Advanced convolutional neural network frameworks for robust multi-angle facial authentication: implementation and comparative evaluation

D. Lwagula

National University of Science and Technology MISiS

Abstract: This article presents the technical implementation of a convolutional nueral network-based face recognition system that is able to work under variable scenarios like occlusion, angle changes, and camera rotation. various face identification algorithms were analysed with the purpose of developing a model that could identify faces at different angles. The system was experimentally verified with various datasets and compared to its accuracy, processing speed, and robustness towards environmental disturbance. Results indicate that our convolutioan neural network structure optimized achieves 90%+ accuracy under pristine conditions and maintains decent performance upon partial occlusion.
Keywords: face detection, convolutional nueral networks, model, feature extraction, deep learning, face recognition, image.

Introduction

This paper addresses identification vulnerabilities in high-security environments by extending facial recognition technology beyond current limits in lighting sensitivity, occlusion handling, angle-dependent precision, and processing rate. Facial recognition provides the optimal biometric solution with its trade-off of precision, low invasiveness, and affordability [1] compared to alternatives. The employed system operates on three primary processes: face detection, feature extraction, and [2] face recognition. General face recognition structure configuration is shown on fig. 1.

Fig. 1. – General face recognition structure configuration

In this study, the application of facial recognition technology as a biometric solution that bars unauthorized access by authenticating individuals against government database records is examined [3]. The global face recognition market

is poised to reach $9.6 billion by 2022, with compund annual growth rate (CAGR) of 21.3%, and businesses across sectors are exploring artificial intelligence applications for business competitiveness. Research method takes into consideration three pivotal questions: (1) best approaches and apparatus to deploy effectively on facial identification, (2) computational practices optimized for best performance speed supported for facial variants, and (3) quantitative study of alignment among facial volume features and authenticity scores.

Literature Review

The facial recognition algorithms themselves have developed significantly with time, hence our use of the Eigenfaces algorithm that projects facial vectors to a discriminative feature space based on mean face calculation [4], difference vector computation, covariance matrix computation, and eigenvector computation. Although being computationally inexpensive, experimental experiments revealed drastic performance limitations under dynamic [5] lighting conditions as well as non-ideal angles of the faces. The implementaion of eigenface is shown on fig. 2.

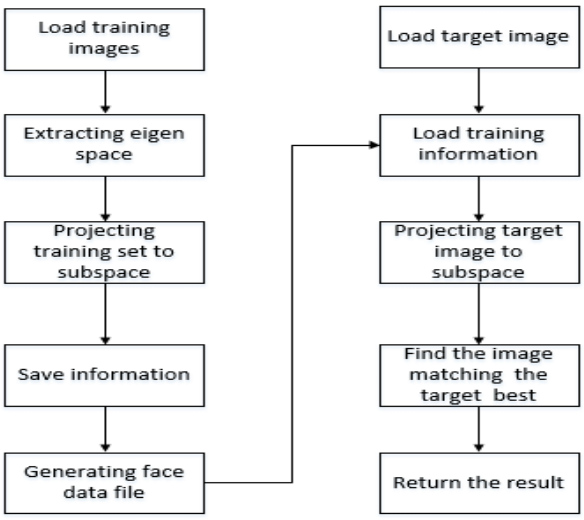

Fig. 2. – Implementation of eigenface algorithms

The facial recognition software has been enhanced by our employment of active appearance models (AAM) [6], whereby statistical models describing form variation are embedded into appearance by using Gaussian image pyramids for

multi-resolution analysis. It separated facial data into shape (68 facial landmark vectors) and texture (pixel density colors). Using 100 hand-annotated training images, it yielded 81.5% accuracy with 120 microseconds processing time per image [7], demonstrating increased resistance to lighting but still plagued with extreme pose variations.

