Department of Computer Engineering

Bu-Ali Sina University

Pattern Recognition Course

Gender Classification from Facial Images using Machine Learning and Feature Extraction Techniques: A Comparative Analysis

By:

Hossein Shakibania

Academic Supervisor:

Professor Hassan Khotanlou

Spring 2023

DECLARATION

I certify that I am the author of this project and also cited any source from which I used data, ideas, or words, either quoted or paraphrased.

No portion of the work referred to in this study has been submitted in support of an application for another degree or qualification to this or any other university or institution of learning.

Hossein Shakibania July 2023

ABSTRACT

Gender classification from facial images is a vital task in computer vision and machine learning, with applications ranging from human-computer interaction to marketing and security systems. In this project, we present a detailed investigation into this problem, employing various machine learning algorithms and feature extraction techniques. The focus is on leveraging the UTKFace dataset, a large-scale collection of face images annotated with age, gender, and ethnicity. Our comprehensive approach involves applying the Histogram of Oriented Gradients (HOG) and Local Binary Pattern (LBP) methods for feature extraction and employing Principal Component Analysis (PCA) for dimensionality reduction. By effectively reducing the dimensions of the problem from 64x64 to 256 features, PCA aids in faster training times and improved classification performance in most cases. Our findings reveal that the combination of the Support Vector Machine (SVM) classifier with HOG and PCA with 256 components achieves the best performance in gender classification, providing valuable insights for similar tasks in the future. The code for this project is available at github.com/hossshakiba/Face2Gender.

Keywords: Classification, Machine Learning, Feature Extraction, Dimensionality Reduction

TABLE OF CONTENTS

1 INTRODUCTION

Gender classification from facial images is an essential and challenging problem in the domain of computer vision and machine learning. The ability to automatically predict the gender of individuals based on facial features has wide-ranging applications in various fields, such as human-computer interaction, marketing, and security systems. Accurate gender classification can offer valuable insights for demographic analysis, personalized services, and targeted advertising campaigns. Moreover, it can be a crucial component in building inclusive and unbiased artificial intelligence systems.

In recent years, significant progress has been made in the development of sophisticated algorithms and feature extraction techniques that can leverage the wealth of information present in facial images to accurately classify gender. These advancements have been driven by the availability of large-scale facial image datasets and the increased computational power of modern hardware. As a result, gender classification has become a compelling research topic with practical implications in real-world scenarios.

In this project, we undertake an extensive investigation into gender classification from facial images using various methodologies. The key objectives encompass data preprocessing, feature extraction, and machine learning model experimentation. To ensure data consistency, we curate and preprocess a diverse dataset of facial images containing male and female subjects. We then apply face detection and cropping techniques to extract relevant facial regions for accurate feature extraction.

The project primarily focuses on two feature extraction techniques: Histogram of Oriented Gradients (HOG) [1] and Local Binary Pattern (LBP) [2]. Through the application of these techniques, we capture distinct characteristics from facial images that may contribute to gender classification. Additionally, we explore an array of machine learning algorithms, such as Support Vector Machines (SVM) [3], k-Nearest Neighbors (k-NN) [4], and Random Forest (RF) [5], to build gender classification models. By adopting a rigorous experimental approach, we evaluate the performance of each method using comprehensive metrics to understand their strengths and limitations. Furthermore, we analyze the impact of varying the number of principal components in Principal Component Analysis (PCA) [6] on classification accuracy, leading to insights into dimensionality reduction and its influence on gender prediction.

2 METHODOLOGY

2.1 Preprocessing

In this section, we present the preprocessing steps carried out on the raw dataset of facial images to prepare them for gender classification. Preprocessing is a crucial stage in any computer vision task, as it helps to enhance the quality of the data and extract meaningful information for subsequent analysis.

2.1.1 Pretrained Face Detection Algorithm (MTCNN)

To accurately extract facial regions from the wild images, we employed the Multi-Task Cascaded Convolutional Networks (MTCNN) [7] algorithm, a widely-used and efficient face detection technique. MTCNN is a deep learning-based approach that simultaneously detects facial landmarks and bounding boxes, enabling precise localization of faces in images.
MTCNN consists of three stages:

1. **Proposal Network (P-Net):** This stage generates candidate bounding boxes for potential faces in the image using a sliding window technique and a deep neural network to classify each window as containing a face or not.

