Comparative study of Machine Learning, Deep Learning and Bayesian Models on the ORL Faces Dataset for Effectual Face Recognition

Vivek Vignesh Baggam
dept. Computer Science & Engineering
Amrita Vishwa Vidyapeetham
Amaravati, India
baggamvivekvignesh@gmail.com

Divya K S
dept. Computer Science & Engineering
Amrita Vishwa Vidyapeetham
Amaravati, India
divyasekar4428@gmail.com

Ayush Pandey
dept. Computer Science & Engineering
Amrita Vishwa Vidyapeetham
Amaravati, India
ayushpandey1177@gmail.com

Rushika Manchala
dept. Computer Science & Engineering
Amrita Vishwa Vidyapeetham
Amaravati, India
manchalarushikagar@gmail.com

Dr.M.Srinivas
dept. Computer Science & Engineering
Amrita Vishwa Vidyapeetham
Amaravati, India
m_srinivas@av.amrita.edu

***Abstract*—This study compares face recognition performance between machine learning, deep learning, and probabilistic reasoning models. On the ORL dataset from Kaggle, the Machine Learning (ML) model (support vector machine (SVM)) achieved a recognition accuracy of 96.25.%, The Deep Learning (DL) model convolutional neural network (CNN) achieved a recognition accuracy of 99.99.% and a Probabilistic (PR) model Bayesian network attained an accuracy of 90.%. This analysis emphasizes CNN's superior accuracy in face recognition, especially on large-scale datasets, while also illuminating the comparatively limited effectiveness of Bayesian Networks with PCA and SVM for dimensionality reduction and classification.**

***Index Terms*—Face recognition, ML, DL, PR, Bayesian Network, Convolutional Neural Network (CNN), PCA, SVM, Soft Voting Classifier, Feature extraction.**

I. Introduction

Face recognition has witnessed a major evolution in modern security systems by redesigning the authentication of identities across the world [1]. The innovative way of capturing individuals through their facial identity is broadly used across the natural spectrum primarily due to its reliable, user-friendly attitude over traditional methods. While these conventional methods are prone to being compromised, biometric systems leverage unique physical traits, such as fingerprints and voice recognition, to provide enhanced security. Among these, face recognition has gained unparalleled attention due to its efficiency and accuracy.

Noticeably, the global market for the revolutionary technique has managed to capture and set its foothold emphatically with its annual rate estimated to be 14.9.% from 2023 to 2030 while its current value in 2022 stood at 5.15 billion US dollar [2]. With the ever-increasing signals of need, technology continues to offer efficient ways to monitor and access the authentication of individuals with minimal physical exertion.

Generally, the method acts as an umbrella for three categories in holistic and hybrid features. Features per name essentially aid in extracting facial landmarks such as the eye, nose, mouth, etc. On the contrary, holistic are utilized in capturing a way wider range of characteristics by covering the entire face as a united unit. As the name suggests, interjecting both techniques and their superiorities end up in superior recognition performance. Despite the underlings of the tech mainly fore-bearing majority of the components, unseen issues such as lighting, pose, and expressions massively hinder the accuracy. The ever-affluent ML and DNNs have paved the way for a steward rise to the face recognition system, equipping them with the ability to overcome various challenges and boost their adaptability. This has established a technology that is equally capable in multi-disciplinary fields such as law enforcement to access control. A FRS is no longer just a fascinating concept or ideal of deep learning [3], it has now found its way into the mainstream due to widespread capability and efficiency. With this field on an exponential rise, the foreword potential of the FSR system is on its way to secure imminent identification and personal security alongside maintaining robustness.

A more often approach to face recognition problem can be summed up as providing a set of images of the face labeled with the individual's identification and an unlabeled set containing pictures of the very exact group of individuals. The labeled data serves the purpose of training while the model makes predictions on the testing set which is the latter one. The primary objective is to identify every person in the test set.

1. We faced challenges in processing NPZ files for face

979-8-3315-2751-8/25/$31.00 ©2025 IEEE

recognition, particularly in data storage and preprocessing, which required efficient handling techniques.

2. We faced difficulties in feature extraction and dimensionality reduction, which were overcome by using CNNs with PCA.

3. We faced issues with robustness to variations in lighting, expressions, and poses, which were addressed by combining CNNs and BNs, enhancing the model's performance.

4. Validating the model on both the ORL faces posed its own set of challenges, but we demonstrated strong generalization and improved performance.

Literature Review

Feifei et al. [4] proposed an innovative approach for noisy face identification using Pulse-Coupled Neural Networks, PCA, and SVM. Tested on the ORL dataset, the method achieved superior accuracy under diverse noise conditions, outperforming traditional algorithms.

