several subjects in the original LFW dataset only have one image, so it doesn't add enough to the process. Therefore, we only chose topics that have seven or more photographs. By doing this, we optimized the data, and the number of subjects decreased to 256 from 5071 and photos to 5095 from 13000. This dataset can be deemed optimum relative to the original one due to less background noise in the photos and greater focus on the region of interest, i.e., faces[22].

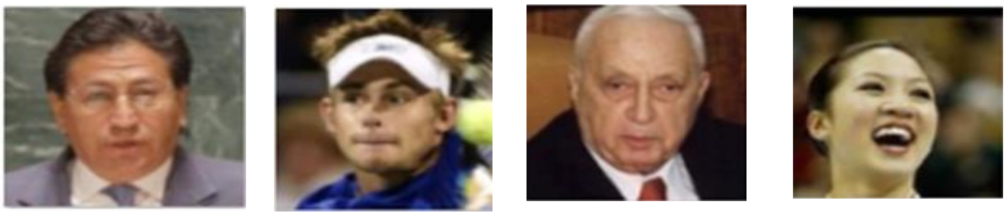

FIGURE 3. Illustrates an example Images of Mini_LFW dataset

4.3 BOLLYWOOD CELEBRITY FACES LOCALIZED DATASET (170)

The original dataset (Bollywood celeb localized face dataset) is extended by this dataset and now includes some more actors and actresses. The bing_image_downloader Python library is used to download the new Images from the Bing search engine. The face has been extracted from the images using the Open-CV DNN model. Each sample includes uncontrolled circumstances like various orientations, illuminations, age transitions, etc. This dataset contains 170 classes; each class has an average of 70 samples, and there are 12,257 total images in the collection [23].

FIGURE 4. illustrates an example Images of Bollywood dataset

4.4 FOOTBALL PLAYER'S FACES DATASET

This dataset was particularly gathered for an artificial intelligence effort to build a facial recognition algorithm. A tiny dataset for testing machine learning methods is provided here. Each of the eight football players has over 300 pictures in the collection (8 classes) [24].

FIGURE 5. illustrates an example Images of Football dataset

4.5 CELEB FACIAL RECOGNITION

This dataset was modified from Pins Face Recognition. This dataset contains 100 classes; each class has 108 samples, and there is an average of 10,800 total images in the collection [25].

FIGURE 6. illustrates an example Images of Celeb Facial dataset

4.6 NEPALI CELEB LOCALIZED FACE DATASET

The collection includes images of 50 well-known people from Nepal (50 classes); each class has (21-47) samples, including actors, singers, and social heavyweights. The participants in the dataset are diverse in terms of age, gender, and occupation. To capture each person's unique facial traits, photos were shot from several angles and perspectives, including front-facing, side-facing, and profile views. The dataset may have errors or inconsistencies, nevertheless, due to things like changes in lighting or image quality. Regarding possible applications for this dataset, they include face recognition, face detection, and emotion recognition. It could also be used to evaluate and improve machine learning models for applications involving facial analysis [26].

FIGURE 7. illustrates an example Images of Nepali dataset

5. FEATURE EXTRACTION

The amount of resources needed to describe enormous amounts of data is being reduced using feature extraction. Furthermore, feature extraction from particular data was a key problem for efficient machine-learning applications. There are several uses for feature extraction, especially in computer vision, which utilizes algorithms to distinguish between the many components of a picture. When an algorithm's input data is so large that it is difficult to handle it simply and it is projected to produce a surplus of data, the data is then translated into a simplified representation of the original data[27]. A face must first be analyzed to extract its features, which are merely numerical descriptions of the face. Then, the features are extracted; this procedure is known as feature extraction. In this study, the pre-trained VGGFace2 with SENet-50 model weights feature extractor was utilized to produce face features from the original face images, and it was then employed in the classification stage.

5.1 PRE-TRAINED CNN MODELS

The ability of the Convolutional neural network to recognize faces is significantly impacted by face representation, which has gained significance in current face recognition research. In the majority of published works, CNNs are used to classify images in four different ways: by creating network weights from scratch (typically only when a huge dataset is available), by adjusting the weights of a pre-trained model (which produces results similar to training from scratch and may be done in smaller datasets), by using unsupervised pre-training to create initial weights before training the model CNN, and by using a pre-trained model CNN (at least in part). In order to train a classifier that is more accurate and effective, the latter method often combines handcrafted traits with features taken from the CNN [11].

The two most popular models for facial recognition are VGGFace and VGGFace2. The term "VGGFace" designates a collection of face recognition algorithms created by Oxford University's Visual Geometry Group (VGG) and evaluated on benchmark datasets for computer vision. Omkar Parkhi described the VGGFace model in the 2015 article; it was later given a name. In order to train modern CNN-based facial recognition systems and beat the gigantic datasets used by Google and Facebook to train their models, the research contributed by describing the procedure for creating the required massive training dataset. This dataset is then used to construct deep CNNs for facial recognition systems like face identification and verification. They first go over the steps for training a face classifier, which uses a soft-max activation in the output layer to identify faces. The network output is thus a face embedding a vector feature representation after removing this layer. The model is then fine-tuned in order to narrow the gap between the vectors generated for the same identity and widen the Euclidean distance between the vectors generated for various identities. The network's architecture employs a deep CNN in the VGG fashion, with fully connected layers at the classifier end of the network and blocks of (convolutional layers) with small kernels and ReLU activations[28].

