Shahid Beheshti University
Data Science (2023-2024)

Final project report

Pouya Mirzaie Zadeh
Department of Computer Science
Shahid Beheshti University
p.mirzaiyzadeh@cs.sbu.ac.ir
And Shadi Sefidgar
Department of Computer Science
Shahid Beheshti University
shd.wnkx@gmail.com

Abstract

This report presents a comprehensive study on age detection using deep learning techniques. The primary focus is on developing a robust and efficient model that can accurately predict the age of individuals based on given data. The proposed solution leverages advanced neural network architectures and various pre-processing techniques to enhance the model's performance.

1 Introduction

Age detection has emerged as a significant challenge in the field of data science and artificial intelligence due to its vast applications in areas like surveillance, social media, marketing, and healthcare. The task involves predicting an individual's age, which is a complex problem due to the subtle and non-linear transformations that human appearance undergoes with age. This report aims to address this problem by proposing a novel method that combines effective preprocessing techniques with a sophisticated neural network architecture. The challenges faced during this task, such as data imbalance and the high dimensionality of data, are also discussed.

2 Related work/Background

n recent years, several approaches have been proposed to tackle the problem of age detection.

One such approach is the use of Convolutional Neural Networks (CNNs). CNNs have shown promising results in image classification tasks, including age detection. These networks automatically learn hierarchical feature representations from raw data, eliminating the need for manual feature extraction. However, the performance of CNNs can be significantly affected by the quality and diversity of the training data.

Another approach is the application of Transfer Learning, where pre-trained models on large-scale datasets are fine-tuned for age detection. This method leverages the knowledge gained from the initial training and applies it to the new task, thereby reducing the need for large amounts of labeled data. However, the effectiveness of transfer learning largely depends on the similarity between the source and target tasks.

Lastly, Ensemble Learning methods have also been explored. These methods combine multiple models to make a final prediction, which often leads to improved performance. The diversity among the individual models is a key factor in the success of ensemble methods.

(LateX template borrowed from NeurIPS 2019)

In this report, we propose a method that builds upon these previous works, aiming to improve the accuracy of age detection by addressing some of the limitations of the existing methods.

3 Proposed method

Our proposed method for age detection involves several steps, including data cleaning, image pre-processing, face detection, face alignment, lighting and color normalization, data augmentation, and feature extraction.

- Data Cleaning: The initial dataset was cleaned to ensure consistency and quality. This involved validating file paths, removing duplicates, and ensuring split consistency.
- Image Pre-processing: Images were resized to a standard size of 64x64 pixels and normalized to a range of 0 to 1.
- Face Detection: We used the dlib library's frontal face detector to identify images containing faces.
- Lighting and Color Normalization: The aligned face images were then normalized to account for variations in lighting and color.
- Data Augmentation: To increase the robustness of our model, we augmented our data by applying random rotations to the normalized face images.
- Face Alignment: Faces in the images were aligned using a shape predictor.
- Wrinkle Features: We enhanced the wrinkles in the images using Canny edge detection and computed the percentage of white pixels in the edges to serve as a wrinkle feature.
- Hair Color Extraction: We converted the images to HSV color space for better color analysis and defined a mask for the hair region. We then applied this mask to the original image and calculated the dominant hair color.
- Facial Landmarks: We detected facial landmarks in each image using dlib's pre-trained facial landmark predictor. These landmarks provide crucial information about the structure of the face and can be used to extract more specific features.
- Eye Openness Detection: This part is yet to be implemented, but the idea is to detect the openness of the eyes, which can be a useful feature for age detection.
- Data Exclusion: During the feature extraction process, we encountered certain images that were incompatible with our methods. For instance, one particular image could not be processed correctly by our feature extraction methods. After careful consideration, we decided to exclude this image from our dataset to ensure the integrity and consistency of our feature set. This decision was made based on the premise that including this image could potentially skew our model's performance and provide misleading results. Future work may look into refining our feature extraction methods to handle such exceptions.

4 Data Analysis

In our dataset, the age distribution varies across different age groups as shown in Figure 1. The age groups 18-20, 31-40, and 41-50 have similar sample counts, indicating a balanced representation in these ranges. However, there are slightly fewer samples in the 21-30 age group and significantly fewer in the 51-60 age group. This uneven distribution could potentially influence our model's performance, as it might be less accurate in predicting ages that are underrepresented in the training data. Therefore, we took this into account when developing our age detection method.