LathaS et al. [5] applied DWT, OLBP, and Euclidean Distance for facial recognition, testing on JAFFE and ORL datasets. The method outperformed DT-CWT, DT-CWT with HE, and DCT+DT-CWT, demonstrating the HE+DWT+OLBP technique as highly resilient and robust to outliers.

Jahan et al. [6] introduced the Spectral Discriminative Projection Pursuit (SDPP) and its enhanced version, Spectral Local Spatial-SDPP (SLS-SDPP), for dimensionality reduction in face recognition. Tested on ORL and Yale datasets, SLS-SDPP increased recognition rates from 66% to 92% on Yale and 79% to 96% on ORL, outperforming Eigenface and Fisherface.

Aniketk et al. [7] compared CNN, PCA, LDA, SVM, Random Forest, and KNN for facial recognition on ORL_faces and Georgia Tech datasets. CNN achieved 95% on ORL_faces and 85% on Georgia Tech, while HOG reached 96% on ORL_faces but dropped to 69% on Georgia Tech.

EimadA et al. [8] proposed a PCA-based image encryption method for face recognition, using cellular automata and gray code to secure facial images. Tested on the ORL dataset, it achieved 92.5% accuracy, with high NPCR and low correlation, proving robustness against attacks.

LiorS et al. [9] analyzed facial recognition on Yale and ORL datasets, showing non-facial features (e.g., background) can inflate accuracy and bias results. For example, Yale B achieved 99%, ORL reached 79%, and FERET performed at 13.5%, emphasizing the need for careful evaluation.

S. Sakthivel et al. [10] evaluated dimensionality reduction techniques on the ORL dataset, including PCA, Kernel PCA, LDA, LPP, and NPE with SVM. Kernel PCA achieved 100% accuracy (avg. 91.75%), while PCA and LDA reached 83.25%, and LPP and NPE lagged at 84.5% and 81.75%, respectively.

ChengjunL et al. [11] proposed the Independent Gabor Features (IGF) method, combining Gabor wavelet with ICA for face recognition. Tested on FERET and ORL datasets, it achieved 98.5% on FERET and 100% on ORL with 180 and 88 features, outperforming Eigenfaces and Fisherfaces.

Enrique.G et al. [12] introduced the LASRC algorithm, combining least-squares speed with SRC robustness for face classification. Tested on 800,000 Facebook faces, it matched SRC in performance with a 100-250x speedup, outperforming SVMs in efficiency.

Jialintang et al. [13] combined CNN with LBP for face recognition, achieving 100% on ORL and 97.51% on Yale-B, outperforming PCA and HOG-CNN. It also boosted pedestrian detection accuracy on CBCL (96.3% to 99.5%) and INRIA (89.2% to 98.1%).

T. Syed Akeel et al. [14] proposed the SGD-LCDRC algorithm, combining gradient descent with momentum and style transfer for enhanced accuracy. Tested on ORL, Yale, and Extended Yale B datasets, it achieved 93.34%, 90.07%, and 98.83% recognition rates, outperforming LDRC by 1.5% to 10.5%.

Stan Z. Li et al. [15] explored the NFL method, using linear interpolation and extrapolation between prototype feature points to enhance face recognition. It achieved a 3.125% error rate, outperforming the previous lowest (3.83%) by CNNs, demonstrating improved representational capacity.

M.Tamilselvi et.al [16] presented the Hybrid Robust Point Similarity based Convolutional Neural Network (HRPSM_CNN), designed for face recognition in uncontrolled environments. The researchers aimed at addressing limitations/persisting hindrances in existing algorithms (like Local Binary Patterns and Multi-SVM) under conditions such as poor lighting and posture. It was tested on ORL, AR, and LFW datasets and it achieved a staggering 97% accuracy, outperforming traditional methods in precision and stability.

Petya D. et al. [17] evaluated a combined SVD-HMM model, training unique HMMs for each subject using SVD features. Tested on ORL and Yale datasets, it achieved 96.6% on ORL and 82.7% on Yale, showing robustness to dimensionality reduction compared to standard SVD-based recognition.

Bai Ling et al. [18] proposed a face recognition method combining wavelet sub-band representation and a memory-associated kernel, achieving low computational costs. Tested on FERET, XM2VTS, and ORL datasets, it reached 91.6% on FERET and over 98% on ORL, outperforming Eigenface and ARENA.

II. Materials and Methods

A. Dataset

In this study, we utilized a face recognition dataset from Kaggle and compared ML, DL and PR models for the detection and classification of different faces and to see how well our methodology works on for different models. The dataset we used was ORL_faces [19] which was obtained from the Kaggle repository and was used to develop a systematic classification method. The details of the dataset are provided below:

Dataset Name: ORL_faces-Kaggle
File Format: NPZ
Feature Type: Images

Entities: 40 subjects, 10 images per subject
Dataset Size: 400 images (92×112 pixels each)

Fig. 1. Sample of ORL_face images of different people

B. Computational and Software Requirements

To work on the above datasets for this study we utilized a laptop with an ideal core i7 with 16 GB of RAM, and Windows 11 with Python 3.11.9 and we have used NumPy, pandas, sklearn, pgmpy, and Tensorflow libraries. This hardware was selected for its high computational performance and compatibility with existing laboratory equipment.