2. **Refinement Network (R-Net):** In this stage, the candidate bounding boxes from the previous stage are refined by regressing the bounding box coordinates and filtering out false detections based on the confidence scores obtained from the neural network.

3. **Output Network (O-Net):** Finally, the candidate bounding boxes are further refined to accurately align with facial landmarks, such as the eyes, nose, and mouth.

The application of MTCNN allowed us to efficiently detect and extract facial regions from the wild images. By focusing solely on the facial regions, we eliminated extraneous background information and any irrelevant features unrelated to gender classification. By removing useless parts of images, we also simplified the dataset and enhanced the performance of subsequent feature extraction and machine learning algorithms for gender classification. Figure 1 showcases the application of MTCNN on four diverse facial images from different age and gender groups.

2.1.2 Image Conversion and Resizing

Initially, the raw dataset contained images in their original format, with varying color channels (RGB). To simplify the computational complexity and to focus on essential facial features for gender

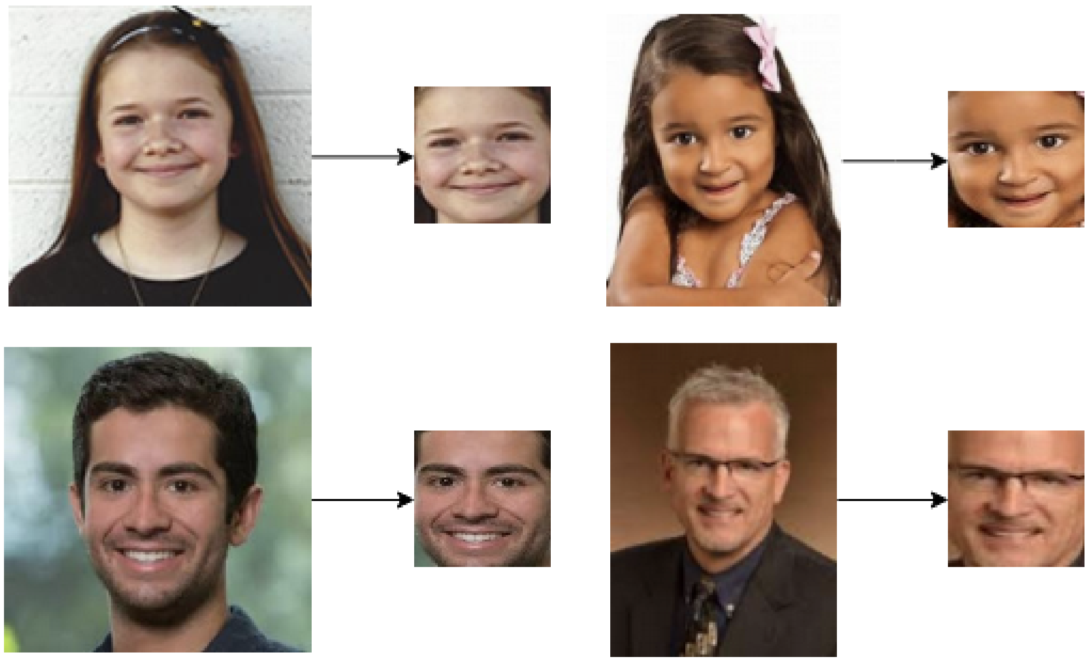


Figure 1: Face detection using MTCNN algorithm.

classification, the images were converted to grayscale. Grayscale images retain the luminance information of the original images, providing sufficient detail while reducing the dimensionality.

Moreover, the images were resized to a uniform size of 64x64 pixels. Resizing the images standardizes the data and facilitates efficient processing in subsequent stages. The selected size of 64x64 pixels balances the trade-off between preserving facial features and avoiding excessive computational overhead.

2.1.3 Manual Wrong Cropped Faces Deletion

During the process of face detection and cropping, there may be instances where the face detection algorithm fails to accurately identify and crop faces from certain images. To address this issue, a manual review of the cropped faces was conducted. Images that were incorrectly cropped or contained irrelevant regions were carefully inspected and removed from the dataset. This step ensures that the dataset used for training and evaluation contains high-quality face images with accurate cropping, thus improving the overall performance and reliability of the gender prediction models.