The VGG's Qiong, et al. present a follow-up study and introduce a new, substantial face data set dubbed VGGFace2 in their 2017 paper. There are 9311 subjects represented by (3.31) million photos or an average of 362.6 images per subject. Images are pulled from Google Search and vary greatly in terms of how they are staged, lit, colored, ethnicities represented, and used (e.g., of athletes, actors, and politicians). The main focuses of the paper are the collection, curation, and image preparation processes of this dataset. However, VGGFace2 is the pre-trained model created for facial recognition using this dataset[29]. In order to train a group of models using CNN using THE architecture of (ResNet-50)[30]. without and with (Squeeze and Excitation) (SE) blocks[31], the SqueezeNet-ResNet-50 model (called SE-ResNet-50 or SENet), a huge database called VGGFace2 was used. The authors have made these models and their modifications, as well as the related code, public. The models' performance is assessed using common facial recognition datasets, showing state-of-the-art results. This study of facial feature generation used models trained with the feature extractor (VGGFace2 with SENet model weights).

6. CLASSIFICATION

The classifier has been fed the extracted features. Six classifications are applied to compare the classifier's performance (Multilayer Perceptron Classifier (MLP), Decision Tree Classifier (DT), Bootstrap Aggregation Decision Tree Classifier (TreeBagger), Extreme Gradient Boosting Classifier (XGB), and Stochastic Gradient Descent Classification (SGD)). The machine learning classifiers are briefly explained.

6.1 MULTILAYER PERCEPTRON ALGORITHM (MLP)

MLP is a class of feedforward artificial neural networks (ANN) that was developed. The term multilayer perceptron is used to describe multilayer neural networks. Input, hidden, and output are the main three layers of a neural network. Through the input layer, feature vectors are delivered as the inputs. Weights are modified in the hidden layer for maximum performance, and the result is Back-propagation method, which belongs to the class of supervised learning, is used for training in Multi-Layer Perceptron. Here, the output is compared to the desired or actual output, and a cost function is developed that denotes prediction mistakes. Keeping the cost function as low as possible is our aim. The input and weights pointing toward the output will be provided to concealed units. The output is then carried out through the output layer, as was previously mentioned. Figure 2 illustrates the neural network design [31]. The input and weights pointing toward the output will be provided to concealed units. The output is then carried out through the output layer, as was previously mentioned. Figure 8 illustrates the neural network design, [32] compared to the target value, and if we are not getting the intended result, the cost function value is back-propagated, and the weights are altered accordingly.

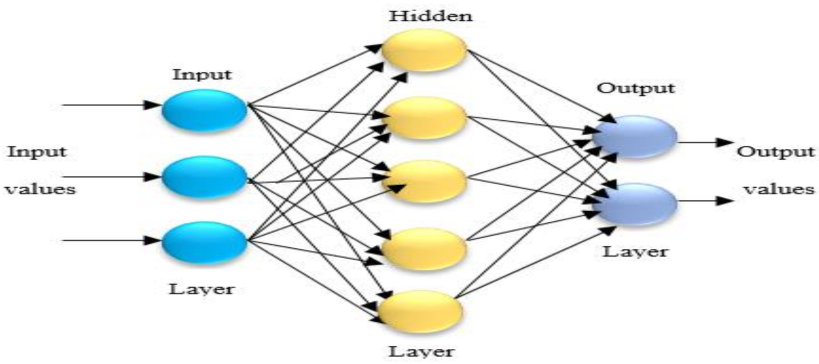

FIGURE 8. MLP Structure[32]

The weights network learns knowledge of the various classes through the updating process. Utilizing various techniques, such as the boot force approach and gradient descent approach, the cost function is minimized in order to alter the weights and provide correct results. Thus, training through weight adjustment is repeated numerous times until we achieve the lowest error rate, or training is repeated till the maximum number of iterations[33].

6.2 DECISION TREE ALGORITHM (DT)

DT can be used in a variety of situations and is surprisingly effective for such a simple approach. It takes minimum pre-processing of the data and can even use empty values. DT focuses on learning straightforward decision-rules inferred from the data, which are then put together as a collection of (if_then_else) decision-rules. Entropy and information gain are the foundation of the classification approach known as DT Eq.(1) and Eq.(2) [34].

$$E(T) = \sum_{i=1}^{m} - p(q_i)\log(p(q_i)) \qquad (1)$$

$$I = E(T) - \sum_{v \in E} p(v)E(v) \qquad (2)$$

Where (q) is a binary label (0 - 1), T is current data, (v) is a sub-set data, and $p(q)$ is the proportion of q label.

6.3 EXTREM GRADIENT BOOSTING CLASSIFIER (XGBCLASSIFIER)

The authors in [35] proposed "tXGBClassifier" and is an improvement to "gradient boosting classifier" . In comparison to other machine learning algorithms nowadays, XGBClassifier produces effective results and addresses the over-fitting issue. For classification, this model requires extremely little time and space.The XGBClassifier can be summed up as follows [32].

a. Regularized Learning: A tree ensemble model called XGBClassifier employs both classification and regression trees (CART). The model is expressed mathematically as follows:

$$\hat{y}_i = \sum_{S=1}^{S} f_S(x_i), \qquad f_S \in F \qquad (3)$$

This function operates on a set of all possible classification and regression trees, where S is the trees number. To further optimize the learning process utilizing Eq. (4), a training target is developed:

$$£(\Phi)=\sum_{i} l\,(\hat{y}_i, y_i)+\sum_{k} \Omega\,(f_k) \qquad (4)$$

Similarly, since the (£) parameters are represented by Φ, (l) measures the disparity between (y_i) and ((y_i) ̂). A regularization technique is used to measure the model's complexity. Mathematically, it is expressed in Eq.(5) as:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda||w||^2 \qquad (5)$$

b. Gradient Tree Boosting: To train the data, it applied the additive approach so £(Φ) function is modified in Eq.(6):

$$£^t = \sum_{i=1}^{n} l\,(\hat{y}_i^{t-1}, y_i + f_t(x_i)) + \sum_{i=1}^{t} \Omega\,(f_i\,) \qquad (6)$$