C. Methodology

Different models from ML, DL, AND PR were applied to the ORL_faces dataset. machine learning algorithms like KNN, RF, and SVM were employed, the Deep Learning model CNN was employed and the Problestic reasoning model Bayesian network model was employed Later the dataset was loaded into the environment which was split into training and testing sets for SVM, KNN, RF, Bayesian network and training, testing, and validation sets for the CNN

Support vector machines (SVM) is a powerful machine learning technique used for regression and classification tasks. It is mostly suitable for image classification, spam detection, spam detection. It focuses on finding the maximum separating hyperplane between different classes in target variable, making it robust for multi-class classification. Maximum Margin in support vector machines maximizes the separation between each class's closest points and the hyperplane. The margin M is defined as:

$$M = \frac{2}{\|\mathbf{w}\|} \tag{1}$$

where w is the weight vector of the hyperplane.

Random Forest (RF) is a machine learning ensemble method used for classification and regression tasks. It constructs multiple decision trees during training and outputs the average prediction for regression tasks or the majority vote for classification tasks. In medical applications like disease diagnosis, Random Forest is effective in analyzing complex medical data and identifying patterns. It is robust to overfitting and can handle large, high-dimensional datasets, making it a valuable tool for medical research and diagnosis.

K-nearest neighbor (KNN) is a technique used to predict output classes for input data by considering the k-nearest neighbors from the training dataset. Euclidean distance is the most used distance matrix. Freshly entered data is assigned a class label based on the most neighbors among the k-nearest neighbors. However, missing values must be addressed before using KNN, as distance calculation is difficult and computation costs are high due to the separation between test cases and training samples. The Euclidean distance between an input sample x and a point x_i from the dataset is computed as:

$$d(x, x_i) = \sqrt{\sum_{j=1}^{n} (x_j - x_{ij})^2} \tag{2}$$

here, the number of features is indicated by n. The majority category of the k nearest neighbors determines the prediction.

A Soft Voting Based Machine learning algorithm is proposed. Three different Classifiers SVM, KNN, RANDOM FOREST stacked, and the model predicts the output class based on the highest probability. The voting process provides a concise summary of the methods used to compare various training models. There are two voting methods: soft voting and hard voting, each with its own advantages and disadvantages. This work proposes the use of a soft voting classifier. In machine learning, soft voting combines the predictions of several models to enhance the overall performance of the ensemble. Soft voting is applicable when the base classifiers are probability-based. Soft voting combines the predicted probabilities of each class from multiple classifiers, weighting their votes based on confidence levels, leading to a more advanced and often more accurate ensemble prediction.

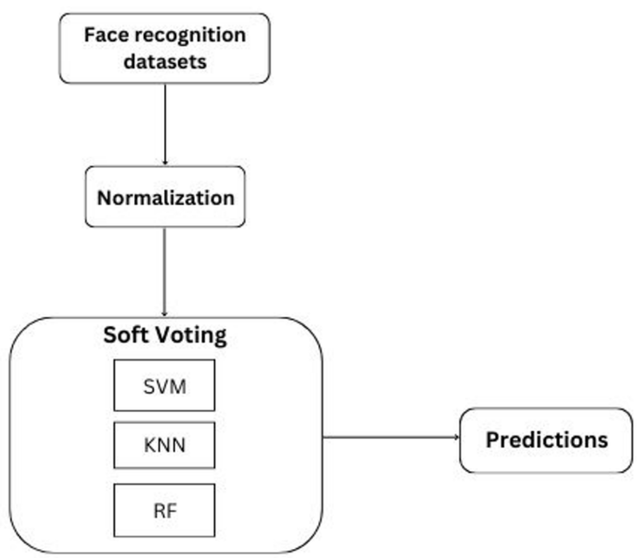

Fig. 2. Architecture of soft-voting

Principal Component Analysis (PCA) is applied to transform high-dimensional image data into a smaller set of uncorrelated principal components. In PCA, the variables are standardized [20]. The process initiates with the standardization of the variable followed by the calculation of the covariance matrix. This is further followed by mathematical calculations of eigenvectors and eigenvalues which serve as pre-requisite steps for extracting feature vectors. It is summed up through projection onto the principal component axes [7]. A binary dimension face image may be expressed as a vector of the form (X_i). Then we can obtain the face image training set X $(X = [x_1, x_2, x_3, \ldots, x_N])$. The obtained mean image is μ.