2.2 Feature Extraction

2.2.1 Histogram of Oriented Gradients (HOG)

Histogram of Oriented Gradients (HOG) is a popular feature extraction technique widely used in computer vision tasks, including gender classification. The HOG algorithm works by capturing local gradient information from an image, which provides a concise representation of the image's texture and shape. The HOG feature extraction process involves the following steps:

1. **Gradient Computation:** The algorithm calculates the gradient of the grayscale image by applying convolution with specific filters, such as the Sobel operator. This step captures the direction and magnitude of edges in the image.

2. **Cell Division:** The image is divided into small, overlapping cells. Within each cell, the histogram of gradient orientations is computed, effectively summarizing the dominant edge orientations in that region.

3. **Block Normalization:** To improve robustness to changes in lighting and contrast, adjacent cells' histograms are combined by block normalization. Normalization helps to make the features more resistant to illumination variations.

4. **Sliding Window:** A sliding window approach is used to extract HOG features from different parts of the image, capturing texture and shape information from all regions.

The resulting HOG features serve as a compact representation of the image's structural characteristics, which are essential for gender classification [1]. HOG has proven effective in handling variations in pose, illumination, and facial expressions, making it a suitable choice for our gender classification task.

2.2.2 Local Binary Pattern (LBP)

Local Binary Pattern (LBP) is another powerful feature extraction method used extensively in facial recognition and gender classification tasks. LBP operates on grayscale images and is particularly robust against illumination variations, making it suitable for processing facial data.

The LBP algorithm extracts features by comparing the intensity of the central pixel with its neighboring pixels. It proceeds as follows:

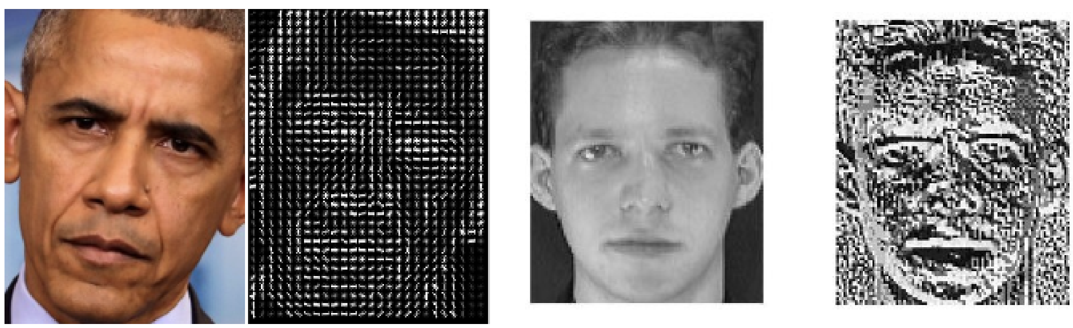
1. **Neighborhood Comparison:** For each pixel in the image, its intensity is compared to the intensities of its neighboring pixels. The neighboring pixels are thresholded, resulting in a binary pattern (0 or 1) for each neighbor based on whether it is greater or smaller than the central pixel.

2. **Pattern Encoding:** The binary patterns obtained from the neighborhood comparison are concatenated to form an LBP code for the central pixel.

3. **Histogram Calculation:** An LBP histogram is computed by counting the occurrences of different LBP codes within a region of interest (ROI), typically the facial region in gender classification [2].

The LBP features effectively capture local texture information in the facial region, which is essential for gender classification. LBP is computationally efficient and requires minimal preprocessing, making it a suitable choice for real-time applications.

By utilizing both HOG and LBP feature extraction techniques, we aim to leverage complementary information from facial images, facilitating gender classification with improved accuracy and robustness to variations in pose, illumination, and facial expressions. Figure 2 demonstrates the application of HOG and LBP feature extraction methods on the face images



(a) HOG　　　　(b) LBP

Figure 2: Feature Extraction using (a) HOG and (b) LBP.

