Face Recognition via Pair Matching

liq23wr

University of Sheffield

Abstract

This paper presents a face verification system that classifies whether two images belong to the same person. The system uses data augmentation, Principal Component Analysis (PCA) for dimensionality reduction, and a stacking ensemble model of SVC, MLP, and GBC. The system achieves an accuracy of **69.8%** on the evaluation dataset, outperforming the baseline model. The report covers the methodology, hyperparameter optimization, results, and future improvements.

1. Introduction

Face verification determines whether two images represent the same person, with applications in security and identity verification. This task is challenging due to variations in lighting, pose, and expression.

The system uses data preprocessing, augmentation, PCA, and a stacking ensemble model to improve accuracy.

The report is organized as follows: Section 2 outlines the system architecture. Section 3 describes the experiments and hyperparameter tuning. Section 4 presents evaluation results, followed by an analysis of performance in Section 5. Section 6 concludes with suggestions for future improvements.

2. System Description

This system includes data preprocessing, augmentation, and ensemble classification. Below is a brief overview of the system:

2.1. Input Processing

Each image pair is reshaped into grayscale images of 62×47 pixels and concatenated into a single image of size 62×94 for further processing.

2.2. Data Augmentation

To improve generalization, the system applies several augmentations, including horizontal flips, brightness and contrast adjustments, sharpness, and Gaussian blur. The augmentation ratio is optimized experimentally, as discussed in the **Experiments** section.

2.3. Dimensionality Reduction

PCA reduces dimensionality while maintaining variance. Experimentation showed that $n = 70$ components worked best for performance and efficiency.

2.4. Model Training

The system employs an ensemble of classifiers combined into a stacking model:

- **Base Models:**
  - **Support Vector Classifier (SVC):** A non-linear classifier with a radial basis function (RBF) kernel.
  - **Multi-Layer Perceptron (MLPClassifier):** A feed-forward neural network optimized for classification.
  - **Gradient Boosting Classifier (GBC):** A tree-based ensemble model.
- **Stacking Ensemble[1]:** Predictions from the base models are combined using a **Logistic Regression** meta-classifier.

2.5. Pipeline and Output

The final system is encapsulated in a **scikit-learn Pipeline**, which streamlines the testing and deployment process. The pipeline includes:

- Preprocessing (image reshaping and flattening)
- Scaling (using *StandardScaler*)
- Dimensionality Reduction (via PCA)
- Classification (using the stacking ensemble)

It outputs a binary classification: Class 1 for the same person, Class 0 for different persons.

3. Experiments

Key areas of experimentation included the optimization of the augmentation ratio, hyperparameter tuning for the classifiers, and ensuring compliance with the model size constraint.

3.1. Augmentation Ratio Optimization

The **augmentation ratio** was systematically varied between 0.2 and 0.7 to identify the optimal value. This ratio is crucial:

- **If too low:** The dataset may not have sufficient diversity, leading to poor generalization[2].
- **If too high:** The model may become over-reliant on augmented data, reducing its accuracy on real, unaugmented test data[2].

Each ratio was tested using a Multi-Layer Perceptron (MLPClassifier) with 5-fold cross-validation. Table 1 shows the Cross-Validated accuracy at different augmentation ratios, with 0.5 achieving the best balance.

Table 1: *Effect of Augmentation Ratio on Cross-Validated Accuracy*

Augmentation Ratio	Cross-Validated Accuracy (%)
0.2	80.6
0.3	83.8
0.4	85.4
0.5	**86.8**
0.6	85.6
0.7	85.2

3.2. Hyperparameter Optimization

The next step involved optimizing the hyperparameters for dimensionality reduction and classifier training:

- **Dimensionality Reduction:** Principal Component Analysis (PCA) was used to reduce the high-dimensional input space. The number of components (n) was varied from 65 to 100, with $n = 70$ providing the highest accuracy (Table 2).

Table 2: *Effect of PCA Components on Cross-Validated Accuracy*

Number of Components	Cross-Validated Accuracy (%)
65	87.6
70	**88.6**
75	88.5
80	88.4
85	88.1
90	88.2
95	88.1
100	88.2

- **Classifier Hyperparameters:** Each base model in the stacking ensemble was tuned using *RandomizedSearchCV* with 5-fold cross-validation. Key parameters explored included:
  - **SVC:** Regularization parameter (C) and kernel coefficient (γ).
  - **MLPClassifier:** Hidden layer sizes, activation functions, and regularization parameter (α).
  - **Gradient Boosting Classifier (GBC):** Learning rate, number of estimators, and maximum depth.

3.3. Model Size Optimization

To ensure compliance with the 80 MB size constraint, the following measures were implemented:

- **Dimensionality Reduction:** PCA effectively reduced the size of intermediate data representations.
- **Ensemble Size:** The stacking classifier used only three base models to maintain compactness without sacrificing performance.

The final model size was approximately 27 MB, well within the allowed limit.

4. Results and Analysis

4.1. Evaluation Results

The final system achieved an accuracy of **69.8%** on the evaluation dataset, significantly outperforming the baseline model's accuracy of **56.3%**. Table 3 summarizes the results.

Table 3: *Evaluation Results on the Test Dataset*

Model	Accuracy (%)	Improvement (%)
Baseline Model	56.3	-
Final System	69.8	+13.5

4.2. Confusion Matrix

The confusion matrix for the final system is shown in Table 4, based on the provided values:

Table 4: *Confusion Matrix for the Final System*

Predicted	Same Person (1)	Different Person (0)
True Same Person (1)	357	143
True Different Person (0)	159	341

Where:

- **True Positives (TP)**: 357 same-class pairs correctly identified as "same".
- **True Negatives (TN)**: 341 different-class pairs correctly identified as "different".
- **False Positives (FP)**: 159 different-class pairs incorrectly identified as "same".
- **False Negatives (FN)**: 143 same-class pairs incorrectly identified as "different".

4.3. Failure Analysis

Misclassifications were analyzed to understand the limitations:

- **Low-Quality Images:** Blurred or noisy images often led to false negatives.
- **Pose and Occlusion Variability:** Extreme pose differences or occlusions like sunglasses and hats caused false positives.
- **Subtle Differences:** Small variations in same-class pairs, such as lighting or aging, occasionally led to errors.

5. Discussion and Conclusions

The system achieved **69.8%** accuracy, significantly outperforming the baseline model. The system's performance was sensitive to hyperparameters, particularly the augmentation ratio and PCA components, with 0.5 augmentation ratio and $n = 70$ components performing best.

Future work could focus on expanding augmentation methods, exploring deep learning-based feature extraction, and refining hyperparameter tuning to improve generalization and robustness.

In conclusion, the system demonstrates strong performance, but addressing limitations related to pose variability and subtle facial differences could further improve its robustness in real-world applications.

6. References

[1] A. S. D. K. A. Kurenkov and M. Lee, "Ct image classification based on stacked ensemble of convolutional neural networks," *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Nov. 2022.

[2] A. Mikołajczyk and M. Grochowski, "Data augmentation for improving deep learning in image classification problem," *2018 International Interdisciplinary PhD Workshop (IIPhDW)*, Jun. 2018.