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$$

The mean training set is $\overline{X}$.

$$\{X\} = [\mu, \mu, \mu, \ldots, \mu]$$

Then, we define the covariance matrix of data X as S_t:

$$S_t = (X - \overline{X})(X - \overline{X})^T$$

If the rank of S_t is p, the eigenvalues of S_t are $\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_p$, the eigenvectors of S_t are $\omega_1, \omega_2, \omega_3, \ldots, \omega_p$. Then we can get the formula:

$$S_t\omega_i = \lambda_i\omega_i$$

The matrix $\lambda_i\omega_i$ constitutes the characteristics of the facial visual data. Because the vector $\omega_1, \omega_2, \omega_3, \ldots, \omega_p$ contains a set of orthogonal vectors, the face images are driven into the subspace defined by the feature image whereas the data is non-correlated in the sense of second-order statistics [4].

As Bayesian networks can handle categorical data better than continuous data, each principal component value is discretized into four bins (quartiles). This transforms the continuous data into discrete categories. This process is known as **Quartile based binning.**

Bayesian Network in simpler terms may be referred to as a graphical model that operates on probability constituting representation of a variable set and their conditional dependencies using a directed Acyclic graph (DAG). Every singular edge denotes conditional dependencies among those varying variables. Bayesian Networks are particularly useful in classification tasks where they can handle uncertainties and infer missing information based on known relationships between variables [21]. The Bayesian network is defined with a node for each PCA component and one label node representing the class label. Using the trained Bayesian network, inference is performed via Variable Elimination.

D. Algorithm Proposed

1) Load Data:
   - Load and normalize the ORL_faces images.
   - Split data into training, validation, and testing sets.
2) Reshape Data:
   - Reshape images to the format expected by CNN.
3) Apply PCA:
   - Flatten images and reduce dimensionality using PCA.
4) Discretize Features:
   - Transform continuous PCA features into quartile-based discrete categories.
5) Define Bayesian Network:
   - Construct the Bayesian network with nodes for each PCA component and a label node.
6) Train Network:
   - Train the Bayesian network on the discretized training data using Bayesian Estimation.
7) Perform Inference:
   - For each test instance:
     a) Transform image using PCA.
     b) Discretize transformed features.
8) Calculate Accuracy:
   - Compare predicted labels with actual test labels.
   - Compute accuracy metric.

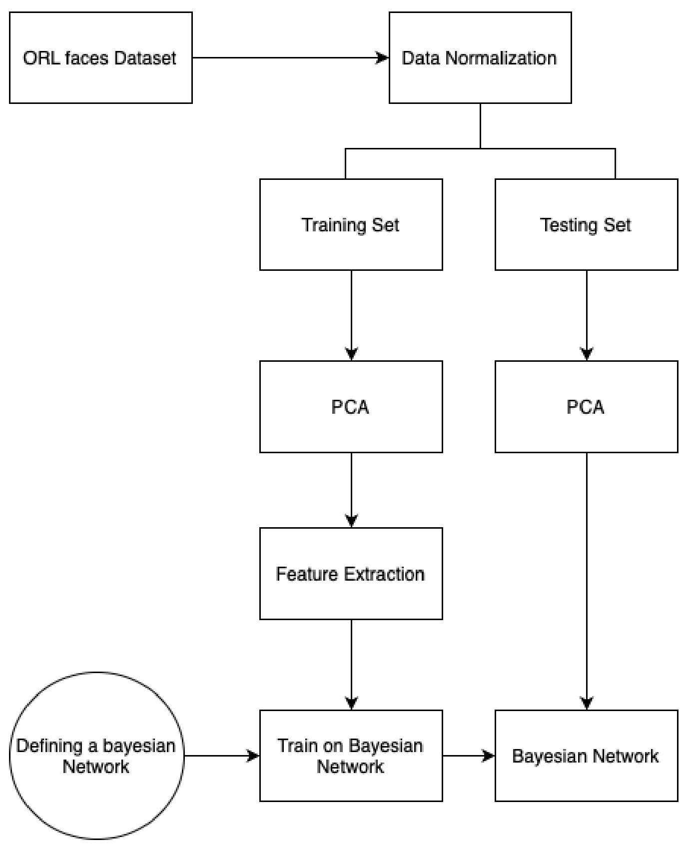

Fig. 3. Bayesian Architecture

CNN (Convolution Neural Networks):

A CNN operates on a similar complexity to of a human brain which consists of nerves and neurons aiding in the transfer of sensory data while convulsing in the extraction of useful information over the rest. Each CNN model contains primarily three layers separated diligently stacked on one other in a sequential order forming a collective model. The layer atop is the convolutional layer accountable for convoluting every available pixel in its receptive field onto a one-digit value. This step convolutes the input image rendering it available for extraction of features. Next in the stack is the activation layer which is tasked to apply the desired activation on each part of the neuron to determine in-step activation of neurons per the subjective conditions. The obtained output is then parsed through the pooling layer. A pooling layer minimizes the dimension of the resulting feature maps adjusting the image well enough for it to be passed to the final layer of the fully connected layer that transforms the entireity of the pooled feature map into a uni-dimensional column for processing [21].