2.3 Dimensionality Reduction

Dimensionality reduction is a crucial technique in machine learning and data analysis, primarily employed to tackle the "curse of dimensionality." The curse of dimensionality refers to the problem of dealing with high-dimensional datasets, which can lead to increased computational complexity, reduced model performance, and increased risk of overfitting. Dimensionality reduction methods aim to reduce the number of features or variables in the data while preserving as much relevant information as possible. By transforming the data into a lower-dimensional space, dimensionality reduction techniques facilitate better visualization, faster computation, and improved generalization of machine learning models.

2.3.1 Principal Component Analysis (PCA)

PCA is one of the most widely used dimensionality reduction techniques in various applications, including computer vision, signal processing, and data analysis. It aims to transform the data into a new coordinate system where the first axis (principal component) captures the maximum variance in the data, the second axis captures the second most significant variance, and so on. In this way, PCA seeks to identify the most informative directions in the data and eliminate less important dimensions.

Let's consider a dataset with multiple features, each representing a dimension. PCA calculates the eigenvectors (principal components) of the covariance matrix of the data and sorts them based on their corresponding eigenvalues (explained variance). The top k eigenvectors, where k is the desired reduced dimensionality, form a projection matrix. By multiplying the data with this projection matrix, we obtain the transformed dataset with reduced dimensions [6].

In this project, PCA is utilized as a dimensionality reduction method for feature extraction. Instead of directly using raw pixel values, PCA is applied to obtain a lower-dimensional representation of the data. By reducing the dimensionality while retaining the most significant variance, PCA not only accelerates computation but also enhances the model's ability to generalize to new, unseen data. Furthermore, the reduced features can be used as input to various machine learning algorithms to predict the gender of faces in the dataset accurately. Figure 3 illustrates the application of PCA with three different principal components, showcasing how the data is transformed and represented in a lower-dimensional space.

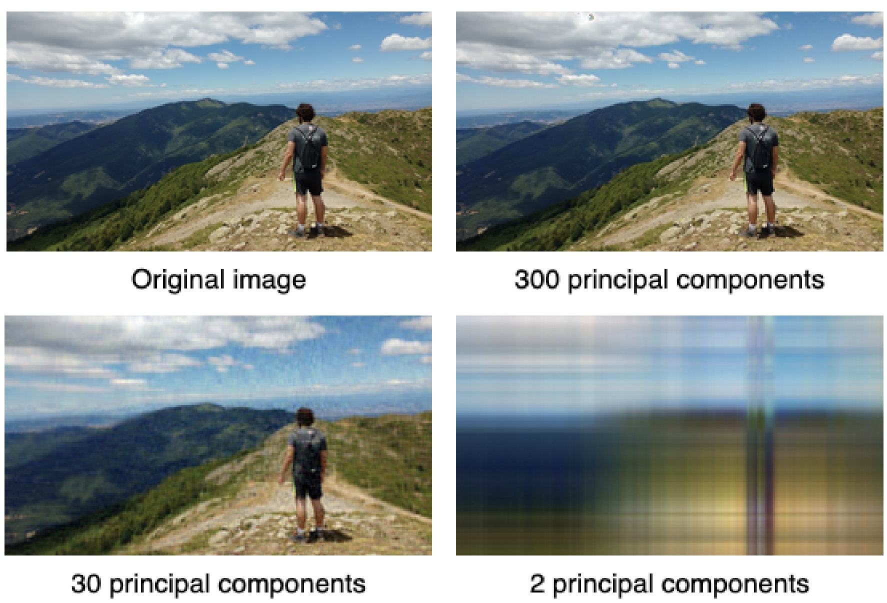


Figure 3: Application of PCA with three different principal components.

2.4 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a powerful supervised learning algorithm used for both classification and regression tasks. SVM aims to find an optimal hyperplane that best separates the data points into different classes in the feature space. It works by maximizing the margin between the support vectors, which are the data points closest to the decision boundary [3]. SVM is effective in handling high-dimensional feature spaces and can handle non-linear relationships between features by using the kernel trick. It is widely used for classification tasks and has shown excellent performance in various applications, including image recognition, text classification, and bioinformatics. In this project, we utilized the SVC (Support Vector Classification) implementation from Scikit-learn, which is specifically designed for binary classification tasks.