To minimize the goal, (f_t) is added and (t) denotes the number of iterations. The target at step t is approximated as follows in Eq.(7) after removing the fixed limits:

$$\tilde{£}^t = \sum_{n=1}^{N} g_i\, f_i(x_i) + \frac{1}{2} h_i f_i^2(x_i) + \Omega(f_t\,) \qquad (7)$$

$$g_i = \partial_{\hat{y}_i^{t-1}} l(\hat{y}_i^t, y_i)$$

$$h_i = \partial^2_{\hat{y}_i^{t-1}} l(\hat{y}_i^t, y_i)$$

Where (gi and hi) are the first- and second-order gradient statistics loss functions.

c. Tree Structure: Gain is then created to evaluate the model's effectiveness using Eqs. (5) and (7).

$$Gain = \frac{1}{2}\left[\frac{Q_L^2}{F_L+\lambda} + \frac{Q_R^2}{F_R+\lambda} - \frac{(Q_L+Q)^2}{F_L+F_R+\lambda}\right] - \gamma \qquad (8)$$

d. Parameter Tuning: The XGBClassifier chooses only a few parameters for multi-classification. The total number of classes utilized for classification was Num_class. The random_state value is 10. Five is the configured maximum tree depth.

6.4 STOCHASTIC GRADIENT DESCENT (SGD) CLASSIFIER

Both machine learning and deep learning applications favor gradient-based optimization techniques for problems involving large-scale of data. Stochastic approaches are preferred because of how well they scale. In contrast to gradient descent, which computes the entire gradient ∇f(wk), Stochastic gradient descent randomly selects one of the gradients ∇fn(wk) and uses that to update wk. Even if this might go from the absolute minimum, it still functions as expected[37].

$$E[\nabla fn(wk)] = \nabla f(wk) \qquad (9)$$

The fact that ∇fn(wk) is a key feature of SGD. The algorithm strength changes (wk), even at the best, where the genuine gradient is 0.

SGD's rate of convergence is noticeably slower than gradient descent's. SGD uses linear convergence to minimize the residual error, while quadratic convergence results in an even quicker asymptotic convergence. However, the value of N in each SGD iteration is independent[37] .

6.5 BOOTSTRAP AGGREGATION DECISION TREE CLASSIFIER (TREEBAGGER)

A robust statistical technique for estimating a quantity from a dataset is bootstrapping. From the original dataset, it generates numerous sub-sample datasets and computes their means. The final amount of the provided dataset is then estimated using the mean value of all the mean sub-samples. Bagging, also known as bootstrap aggregate, is an extremely effective ensemble approach. An ensemble is a method that combines predictions from various machine learning algorithms to make predictions that are more accurate than predictions from individual models. The typical method to lower the algorithm's high variances is the bootstrap algorithm. The trained data has a significant impact on the decision tree. The tree and the forecast value will vary if the data changes. The bagging classifier, Tree Bagger technique, is employed in this experiment to forecast the model. Simply said, bagging is a paradigm in which a large number of "weak" learners are simultaneously educated to address the same issue and are then combined to get superior outcomes [38].

7. RESULTS AND DISCUSSION

This section explains the results of applying the transfer learning VGGFace2 model with SENet model weights for extracting features and (Multilayer Perceptron Classifier (MLP), Decision Tree Classifier (DT), Bootstrap Aggregation

Decision Tree Classifier (TreeBagger), Extreme Gradient Boosting Classifier (XGB), and Stochastic Gradient Descent Classification (SGD)) machine learning algorithms for classifying these features. The six data sets were split into 70% for training and 30% for testing. As illustrated in Figure 3, the results of this experimentation displayed the highest accuracy after the training and testing process in all datasets in the MLP and SGD classifiers.

Our system is implemented in Python 3.11, trained using a CPU of Intel Core i7-7700HQ with RAM (16GB). The Lenovo computer also has a Nvidia GEFORCE GTX with 6 GB of memory operated to accelerate the neural networks' calculations on 64-bit Windows 10 OS. To evaluate performance, we used a variety of measures (precision, recall, F-measure, and accuracy). The F-measure (F-Score), which balances precision and recall, is used to assess a test's accuracy. The more accurate performance measurements (precision and recall) may be achieved by using this metric. On a more serious note, the results are greatest when the F-measure is higher. Less false positives and false negatives result from increased accuracy. The metrics listed below [39]:

$$\mathrm{Pr}ecision = \frac{TP}{TP + FP}$$

$$\mathrm{Re}call = \frac{TP}{TP + FN}$$

$$F - measure = 2^* = \frac{\mathrm{Pr}ecison^* \mathrm{Re}call}{\mathrm{Pr}ecison + \mathrm{Re}call}$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

Tables (1, 2, 3, 4, 5, and 6) illustrate the experimental results for MLP, DT, TreeBagger, XGB, and SGD with transfer learning (VGGFace2).

Table 1. The results of implementing the MLP, DT, TreeBagger, XGB, and SGD with VGGFace2In Open Famous People Faces

Methods	Precision %	Recall %	F-measure %	Accuracy %
MLP	0.99	0.99	0.99	0.99
DT	0.64	0.57	0.57	0.57
TreeBagger	0.93	0.91	0.91	0.91
XGB	0.97	0.96	0.96	0.96
SGD	0.99	0.98	0.99	0.99

Table 2. The results of implementing the MLP, DT, TreeBagger, XGB, and SGD with VGGFace2 in Bollywood Celebrity Faces

Methods	Precision %	Recall %	F-measure %	Accuracy %
MLP	0.96	0.96	0.95	0.96
DT	0.64	0.62	0.62	0.62
TreeBagger	0.91	0.90	0.90	0.90
XGB	0.90	0.91	0.90	0.90
SGD	0.98	0.97	0.98	0.98

Table 3. The results of implementing the MLP, DT, TreeBagger, XGB, and SGD with VGGFace2 in Celeb Facial Recognition

Methods	Precision %	Recall %	F-measure %	Accuracy %
MLP	0.98	0.98	0.98	0.98
DT	0.72	0.71	0.71	0.72
TreeBagger	0.94	0.94	0.93	0.94
XGB	0.94	0.93	0.93	0.94
SGD	0.99	0.99	0.99	0.99