CNN model is coupled with two convolutional layers, succeded over by a max pooling layer that extracts features from the dataset. The architecture initiates along a convolutional layer of 36 filters with a kernel size of 7 and ReLU activation to capture low-level features which is further passed on to a max pooling layer for reduction of spatial dimensions. The step is repeated with a second convolutional layer of 54 filters and a kernel size of 5, followed by another max pooling layer. The output is then flattened to prepare for fully connected layers.

After flattening, the model includes three dense layers with ReLU activation (with units 2024, 1024, and 512, respectively) and dropout layers to reduce overfitting. Finally, a softmax output layer with 20 units represents the 20 classes, providing class probabilities for each face image. The model is compiled by utilizing sparse categorical cross-entropy loss, an Adam optimizer valued with a learning rate of 0.0001, and accuracy as the metric. It was trained for 250 epochs on the training data, validated on a separate validation set, and then evaluated on the test set for final accuracy and loss measurements. We observed an accuracy of 99.99.% on the ORL_FACES dataset.

E. Algorithm Proposed

1) Load and Preprocess Data:
   - Load the `ORL_faces` dataset.
   - Normalize images by dividing pixel values by 255.
   - Split the dataset into training, validation, and testing sets.
   - Reshape images to the required format.
2) Define ML, DL, and PR Models:
   - Initialize a Sequential model with the following architecture:
     a) Two `Conv2D` layers with:
        – Filters: 36 and 54.
        – Kernel sizes: 7 and 5.
        – Each followed by a `MaxPooling2D` layer.
     b) Flatten the output.
     c) Add three `Dense` layers with:
        – Units: 2024, 1024, and 512.
        – Each followed by a `Dropout` layer with a rate of 0.5.
     d) End with a `Dense` layer with a `softmax` activation for 20 classes.
   - Divide the dataset into training and testing sets for model training and evaluation.
3) Compile and Train Model:
   - Compile the model using:
     – Loss: Sparse categorical cross-entropy.
     – Optimizer: Adam with a learning rate of 0.0001.
   - Train the model for 250 epochs on the training data with validation on the validation set.
4) Evaluate Model:
   - Evaluate the model on the test set.
   - Output the test loss and accuracy metrics.

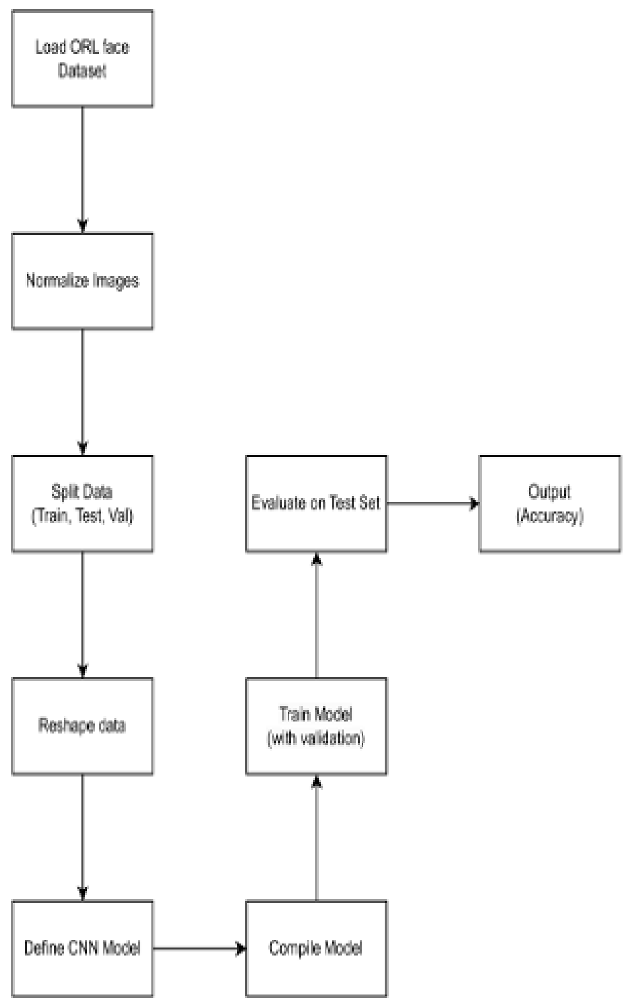

Fig. 4. Architecture of convolution neural network

F. RESULTS AND DISCUSSIONS

Confusion Matrix(Before Normalization) The Count on the diagonal indicates correct classifications, while off-diagonal values indicate misclassification. It reflects high specificity for most classes with minimal misclassification.