2.5 k-Nearest Neighbor (k-NN)

k-Nearest Neighbor (k-NN) is a simple and intuitive classification algorithm that works based on the principle of similarity. Given a new data point, k-NN classifies it based on the majority class of its k-nearest neighbors in the training dataset [4]. The choice of k is a critical parameter in k-NN, as smaller values make the model more sensitive to noise, while larger values may oversimplify the decision boundaries. k-NN is non-parametric and lazy, meaning it does not require training a model and instead stores all the training data for prediction. While k-NN is straightforward and easy to implement, it may suffer from computational inefficiency and sensitivity to the choice of distance metric. In this project, we set the value of k to 5 for the algorithm.

2.6 Random Forest

Random Forest is an ensemble learning algorithm that combines the power of multiple decision trees to improve the accuracy and generalization of the model. It works by building multiple decision trees using random subsets of the training data and features. Each tree in the forest makes a prediction, and the final output is determined by majority voting [5]. Random Forest addresses the issue of overfitting and enhances the robustness of the model by reducing variance. It is effective in handling high-dimensional data and can capture complex interactions between features. Random Forest has become a popular choice in various machine learning tasks, including classification, regression, and feature selection.

3 EXPERIMENTAL RESULTS

In this study, we conducted a series of comprehensive experiments to explore the performance of different machine learning algorithms on our curated dataset. The experiments were carried out on hardware equipped with the M1 Max Apple chip and the Scikit-learn library was utilized for code implementation.

3.1 Dataset

The UTKFace dataset served as the foundation for our project, providing a vast collection of face images with a wide age range, spanning from newborns to elderly individuals up to 116 years old. This extensive dataset comprises over 20,000 face images, each annotated with essential attributes, including age, gender, and ethnicity. The images exhibit significant variations in terms of pose, facial expression, illumination, occlusion, and resolution, offering a diverse and challenging dataset for various tasks, such as face detection, age estimation, age progression/regression, and landmark localization [8].

For our project, we selected a small proportion of the UTKFace dataset, utilizing only 15% (3679) of the entire collection to facilitate efficient experimentation. To ensure robust model training and evaluation, we divided the dataset into two distinct sets: 70% for training and 30% for testing.

3.2 Evaluation Metrics

During the evaluation of our models, we employed several key metrics to assess their performance. Accuracy, sensitivity (recall), and F1-score were among the metrics used to evaluate the effectiveness of different approaches used in this project. We used macro averaging for evaluating the performance of our models.

Accuracy is a metric that quantifies the ratio of correct predictions made by the model to the total number of predictions (Equation 1). A higher accuracy indicates that more predictions were correct.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{1}$$

The variables TP, TN, FP, and FN are defined as True Positive, True Negative, False Positive, and False Negative, respectively. Precision, as indicated in Equation (2), measures the ratio of TPs to the total number of positive predictions.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

Sensitivity, also known as recall, measures the proportion of actual positive instances that are correctly identified by the model (Equation 3).

$$Sensitivity = \frac{TP}{TP + FN} \tag{3}$$

F1-score, as defined by Equation 4, is calculated as the harmonic mean of precision and sensitivity, and ranges from 0 to 1, where higher scores indicate superior performance.

$$F1 - Score = 2 \times \frac{Precision \times Sensitivity}{Precision + Sensitivity} \tag{4}$$

3.3 Results

The performance of the gender classification models was evaluated using various metrics, including accuracy, precision, sensitivity, and F1-score. Table 1 displays the performance comparison of classifiers with and without PCA using features extracted with the Local Binary Patterns (LBP) method. It can be observed that the SVM classifier achieved an accuracy of 80.49% without PCA, while the accuracy slightly decreased to 79.84% with PCA using 1024 components. Similarly, the Random Forest classifier achieved an accuracy of 78.29% without PCA and 68.22% with PCA using 1024 components. Also, the k-Nearest Neighbors classifier attained an accuracy of 75.32% without PCA and 75.19% with PCA using 1024 components. Interestingly, the application of PCA with 256 components resulted in better performance in most cases, with improved accuracy observed across different classifiers.

Table 1: Performance comparison of classifiers with and without PCA using features extracted with the LBP method.