Table 4. The results of implementing the MLP, DT, TreeBagger, XGB, and SGD with VGGFace2 in Mini_LFW

Methods	Precision %	Recall %	F-measure %	Accuracy %
MLP	0.96	0.94	0.94	0.94
DT	0.57	0.54	0.53	0.54
TreeBagger	0.82	0.81	0.80	0.81
XGB	0.88	0.87	0.86	0.87
SGD	0.97	0.95	0.96	0.95

Table 5. The results of implementing the MLP, DT, TreeBagger, XGB, and SGD with VGGFace2 in Football players faces

Methods	Precision %	Recall %	F-measure %	Accuracy %
MLP	0.98	0.98	0.98	0.98
DT	0.91	0.90	0.90	0.90
TreeBagger	0.97	0.96	0.97	0.97
XGB	0.98	0.98	0.98	0.98
SGD	0.98	0.97	0.98	0.98

Table 6. The results of implementing the MLP, DT, TreeBagger, XGB, and SGD with VGGFace2 in Nepali Celeb Localized Face

Methods	Precision %	Recall %	F-measure %	Accuracy %
MLP	0.94	0.98	0.93	0.93
DT	0.66	0.64	0.64	0.64
TreeBagger	0.89	0.86	0.86	0.86
XGB	0.93	0.92	0.92	0.92
SGD	0.95	0.94	0.94	0.94

The given (Figures 9 and 10) compare different classifier precision implementation metrics (MLP and SGD) using a plotted graph.

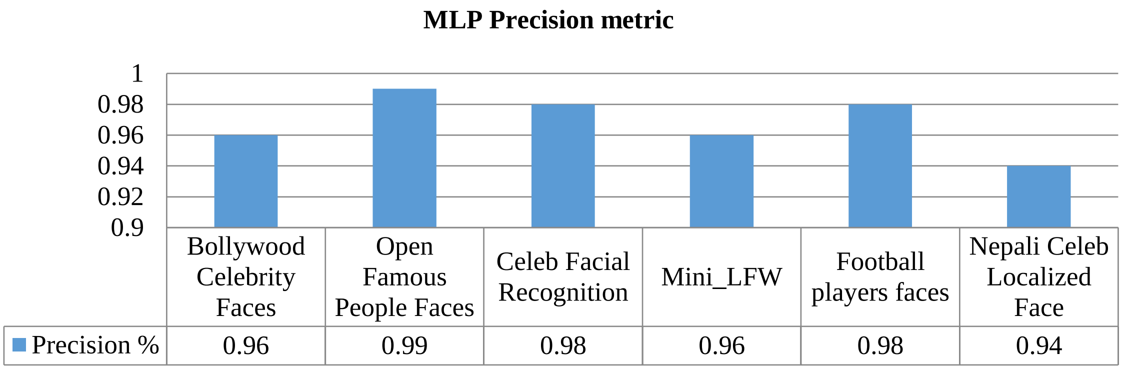

FIGURE 9. MLP classifier precision Comparative

To compare with the previous works of performance with the latest system models in terms of the datasets. All facial datasets are from the "Kaggle website" and are newly published, and there are no previous published studies that used this data until the write of this research paper.

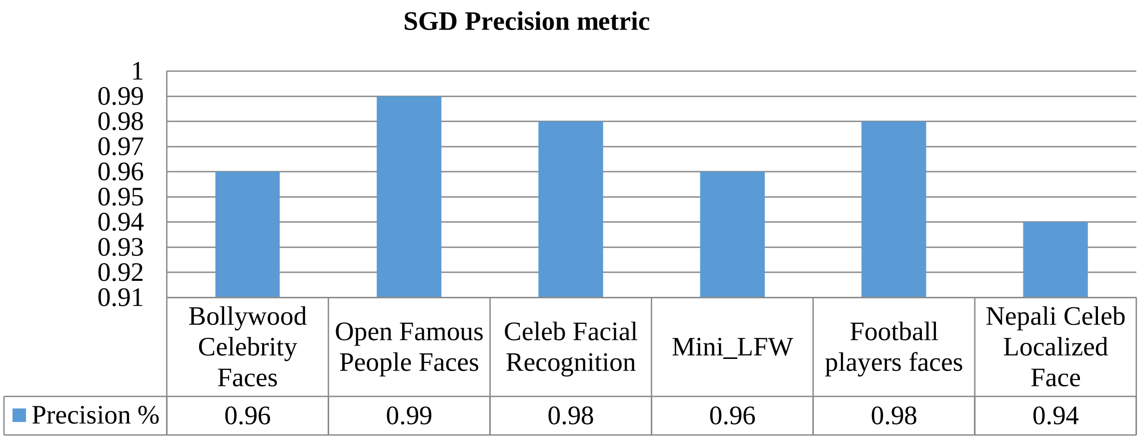

FIGURE 10. SGD classifier precision Comparative

The given (Figures 11,12,13,14,15 and 16) shows the micro-average ROC for MLP and SGD classifiers. By analyzing the relationship between the True Positive Rate (TPR) and the False Positive Rate (FPR), the Receiver Operating Characteristic (ROC) curve is used to evaluate how well classification models work. It is best suited for binary classification tasks because it demonstrates an algorithm's ability to discriminate between two classes. To solve multi-class classification issues utilizing the Micro-Average ROC Curve, the "One-vs-All" method splits the task into several binary classification problems. Because every instance is treated equally, regardless of class, the Micro-Average ROC curve aggregates the contributions of all classes. This approach gives an overall view of how well the algorithm is performing.

In this research, the micro-average ROC was utilized. Since there are many classes in the datasets used in this study, the typical ROC curve is less useful because of its complexity and the several curves produced for each class. We made the performance analysis for every class simpler by using the Micro-Average ROC curve. This method provides a more thorough and understandable assessment of the model's overall performance by combining true positives and false positives from all classes.