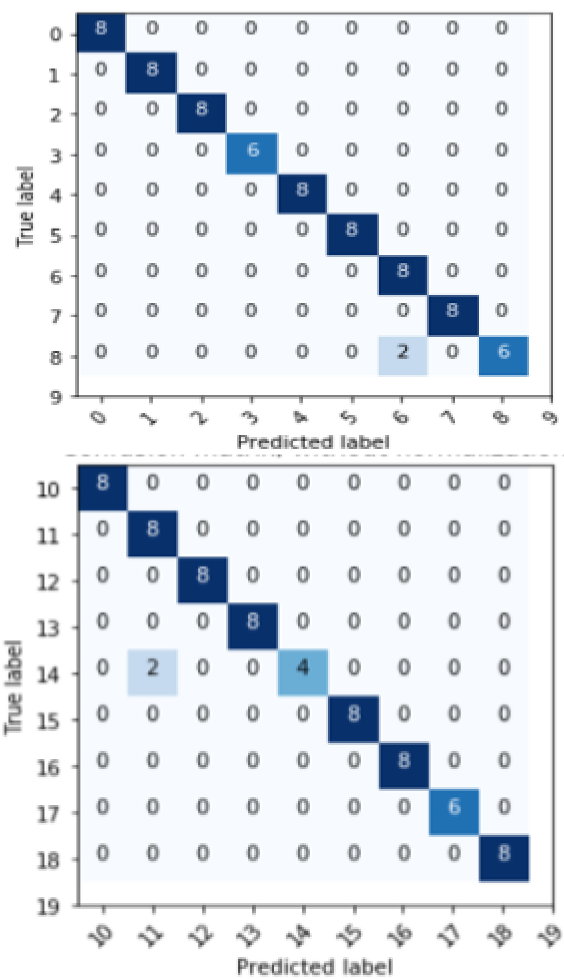

Fig. 5. confusion matrix without normalization

Training vs Testing Accuracy The Model's accuracy curve indicates a steady increase in accuracy reaching near-perfect accuracy (almost 1) after around 50 epochs. Also, the test accuracy follows the training accuracy, showing that the model generalizes well.

Training vs Testing Loss The loss curve indicates a sharp decline within 50 epochs that stabilizes at a low loss value for both training and testing. This convergence suggests that the model effectively minimizes error across both datasets without significant Overfitting or Underfitting.

The CNN model demonstrates strong classification ability on the ORL_Face dataset, achieving high accuracy with minimal loss compared to the Bayesian model with 80.% accuracy. These are the predictions if the face is recognized or not and the total number of prediction and if it's true or not.

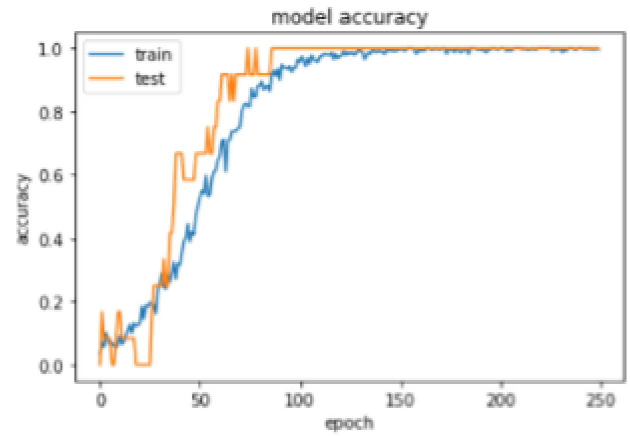

Fig. 6. Training and Testing Accuracy

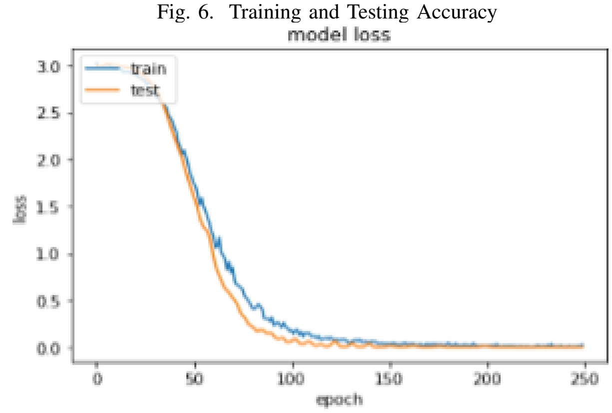

Fig. 7. Training vs testing Loss

In this paper, we have chosen two models one is a Deep learning model, and the other is a Probabilistic Graphical Model to accurately detect the faces: convolution neural network and Bayesian network using the face recognition dataset ORL_Faces from Kaggle we used feature extraction techniques like principal component analysis (PCA) we applied CNN and Bayesian network to the model.