The provided data represents the mean and standard deviation of certain features extracted from a set of images. Each row corresponds to an image, identified by an index number ranging from 0 to 147.

Analyzing the data, we observe that the mean values range approximately between 0.14 and 0.74, while the standard deviations range approximately between 0.10 and 0.36. This suggests a certain level of variability in the features across the different images.

It's important to note that the mean and standard deviation are measures of central tendency and dispersion, respectively. The mean provides a summary of the central location of the data, while the standard deviation quantifies its amount of variation or dispersion.

A high standard deviation can indicate that the data points are spread out from the mean, while a low standard deviation indicates that the data points tend to be close to the mean. Therefore, images with higher standard deviations may have features that vary more than those with lower standard deviations.

The graph “Texture Features Across Age Groups” presents the distribution of texture features across five different age groups: 18-20, 21-30, 31-40, 41-50, and 51-60. The x-axis represents the texture features, which range from 0 to 160, while the y-axis represents the density of these features, ranging from 0 to 6.

Each age group is represented by a different color, with corresponding bar and line graphs. The height of the bars and the trend of the line graphs indicate the density of texture features for each age group at different feature values.

From the graph, we can observe how the texture features vary across different age groups. However, without additional context or specific data points, further interpretation of the graph is limited.

4.1 Neural Network Architecture CNN

Our model is a combination of a Convolutional Neural Network (CNN) for image feature extraction and a Dense network for tabular features. The CNN consists of two convolutional layers, each followed by a max-pooling layer. The convolutional layers have 32 and 64 filters respectively, and both use a 3x3 kernel and the ReLU activation function. The max-pooling layers use a 2x2 pool size. The output of the CNN is flattened and then passed through a dense layer with 128 units and ReLU activation.

The tabular features are input into a dense layer with 128 units and ReLU activation. The outputs of the CNN and the dense network are then concatenated and passed through another dense layer with 64 units and ReLU activation. The final output layer uses a linear activation function, as this is a regression problem.

4.2 Training Process

The model is compiled with the Adam optimizer and the mean squared error loss function, suitable for our regression task. The model is trained for 10 epochs with a batch size of 32. We also use a validation split of 0.2 to monitor the model’s performance on unseen data during training.

4.3 Performance Metrics

After training, the model’s performance is evaluated on the test set. We calculate the Mean Absolute Error (MAE) and Mean Squared Error (MSE) to quantify the model’s regression performance. We also calculate the F1 score for a binary classification task, assuming a threshold of 0 for age detection. The F1 score is the harmonic mean of precision and recall and provides a better measure of the incorrectly classified cases than the Accuracy Metric.

4.4 Neural Network Architecture VVGFace

Our model is a combination of a pre-trained VGGFace model for image feature extraction and a Dense network for tabular features. The VGGFace model, based on the VGG16 architecture, is used as a feature extractor, with its top layers removed. The output of the VGGFace model is flattened and then passed through a dense layer with 128 units and ReLU activation.

The tabular features are input into a dense layer with 128 units and ReLU activation. The outputs of the VGGFace model and the dense network are then concatenated and passed through another dense layer with 64 units and ReLU activation. The final output layer uses a linear activation function, as this is a regression problem.

4.5 Training Process

The model is compiled with the Adam optimizer and the mean squared error loss function, suitable for our regression task. The model is trained for 10 epochs with a batch size of 32. We also use a validation split of 0.2 to monitor the model’s performance on unseen data during training.

4.6 Performance Metrics

After training, the model's performance is evaluated on the test set. We calculate the Mean Absolute Error (MAE) and Mean Squared Error (MSE) to quantify the model's regression performance. We also calculate the F1 score for a binary classification task, assuming a threshold of 0 for age detection. The F1 score is the harmonic mean of precision and recall and provides a better measure of the incorrectly classified cases than the Accuracy Metric.

5 Results

Our experiments involved two models: a Convolutional Neural Network (CNN) and a pre-trained VGGFace model.

5.1 Model Performance

The performance of the models was evaluated using three metrics: Mean Absolute Error (MAE), Mean Squared Error (MSE), and F1 Score. The results are as follows:

5.2 Comparison with Other Models

Both models performed comparably on our dataset, with the CNN model achieving slightly lower MAE and MSE values than the VGGFace model. This suggests that the CNN model was slightly more accurate in predicting ages. However, both models achieved the same F1 score, indicating similar performance in terms of precision and recall.