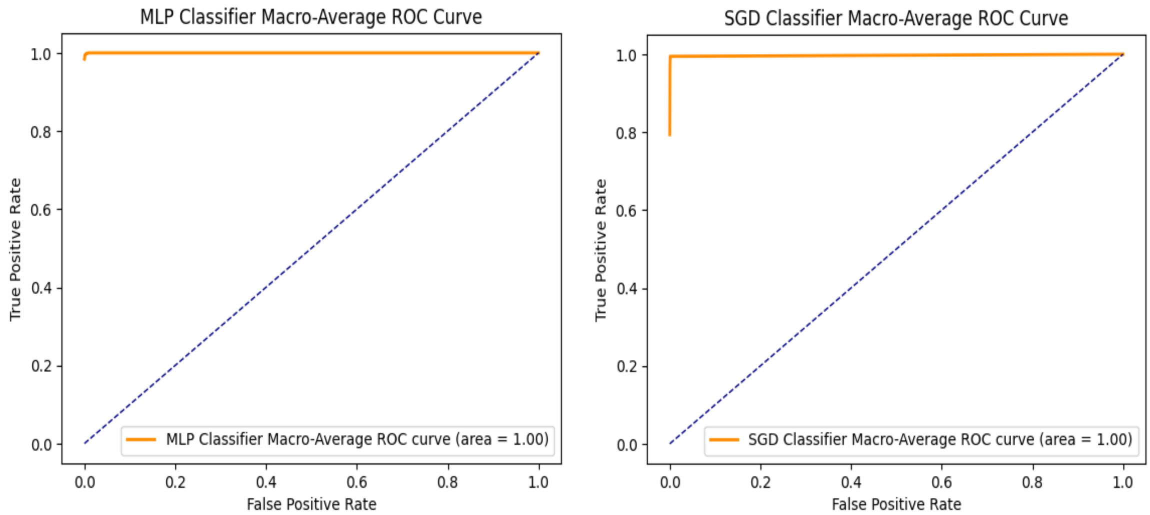

FIGURE 11. **MLP and SGD Classifiers Macro-Average ROC Curve in Celeb Facial Recognition dataset**

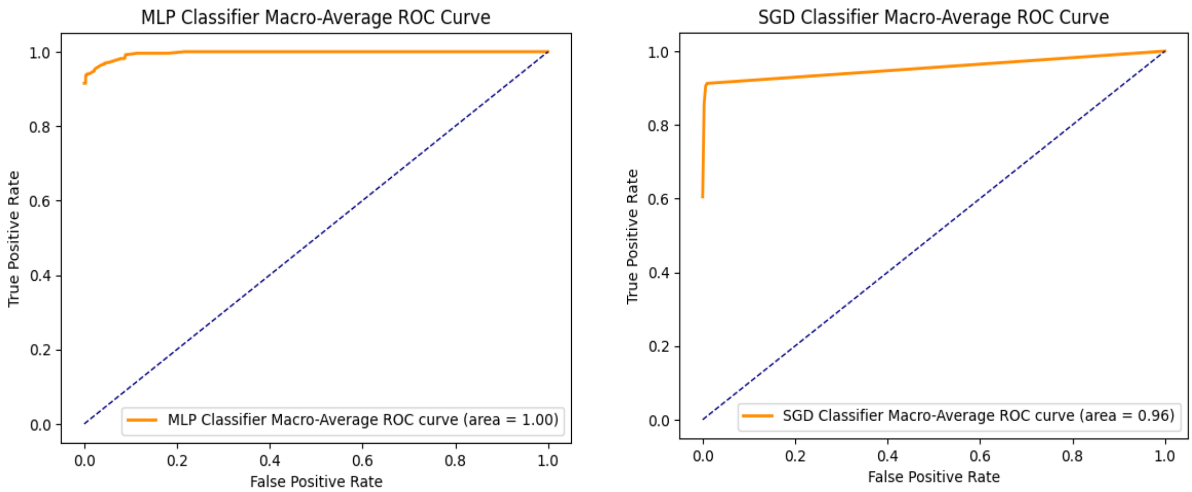

FIGURE 12. **MLP and SGD Classifiers Macro-Average ROC Curve in Nepali Celeb Localized Face dataset**

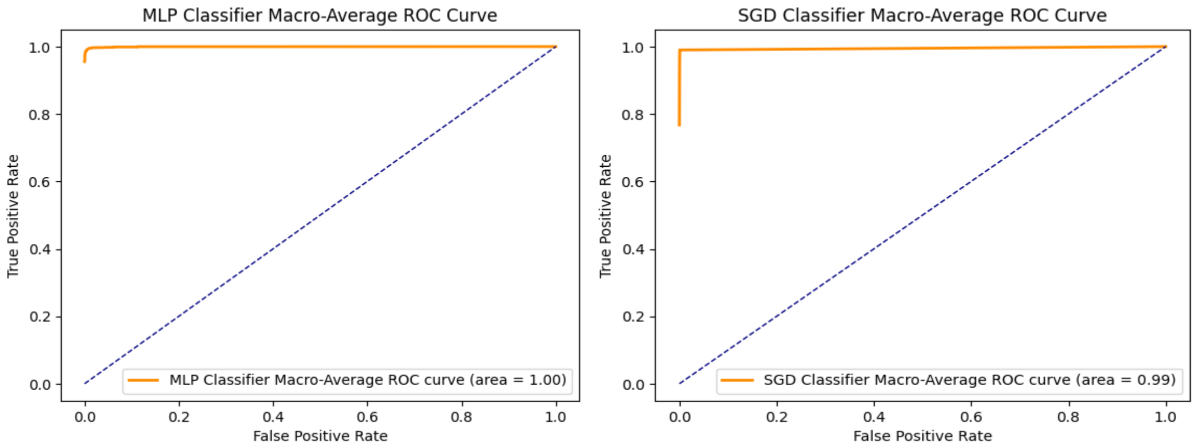

FIGURE 13. **MLP and SGD Classifiers Macro-Average ROC Curve in Bollywood Celebrity Faces dataset**