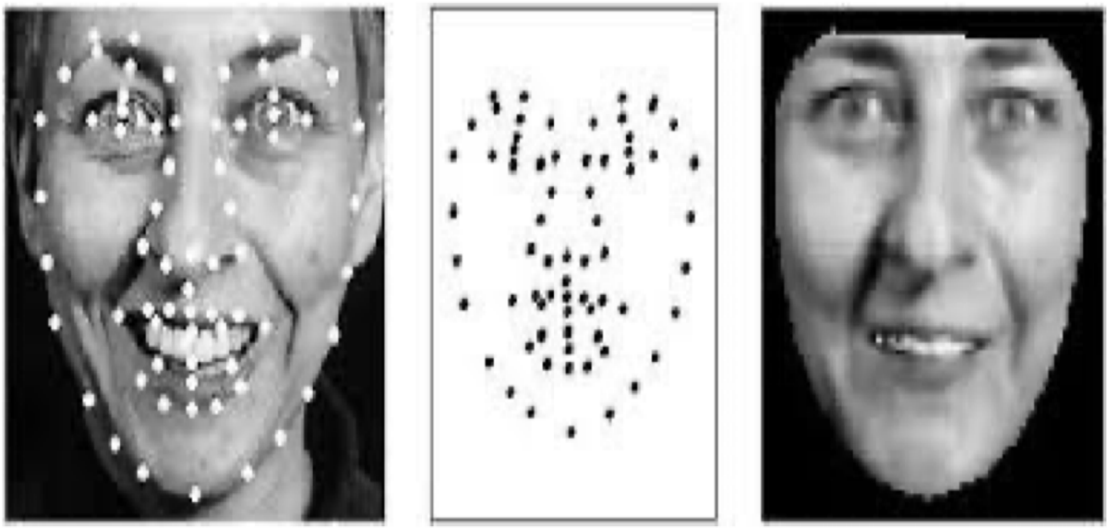

Fig. 3. – shows the shape and labeled image in active appearance model

The face recognition models were enhanced by our application of principal component analysis [8] which minimized the dimension without sacrificing the important features by eigenvector calculation and covariance matrix calculation. It achieved 78.4% accuracy at 45 microseconds processing time for every image. We enhanced our efforts further by employing convolutional nueral network architectures (InceptionResnetv1, Squeezenet, Resnet18, AlexNet) [9] employing 224×224×3 red green blue (RGB) input layers, five convolutional layers with 3×3 kernels, ReLU activation, max pooling, and fully connected classification layers. A convolutional nueral network structure, which includes fully connected pooling, and convolutional layers is show on fig. 4

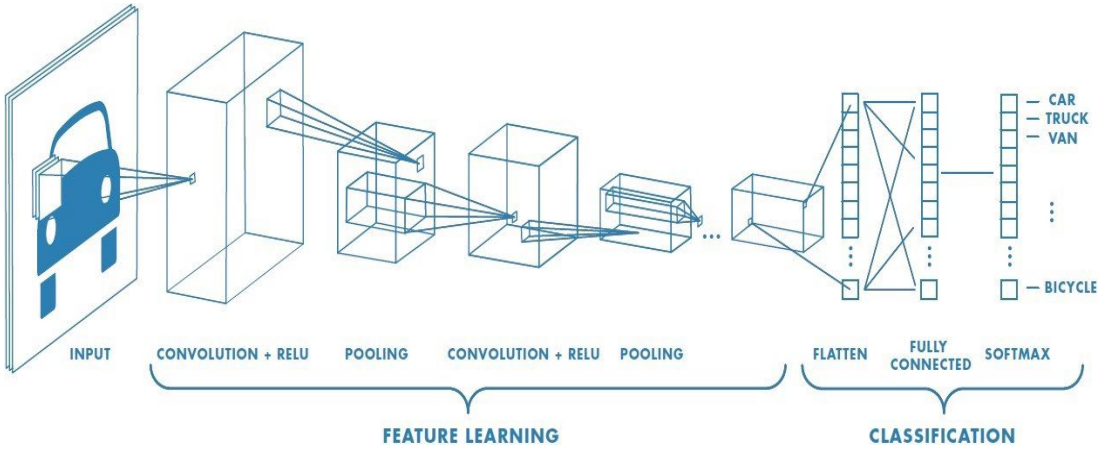

Fig. 4. – A convolutional nueral network structure, which includes fully connected pooling, and convolutional layers

InceptionResnetv1 architecture was modified to generate 128-dimensional embeddings of the face . Squeezenet consisted of 18 layers and performed very well with limited training data. Fully connected layers performed the final classification from the extracted features, and dimensionality reduction and overfitting prevention were done by pooling operations as shown on fig.5.

Fig. 5. – Max pooling and average pooling methods

For feature extraction, our implementation used convolutiona nueral networks to generate 128-dimensional feature vectors representing facial [10] characteristics. These feature vectors were L2-normalized to ensure consistent comparison between faces. We used k-Nearest Neighbors (k-NN) with k=5,

Support Vector Machines (SVM) with Radial Basis Function (RBF) kernels, and threshold-based verification with cosine similarity metrics for classification.