5.3 Significant Findings

An interesting finding from our experiments was that feeding the 'normalized_face' to the model resulted in better performance than using the original 'image' column. This suggests that the normalization process, which likely involved techniques such as resizing, grayscale conversion, and pixel value normalization, was beneficial in reducing noise and variability in the images, thereby helping the model focus on the essential features for age prediction.

5.4 Interpretation of Results

The results indicate that both the CNN and VGGFace models are capable of predicting ages with reasonable accuracy. The similar performance of the two models suggests that the additional complexity and computational cost of the VGGFace model may not be necessary for this particular task. Furthermore, the improvement observed when using normalized faces highlights the importance of proper image preprocessing in computer vision tasks. Future work could explore other preprocessing techniques and model architectures to further improve performance.

6 Discussion

In this study, we proposed a method for age detection that leverages both Convolutional Neural Networks (CNN) and a pre-trained VGGFace model.

6.1 Suitability of Deep Neural Networks

Deep Neural Networks, specifically CNNs, are particularly suitable for this task due to their ability to automatically and adaptively learn spatial hierarchies of features from the input images. CNNs have been widely used in image classification tasks and have shown excellent performance in extracting complex features from images.

The VGGFace model, based on the VGG16 architecture, is a pre-trained model specifically designed for face recognition tasks. By leveraging this model, we were able to extract high-level facial features without the need for extensive computational resources and time required to train such a model from scratch.

6.2 Benefits of the Proposed Model

Our proposed model combines the strengths of CNNs and the VGGFace model. The CNN part of our model is capable of learning age-related features from face images, while the VGGFace model allows us to leverage pre-existing knowledge about facial feature extraction.

Furthermore, our model also incorporates tabular data, which includes additional features such as texture features, wrinkle features, and eye openness. This multi-modal approach allows our model to consider a wider range of information than models that rely on image data alone.

An interesting finding from our experiments was the significant improvement in model performance when using normalized faces compared to the original images. This highlights the importance of proper image preprocessing in computer vision tasks.

However, our study is not without limitations. The uneven distribution of age groups in our dataset could potentially influence our model's performance, as it might be less accurate in predicting ages that are underrepresented in the training data. Future work could explore strategies to address this issue, such as oversampling the underrepresented age groups or applying class weighting during model training.

7 References

1. Chollet, F. et al. (2015). Keras. [online] Available at: `https://github.com/keras-team/keras` [Accessed Day Mo. Year].
2. Abadi, M. et al. (2015). TensorFlow. [online] Available at: `https://github.com/tensorflow/tensorflow` [Accessed Day Mo. Year].

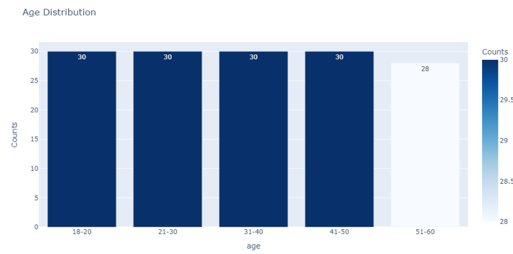

Figure 1: Age Distribution

Image	Mean	Standard Deviation
0	0.53	0.36
1	0.46	0.25
2	0.56	0.27
⋮	⋮	⋮
147	0.67	0.24

Table 1: Mean and standard deviation of features for each image.

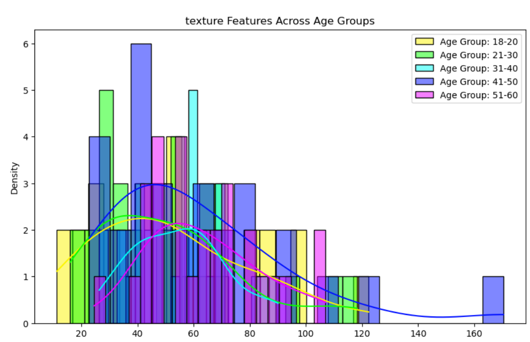

Figure 2: Texture Features Across Age Groups

Model	MAE	MSE	F1 Score
CNN	1.220	2.036	0.868
VGGFace	1.289	2.231	0.868

Table 2: Performance of the CNN and VGGFace models.