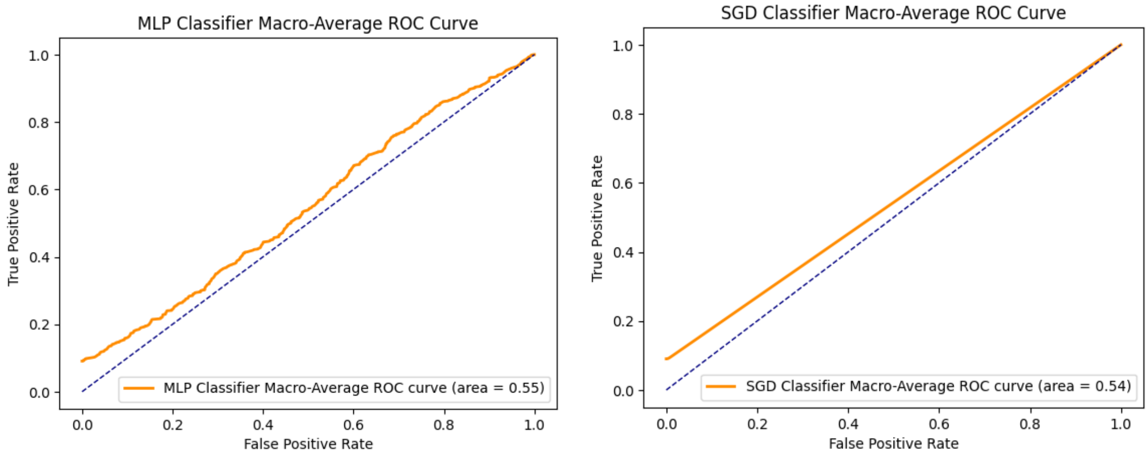

FIGURE 14. **MLP and SGD Classifiers Macro-Average ROC Curve in Open Famous People Faces dataset**

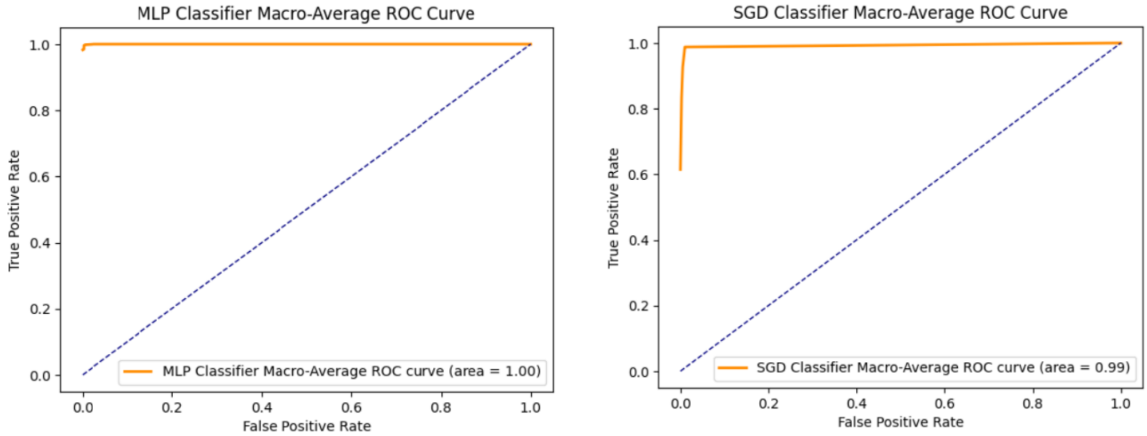

FIGURE 15. **MLP and SGD Classifiers Macro-Average ROC Curve in Football Players Faces dataset**

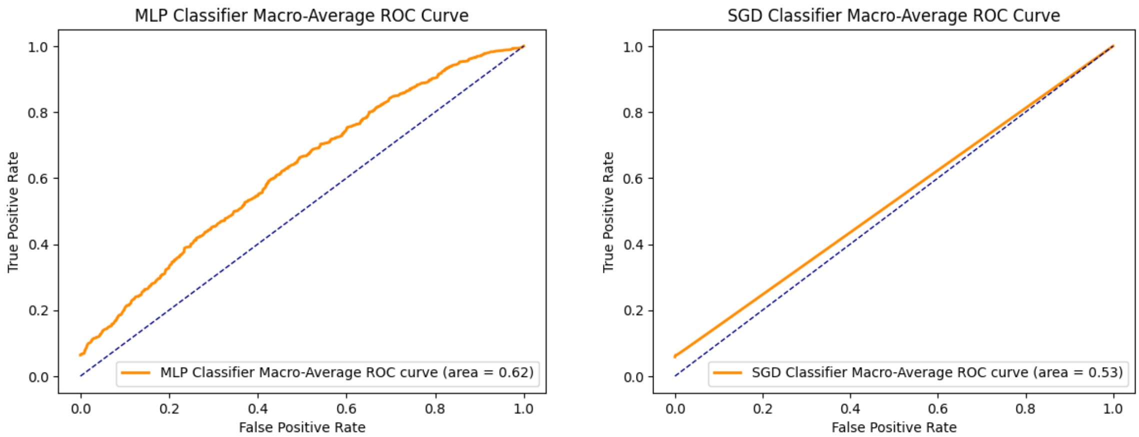

FIGURE 16. MLP and SGD Classifiers Macro-Average ROC Curve in Mini_LFW Dataset

The reason behind the low values in ROC Curves in (Mini_LFW and Open Famous People Faces) datasets, despite the high results in other metrics, can be attributed to several factors. Nature of ROC Curves in Multi-Class Classification, Multi-class classification is transformed into multiple binary classifications using the "One-vs-All" approach. This often results in lower apparent performance when averaging across all classes. Characteristics of the Dataset, The diversity in image quality (high and low resolution) makes it harder to distinguish between classes. Variations within the same class (e.g., images of the same person at different ages) lead to overlapping distributions among classes. finally, Imbalanced numbers of images across classes negatively impact the results.