Classifier	Accuracy (%)	Precision (%)	Sensitivity (%)	F-1 Score (%)
SVM				
without PCA	**80.49**	**80.49**	**80.49**	**80.49**
PCA (1024)	79.84	79.84	79.84	79.84
PCA (512)	79.33	79.33	79.33	79.33
PCA (256)	79.07	79.11	79.07	79.06
Random Forest				
without PCA	78.29	78.40	78.30	78.28
PCA (1024)	68.22	68.48	68.23	68.12
PCA (512)	69.77	70.41	69.77	69.54
PCA (256)	77.00	77.50	77.02	76.91
5-Nearest Neighbor				
without PCA	75.32	75.77	75.34	75.22
PCA (1024)	75.19	75.67	75.19	75.09
PCA (512)	75.84	76.23	75.86	75.76
PCA (256)	77.13	77.57	77.15	77.05

Table 2: Performance comparison of classifiers with and without PCA using features extracted with the HOG method.

Classifier	Accuracy (%)	Precision (%)	Sensitivity (%)	F-1 Score (%)
SVM				
without PCA	79.07	79.08	79.07	79.07
PCA (1024)	79.07	79.08	79.07	79.07
PCA (512)	80.23	80.26	80.24	80.23
PCA (256)	**80.88**	**80.88**	**80.88**	**80.88**
Random Forest				
without PCA	78.29	78.40	78.30	78.28
PCA (1024)	68.99	69.89	69.02	68.66
PCA (512)	71.96	72.82	71.99	71.71
PCA (256)	75.45	75.73	75.47	75.39
5-Nearest Neighbor				
without PCA	75.32	75.77	75.34	75.22
PCA (1024)	75.19	75.66	75.21	75.09
PCA (512)	75.32	75.67	75.34	75.25
PCA (256)	75.97	76.34	75.98	75.89

Table 2 presents the performance comparison of classifiers with and without PCA using features extracted with the Histogram of Oriented Gradients (HOG) method. The SVM classifier achieved an accuracy of 79.07% without PCA, and the accuracy remained the same (79.07%) with PCA using 1024 components. The Random Forest classifier achieved an accuracy of 78.29% without PCA, and the accuracy decreased to 68.99% with PCA using 1024 components. Similarly, the k-Nearest Neighbors experienced a decline, dropping from 75.32% without PCA to 75.19% with PCA using 1024 components.

The application of PCA resulted in a reduction in feature dimensionality, which facilitated the classification task and led to shorter training times. However, the performance of the classifiers was not significantly impacted by the use of PCA, as evidenced by the marginal changes in accuracy. Both the HOG and LBP feature extraction methods provided valuable information for gender classification, as demonstrated by the relatively high accuracy achieved by the classifiers. Overall, the results of the experiments indicate that the combination of the SVM classifier with the HOG feature extraction method yielded the best performance for gender classification, achieving an accuracy of 80.88% with PCA using 256 components.

4 CONCLUSION

In conclusion, this project aimed to predict the gender of faces using machine learning algorithms and various feature extraction techniques. A proportion of 15% of the UTKFace dataset, consisting of over 20,000 face images, was utilized for training and evaluation. PCA was applied to reduce the dimensions of the problem from 4096 to only 256 features, resulting in faster training times. Additionally, the implementation of PCA with 256 components yielded slightly better accuracy compared to using the original 64x64 image size. Moreover, the application of PCA helped to eliminate irrelevant information, leading to more efficient and effective gender classification. Overall, the combination of the Support Vector Machine (SVM) classifier with the Histogram of Oriented Gradients (HOG) feature extraction method and PCA with 256 components achieved the best performance in gender classification, providing a promising approach for similar tasks in the future.

REFERENCES

[1] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, pp. 886–893, 2005.

[2] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[3] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[4] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on information theory*, vol. 13, no. 1, pp. 21–27, 1967.

[5] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[6] I. T. Jolliffe, *Principal Component Analysis*. Springer, 2002.

[7] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016. DOI: 10.1109/LSP.2016.2603342.

[8] Yang Song, Zhifei Zhang, *Utkface*, https://susanqq.github.io/UTKFace/, Accessed on Jul 20, 2023.