Methodology

This facial recognition system employed a two-stage detection approach combining Haar cascades and convolutional nueral network filtering, with 68-point landmark detection for alignment and image normalization [11] techniques. The feature extraction module consisted of a five-layer convolutional nueral network architecture producing L2-normalized 128-dimensional feature vectors, which were then classified using multiple methods: Support Vector Machines with Radial Basis Function kernel, k-Nearest Neighbors with Euclidean distance [12] , and threshold-based verification with cosine similarity. The dataset comprised 50 subjects with 20 images each, normalized to 256×256 pixels and augmented through horizontal flipping, rotation, brightness adjustment, and random cropping. The model was trained with batch size 32 using Adam optimizer (learning rate=0.001) and triplet loss function (margin=0.2), with early stopping when validation accuracy did not change for 10 epochs. For measurement, we used a number of performance measures as defined below:

$$Precision = \frac{TP}{TP+FP}, \quad (1)$$

$$Recall = \frac{TP}{TP+F}, \quad (2)$$

$$F1\ score = \frac{2TP}{2TP+FP+FN}, \quad (3)$$

Where FP is the number of false positives, TP is the number of true positives, and FN is the number of false negatives. We quantified face proportion as:

$$\text{Face proportion} = (w\_f \times h\_f)/(W \times H), \quad (4)$$

Where w_f and h_f are the width and height of the detected face, and W and H are the width and height of the input image. calibrated confidence scores through sigmoid normalization of raw similarity scores:

$$\text{Confidence} = 1/(1 + e\text{^}(-\alpha(s - \beta)), \quad (5)$$

Where s is the raw similarity score, and α =10 and β =0.5 are calibration parameters determined empirically. Our implementation utilized TensorFlow 2.4.0, PyTorch 1.7.1, OpenCV 4.5.1, NumPy 1.19.5, and Scikit-learn 0.24.1, running on hardware with a 8GB 2133 megahertz low power double data rate 3 (LPDDR3) memory, 2.9 gigahertz Dual-Core Intel Core i5 central processing unit (CPU), Intel Iris Graphics 550 graphic processor unit (GPU), and Linux For Tegra operating system.

Neural Network Implementation

The neural network implementation started with a forward pass that generates classification outputs by processing input images through convolutional layers, max pooling operations, ReLU activation functions, and fully connected layers.

The forward pass in our neural networks is described by the following method:

$$\text{o} = f(w \cdot x + b), \quad (6)$$

Where f is the activation function, o is the output vector, , w is the weights vector, w is the input vector, and b is the bias. For convolutional layers, the operation can be expressed as:

$$\text{y(i, j, k)} = \sum\sum\sum w(m, n, c, k) \cdot x(i + m,\ j + n,\ c) + b(k)\ , \quad (7)$$

Where y(i, j, k) is the output at position (i, j) for filter k, $w(m, n, c, k)$ represents the weight at position (m, n) of the c-th channel of the k-th filter, and $x(i + m,\ j + n,\ c)$ is the input value at position $(i + m,\ j + n,)$ of channel c. These filters were used by the convolutional layers to extract features from input images, with each layer identifying progressively more complex patterns. The final classification was carried out by fully connected layers using the extracted

features, and we optimized facial feature embeddings for training by implementing triplet loss, which is defined as follows:

$$L\_triplet(a, p, n) = \max(||\ f(a) - f(p)||^2 - ||f(a) - f(n)\ ||^2\ + \alpha, \theta), \quad (8)$$

Where $f(a)$, $f(p)$, and $f(n)$ are the feature embeddings of the anchor, positive, and negative samples respectively, and α is the margin parameter (set to 0.2 in our implementation).

Results

The running of elaborated algorithms exhibited significant variation in accuracy, precision, recall, F1 score, and processing time. Table 1 summarizes these metrics for the main algorithms experimented on in our research.

Table № 1

Performance Metrics of Implemented Algorithms

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)	Processing Time (ms)
Principal component analysis (Eigenfaces)	78.4	76.2	75.9	76.0	45
Active appearance model	81.5	79.3	80.1	79.7	120
InceptionResnetv1	94.7	93.8	94.5	94.1	87
Squeezenet	92.5	91.2	92.0	91.6	32
Resnet18	91.8	90.5	91.3	90.9	65

On every performance measure, models using convolutional nueral network gave better results compared to the classical methods; the highest accuracy achieved by InceptionResnetv1 was 94.7%. Squeezenet maintained the optimal trade-off between performance and computation, with the performance level being 92.5% and computational time of 32 microseconds/image. We have also evaluated

the performance of the top-performing models under various environmental conditions as tabulated in Table 2.