Metrics such as accuracy and F1-Score prioritize correct classification, while ROC Curves measure the model's ability to distinguish between classes based on probabilities. This makes ROC Curves less reflective of overall performance in some cases. Improving ROC Curves can be achieved by enhancing image quality, addressing class imbalance, and fine-tuning the model's probability predictions.

8. CONCLUSION

In this study, we combined the most well-known machine learning classifiers for facial photo Identification with deep learning models for feature extraction. The six Face datasets was used in this studies that published on Kaggle. Images in this dataset cover the largest number of famous faces (actors, singers, players, politicians, actresses, athletes, and social figures) from different countries and nationalities. and cover large deviations in phrases of the pose, background chaos, illumination, low quality, age, expressions, and varied individuals, and are supported by a big number of face images. Also, it has useful range of accessories like as hats, eyeglasses, sun-glasses, and etc. In this paper, the feature extraction step, we applied transfer learning VGGFace2 with SENet-50 model weights, and then used the well-known classifiers on this features that results to achieve the best result for each one and compared them. The result showed that MLP and SGD classifiers achieved higher accuracy and provided the best results on precision. Further studies may include other machine learning classifiers or deep learning models, and any of these might be combined to produce a more complex framework. They ought to be more accurate at recognizing

CONFLICTS OF INTEREST

There is no conflict of interest. We would like to certify that there are no known conflicts of interest linked with the publication.

REFERENCES

[1] A. Almadan, A. Krishnan, and A. Rattani, “BWCFace: Open-set Face Recognition using Body-worn Camera,” *Proc. - 19th IEEE Int. Conf. Mach. Learn. Appl. ICMLA 2020*, pp. 1036–1043, 2020, doi: 10.1109/ICMLA51294.2020.00168.

[2] S. Hapani, N. Prabhu, N. Parakhiya, and M. Paghdal, “Automated Attendance System Using Image Processing,” *Proc. - 2018 4th Int. Conf. Comput. Commun. Control Autom. ICCUBEA 2018*, pp. 1–4, 2018, doi: 10.1109/ICCUBEA.2018.8697824.

[3] U. Jayaraman, P. Gupta, S. Gupta, G. Arora, and K. Tiwari, “Recent development in face recognition,” *Neurocomputing*, vol. 408, pp. 231–245, 2020, doi: 10.1016/j.neucom.2019.08.110.

[4] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, “Deepface: Closing the Gap to human-level performance in

face verification," *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, pp. 1701–1708, 2014, doi: 10.1109/CVPR.2014.220.
[5] C. Boufenar, A. Kerboua, and M. Batouche, "Investigation on deep learning for off-line handwritten Arabic character recognition," *Cogn. Syst. Res.*, vol. 50, no. August, pp. 180–195, 2018, doi: 10.1063/5.0068646.
[6] A. M. Abdalah and N. R. Alkazaz, "Predicting Movie Production Years through Facial Recognition of Actors with Machine Learning," *Baghdad Sci. J.*, vol. 21, pp. 4146–4162, 2024, doi: 10.21123/bsj.2024.8996.
[7] T. Akter *et al.*, "Improved transfer-learning-based facial recognition framework to detect autistic children at an early stage," *Brain Sci.*, vol. 11, no. 6, 2021, doi: 10.3390/brainsci11060734.
[8] O. A. Aghdam, B. Bozorgtabar, H. K. Ekenel, and J. P. Thiran, "Exploring factors for improving low resolution face recognition," *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, vol. 2019-June, pp. 2363–2370, 2019, doi: 10.1109/CVPRW.2019.00290.
[9] V. Aswal, O. Tupe, S. Shaikh, and N. N. Charniya, "Single Camera Masked Face Identification," *Proc. - 19th IEEE Int. Conf. Mach. Learn. Appl. ICMLA 2020*, pp. 57–60, 2020, doi: 10.1109/ICMLA51294.2020.00018.
[10] P. S. Prasad, R. Pathak, V. K. Gunjan, and H. V. Ramana Rao, "Deep Learning Based Representation for Face Recognition," *Lect. Notes Electr. Eng.*, vol. 570, no. January, pp. 419–424, 2020, doi: 10.1007/978-981-13-8715-9_50.
[11] M. Ali and D. Kumar, "A Combination between Deep learning for feature extraction and Machine Learning for Recognition," *2021 12th Int. Conf. Comput. Commun. Netw. Technol. ICCCNT 2021*, pp. 1–4, 2021, doi: 10.1109/ICCCNT51525.2021.9579913.
[12] M. Usgan, R. Ferdiana, and I. Ardiyanto, "Deep learning pre-trained model as feature extraction in facial recognition for identification of electronic identity cards by considering age progressing," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1115, no. 1, p. 012009, 2021, doi: 10.1088/1757-899x/1115/1/012009.
[13] E. Al Daoud and G. Samara, "Improving the Face Recognition Performance Using Gabor and VGGFace2 Features Concatenation," *6th Int. Conf. Inf. Technol. InCIT 2022*, pp. 187–190, 2022, doi: 10.1109/InCIT56086.2022.10067669.
[14] A. M. Hemmatiyan-Larki, F. Rafiee-Karkevandi, and M. Yazdian-Dehkordi, "Facial Expression Recognition: A Comparison with Different Classical and Deep Learning Methods," *Iran. Conf. Mach. Vis. Image Process. MVIP*, vol. 2022-Febru, pp. 1–4, 2022, doi: 10.1109/MVIP53647.2022.9738553.
[15] M. C. Keshava, C. S. Prasanna, V. Kavitha, K. Mahima, V. Padmavathi, and P. Srinivasulu, "Missing Person Identification System Using Deep Learning Algorithm (CNN) and Machine Learning Classifiers," *Lect. Notes Electr. Eng.*, vol. 915, pp. 267–276, 2022, doi: 10.1007/978-981-19-2828-4_26.
[16] S. P. Adithama, M. Maslim, and R. Gunawan, "Implementation of Face Recognition for Patient Identification Using the Transfer Learning Method," *TEM J.*, vol. 12, no. 2, pp. 775–784, 2023, doi: 10.18421/TEM122-22.
[17] A. Singh, S. Bhatt, V. Nayak, and M. Shah, "Automation of surveillance systems using deep learning and facial recognition," *Int. J. Syst. Assur. Eng. Manag.*, vol. 14, no. s1, pp. 236–245, 2023, doi: 10.1007/s13198-022-01844-6.
[18] R. Butuner, I. Cinar, Y. S. Taspinar, R. Kursun, M. H. Calp, and M. Koklu, "Classification of deep image features of lentil varieties with machine learning techniques," *Eur. Food Res. Technol.*, vol. 249, no. 5, pp. 1303–1316, 2023, doi: 10.1007/s00217-023-04214-z.
[19] C. Y. Chang, A. S. Santra, I. H. Chang, S. J. Wu, D. S. Roy, and Q. Zhang, "Design and implementation of a real-time face recognition system based on artificial intelligence techniques," *Multimed. Syst.*, vol. 30, no. 2, pp. 1–9, 2024, doi: 10.1007/s00530-024-01306-y.
[20] P. Grd, I. Tomičić, and E. Barčić, "Transfer Learning with EfficientNetV2S for Automatic Face Shape Classification," *J. Univers. Comput. Sci.*, vol. 30, no. 2, pp. 153–178, 2024, doi: 10.3897/jucs.104490.
[21] Y. ROMERO and A. Rocha, "Open Famous People Faces," *Kaggle*, 2022. https://www.kaggle.com/datasets/yveslr/open-famous-people-faces
[22] AMSHRA267, "Mini LFW dataset," *Kaggle*, 2021. https://www.kaggle.com/datasets/amanmishra4yearbtech/cleaned-lfw?resource=download
[23] S. Roy, "Bollywood Celebrity Faces explore Localized Dataset (170)," 2023. https://www.kaggle.com/datasets/sroy93/bollywood-celeb-localized-face-dataset-extended
[24] A. BALI, Mohamed Aleya, and A. Riahi, "Football players faces dataset," 2021. https://www.kaggle.com/datasets/azizbali/football-players-faces-dataset
[25] S. Satpute, "Celeb Facial Recognition," 2022. https://www.kaggle.com/datasets/soujanyasatpute/celeb-facial-recognition
[26] Amitpant7, "Nepali Celeb Localized Face Dataset," 2023. https://www.kaggle.com/datasets/soujanyasatpute/celeb-facial-recognition
[27] M. O. Oloyede, G. P. Hancke, and H. C. Myburgh, "A review on face recognition systems: recent approaches and challenges," *Multimed. Tools Appl.*, vol. 79, no. 37–38, pp. 27891–27922, 2020, doi: 10.1007/s11042-020-09261-2.
[28] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep Face Recognition," in *British Machine Vision Association*,