TABLE I
ORL FACE DATASET

Classifier	Accuracy	Precision	F-measure
CNN	99.99	98.43	98.32
Bayesian network	90.23	88.64	88.26
Soft-Voting	95.25	94.24	94.78

applied on different datasets in which the CNN model was classified to give more accuracy when compared to the Bayesian network.

Comparison with relevant papers The above compares the

H

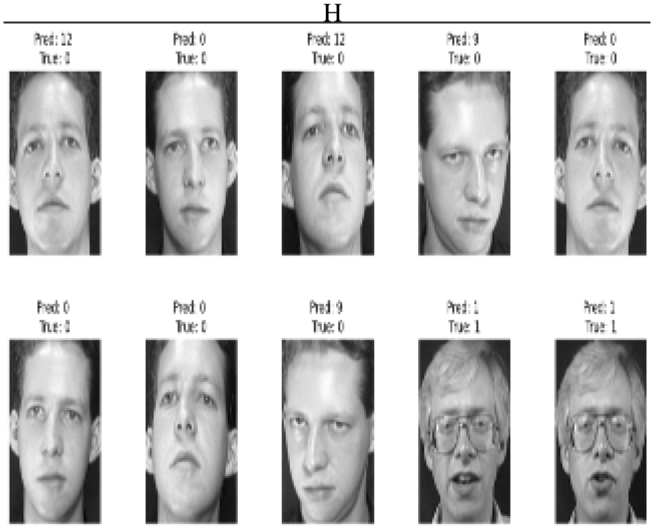

Fig. 8.

TABLE II
PROPOSED ALGORITHMS AND THEIR PERFORMANCE ON THE ORL FACE DATASET

Proposed Algorithm	**Dataset Used**	**Accuracy (%)**
DWT, OLBP, ED [4]	JAFFE, ORL_faces	90.3
SLS-SDPP [7]	ORL_faces, YALE	96.0
CNN, SVM, KNN [8]	ORL_faces	96.0
CNN	ORL_faces, LFW	99.99

performance of the proposed model with various state-of-the-art results. This suggests that the deep learning model CNN is more accurate in face recognition when compared to the Bayesian network and other Machine Learning models.

G. CONCLUSION

This study, methodically contrasts different ML, DL and PR models using face recognition dataset obtained from Kaggle. We maximized model readiness and assured dataset integrity through careful feature selection and preprocessing. PCA was applied on CNN, Bayesian Networks and various machine learning models like KNN, SVM and RF among which CNN gave the highest accuracy of 99.99.%. These reduce and highlight the difference between different classification model of ML, DL and PR in face recognition. This research handles a critical gap in already existing literature by conducting a comparison of a Deep learning model and a Probiotic Reasoning Model and applied to face recognition datasets from Kaggle, meticulously pre-processed data by addressing missing values, outliers, and performing feature selection to optimize model efficiency Rigorous PCA and LDA further validated these results, affirming consistent high-performance metrics including sensitivity, specificity, precision, and F1 score. Our study thus contributes valuable insights into the advanced methodologies for classifying facial images in various conditions and constraints. In the feature, our objective is to use various feature extraction techniques, and other models and develop a face recognition model using machine learning.

REFERENCES

[1] L. Li, X. Mu, S. Li, and H. Peng, "A review of face recognition technology," *IEEE Access*, vol. 8, pp. 139 110–139 120, 2020. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9145558/

[2] P. Dauvergne, "The globalization of facial recognition technology," in *Identified, Tracked, and Profiled*. Edward Elgar Publishing, 2022, pp. 76–87. [Online]. Available: https://www.researchgate.net/profile/Dauvergne-Peter/publication/366280625_Dauvergne_FRT_Ch_9_2022/links/639a6b58e42faa7e75c34317/Dauvergne-FRT-Ch-9-2022.pdf

[3] C. Dalvi, M. Rathod, S. Patil, S. Gite, and K. Kotecha, "A survey of ai-based facial emotion recognition: Features, ml & dl techniques, age-wise datasets and future directions," *IEEE Access*, vol. 9, pp. 165 806–165 840, 2021. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9631205/

[4] S. Feifei, H. Min, N. Rencan, and W. Zhangyong, "Noisy faces recognition based on pcnn and pca," in *2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI)*. IEEE, 2017, pp. 300–304. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8265797/?casa_token=53p1NtEiYXAAAAAA:z_rMkNBsgnpZX2a2KBnGSLaq2CkoyOXXWBNmqa_ktQQYBEwD__uJBW4K95E4iX4VaJqeWIS3r2uT7fY