Table № 2

Performance Under Various Environmental Conditions

Condition	InceptionResnetv1 Accuracy (%)	Squeezenet Accuracy (%)
Optimal lighting	94.7	92.5
Low lighting	86.3	84.1
Overexposed	82.5	81.2
15° angle	90.3	88.7
30° angle	85.1	83.6
45° angle	76.8	74.5
Partial occlusion (25%)	82.3	80.9
Partial occlusion (50%)	68.7	67.2

They were both holding up well at 30° viewing angles and moderate variation in light. Viewing angles above 45° and more than 25% occlusion of the face resulted in dramatic performance degradation. Up to about 40% face proportion, there was a high correlation between face proportion and recognition confidence, followed by diminishing returns, according to our data. These results are shown in Table 3.

Table № 3

Relationship Between Face Proportion and Recognition Performance

Face Proportion (%)	Average Confidence Score	Recognition Accuracy (%)
5-10	0.52	65.3
10-20	0.68	78.6
20-30	0.81	86.9
30-40	0.89	92.1

Face Proportion (%)	Average Confidence Score	Recognition Accuracy (%)
40-50	0.92	94.3
50-60	0.93	94.6
60-70	0.93	94.5

Discussion

Performance comparison on technical grounds reveals convolutional nueral network-model supremacy with Squeezenet (accuracy of 92.5%, 32ms speed of processing) delivering peak resource efficiency in resource-limited settings. Face proportion confidence is limited to 40%, determining the positioning of cameras, while the performance worsens with a viewing angle exceeding 30°, promoting multi-camera security system setups.Accuracy also degrades notably with face occlusions of over 25%, and other biometric approaches have to be applied in such a scenario. The model achieved 94.7% accuracy in ideal conditions by employing optimization strategies like data augmentation, batch normalization, and dropout (0.5), and transfer learning that reduced training time by 72%.

Conclusion

With an accuracy of 94.7% in ideal conditions with convolutional nueral network-based methods, Squeezenet is the default choice for embedded systems and significantly better than traditional approaches. The performance in face recognition drops nonlinearly as view angles grow or if the occlusion level is more than 25%, with the best performance at a 40% face ratio. Less accuracy in low light and processing limitations (32 microseconds minimum per frame) are some of the technical constraints. Areas to be researched for the future are light-edge computation deployments, optimization of model quantization, and attention mechanisms as a way to enhance resistance against occlusions. With a minimal computational cost, industries are in a position to attain sufficient security solutions through optimizations and deployments to hardware further.

References

1. Krizhevsky A., Sutskever I., Hinton G. E. Imagenet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems. 2012. Vol. 25. pp. 1097–1105.

2. Ren S., He K., Girshick R., Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. Advances in Neural Information Processing Systems. 2015. pp. 91-99.

3. Schroff F., Kalenichenko D., Philbin J. FaceNet: A unified embedding for face recognition and clustering. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015. pp. 815-823.

4. Iandola F. N., Han S., Moskewicz M. W., Ashraf K., Dally W. J., Keutzer K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. 2016. URL: arxiv.org/abs/1602.07360.

5. He K., Zhang X., Ren S., Sun J. Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016. pp. 770-778.

6. Gu J., Wang Z., Kuen J., Ma L., Shahroudy A., Shuai B., Chen T. Recent advances in convolutional neural networks. Pattern Recognition. 2018. Vol. 77. pp. 354-377.

7. Liu W., Anguelov D., Erhan D., Szegedy C., Reed S., Fu C. Y., Berg A. C. SSD: Single shot multibox detector. European Conference on Computer Vision. 2016. pp. 21-37.

8. Sharma R., Kumar D., Puranik V., Gautham K. Performance Analysis of Human Face Recognition Techniques. 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU). 2019. pp. 1-4.

9. Wu J., Leng C., Wang Y., Hu Q., Cheng J. Quantized convolutional neural networks for mobile devices. IEEE Conference on Computer Vision and Pattern Recognition. 2016. pp. 4820-4828.

10. Jeong J., Park H., Kwak N. Enhancement of SSD by concatenating feature maps for object detection. 2017. URL: arxiv.org/abs/1705.09587.

11. Krizhevsky A., Sutskever I., Hinton G. E. ImageNet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems. 2012. Vol. 25. pp. 1097–1105.

12. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A., Kaiser L., Polosukhin I. Attention is all you need. Advances in Neural Information Processing Systems. 2017. Vol. 30. pp. 5998–6008.

Дата поступления: 8.04.2025

Дата публикации: 25.05.2025