2015, pp. 1–12. doi: 10.1201/9781003091356-4.
[29] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "VGGFace2: A dataset for recognising faces across pose and age," *Proc. - 13th IEEE Int. Conf. Autom. Face Gesture Recognition, FG 2018*, pp. 67–74, 2018, doi: 10.1109/FG.2018.00020.
[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778. doi: 10.14456/mijet.2021.8.
[31] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation_Networks," in *IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141. [Online]. Available: http://openaccess.thecvf.com/content_cvpr_2018/html/Hu_Squeeze-and-Excitation_Networks_CVPR_2018_paper.html
[32] G. Al-Naymat, M. Al-Kasassbeh, N. Abu-Samhadanh, and S. Sakr, "Classification of VoIP and non-VoIP traffic using machine learning approaches," *J. Theor. Appl. Inf. Technol.*, vol. 92, no. 2, pp. 403–414, 2016.
[33] R. C. Damale and B. V. Pathak, "Face Recognition Based Attendance System Using Machine Learning Algorithms," in *Proceedings of the Second International Conference on Intelligent Computing and Control Systems (ICICCS 2018)*, 2018, no. Iciccs, pp. 414–419.
[34] M. Loey, G. Manogaran, M. Hamed, and N. Taha, "A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic," *Measurement*, vol. 167, pp. 1–8, 2021.
[35] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, vol. 13-17-Augu, pp. 785–794, 2016, doi: 10.1145/2939672.2939785.
[36] Monika, M. Kumar, and M. Kumar, "XGBoost: 2D-object recognition using shape descriptors and extreme gradient boosting classifier," *Adv. Intell. Syst. Comput.*, vol. 1227, pp. 207–222, 2021, doi: 10.1007/978-981-15-6876-3_16.
[37] M. Ayoub, M. S. Khokha, J. Zakria, and W. Rasheed, "Efficient Face Re-Identification through PSO Based Adaptive Deep Learning Models," *J. Indep. Stud. Res. Comput.*, vol. 19, no. 2, pp. 18–35, 2021, doi: 10.31645/jisrc.37.19.2.4.
[38] R. Sukmawati and M. I. Tarmizi, "A Method for Identification of Face of the Person and Presence of Facial Spot in the Face," *Int. J. Res. Appl. Sci. Biotechnol.*, vol. 7, no. 5, pp. 42–49, 2020, doi: 10.20944/preprints202008.0330.v1.
[39] N. S. Mullah and W. M. N. W. Zainon, "Advances in Machine Learning Algorithms for Hate Speech Detection in Social Media: A Review," *IEEE Access*, vol. 9, pp. 88364–88376, 2021, doi: 10.1109/ACCESS.2021.3089515.