[5] S. Latha and S. S. Kudakunti, "Face recognition using transform domain and overlapping lbp techniques," *International Journal of Engineering Research & Technology*, pp. 343–347, 2017. [Online]. Available: https://www.academia.edu/download/117147150/face-recognition-using-transform-domain-and-overlapping-lbp-techniques-IJERTCON pdf

[6] S. Jahan, "On dimension reduction using supervised distance preserving projection for face recognition," *Universal Journal of Applied Mathematics*, vol. 6, no. 3, pp. 94–105, 2018. [Online]. Available: https://www.academia.edu/download/84033308/UJAM3-12611697.pdf

[7] A. Kumar, A. K. Biswas, A. Kumar, and R. K. Yadav, "Optimising iot-based smart home systems using machine-learning algorithm," in *2022 7th International Conference on Communication and Electronics Systems (ICCES)*. IEEE, 2022, pp. 1133–1139. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9835769/?casa_token=dbbNS0IeQFoAAAAA:EDzBKGeCFozZFlbFe6yfDFZezvnbBVMmcDBqgZqQFPHn_hk1sAAJz7Pm67t6-3199arDqLN5zaN0rew

[8] E. Abusham, B. Ibrahim, K. Zia, and S. Al Maskari, "An integration of new digital image scrambling technique on pca-based face recognition system," *Scientific Programming*, vol. 2022, no. 1, p. 2628885, 2022. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1155/2022/2628885

[9] L. Shamir, "Evaluation of face datasets as tools for assessing the performance of face recognition methods," *International Journal of Computer Vision*, vol. 79, pp. 225–230, 2008. [Online]. Available: https://link.springer.com/article/10.1007/s11263-008-0143-7

[10] S. Sakthivel and R. Lakshmipathi, "Enhancing face recognition using improved dimensionality reduction and feature extraction algorithms–an evaluation with orl database," *International Journal of Engineering Science and Technology*, vol. 2, no. 6, pp. 2288–2295, 2010. [Online]. Available: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=694964dca170ea44963a1d6b2d8efd6addc62ebc

[11] C. Liu and H. Wechsler, "Independent component analysis of gabor features for face recognition," *IEEE Transactions on Neural Networks*, vol. 14, no. 4, pp. 919–928, 2003. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/1215407/?casa_token=5Tfx51T8yFYAAAAA:AUEaARs_ldhcIiZPhHJgu4ikHV9q-boiCBP8edBgyzbcWJtZi9-Alt_vAhI8_tNYoEuB3fOLvkVMlB0

[12] E. G. Ortiz and B. C. Becker, "Face recognition for web-scale datasets," *Computer Vision and Image Understanding*, vol. 118, pp. 153–170, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1077314213001744

[13] J. Tang, Q. Su, B. Su, S. Fong, W. Cao, and X. Gong, "Parallel ensemble learning of convolutional neural networks and local binary patterns for face recognition," *Computer Methods and Programs in Biomedicine*, vol. 197, p. 105622, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169260720314553

[14] T. S. Akheel, V. U. Shree, and S. A. Mastani, "Stochastic gradient descent linear collaborative discriminant regression classification based face recognition," *Evolutionary Intelligence*, vol. 15, no. 2, pp.

223–235, 2022. [Online]. Available: https://link.springer.com/article/10.1007/s12065-021-00585-y

[15] S. Z. Li and J. Lu, "Face recognition using the nearest feature line method," *IEEE Transactions on Neural Networks*, vol. 10, no. 2, pp. 439–443, 1999. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/750575

[16] M. Tamilselvi and S. Karthikeyan, "An ingenious face recognition system based on hrpsm_cnn under unrestrained environmental condition," *Alexandria Engineering Journal*, vol. 61, no. 6, pp. 4307–4321, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1110016821006451

[17] Q. Feng, J.-S. Pan, and L. Yan, "Restricted nearest feature line with ellipse for face recognition," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 3, no. 3, pp. 297–305, 2012. [Online]. Available: https://www.academia.edu/download/98458007/JIH-MSP-2012-03-006.pdf

[18] B.-L. Zhang, H. Zhang, and S. S. Ge, "Face recognition by applying wavelet subband representation and kernel associative memory," *IEEE Transactions on Neural Networks*, vol. 15, no. 1, pp. 166–177, 2004. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/1263588

[19] "Orl_faces dataset from kaggle." [Online]. Available: https://www.kaggle.com/datasets/sifboudjellal/orl-faces/data

[20] V. Kedia, S. R. Regmi, K. Jha, A. Bhatia, S. Dugar, and B. K. Shah, "Time efficient ios application for cardiovascular disease prediction using machine learning," in *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*. IEEE, 2021, pp. 869–874. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9418453

[21] J. Pearl, *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. Morgan Kaufmann, 1988. [Online]. Available: https://books.google.com/books?hl=en&lr=&id=mn2jBQAAQBAJ