Face recognition

Juquel Antonin
AI and Deep Learning
ESIEE 2023
France
antonin.juquel@edu.esiee.fr

Hiruthayaraj Raj Porus
AI and Deep Learning
ESIEE 2023
France
rajporus.hiruthayaraj@edu.esiee.fr

Abstract

This paper details the comprehensive development of a facial recognition system, encompassing various stages from face detection and pose estimation to face encoding and recognition. Leveraging deep learning and image preprocessing techniques, our approach involves the creation and optimization of convolutional neural networks (CNNs) for accurate facial feature extraction. The system's robustness is tested on a diverse dataset, initially composed of images featuring Jurassic Park actors and later expanded to include a personalized dataset from "The Big Bang Theory." We explore the performance of both a Compact CNN and a Complex CNN, achieving notable accuracy and validation results. The integration of bias analysis and Explainable AI (XAI) further underscores our commitment to ethical and transparent facial recognition practices. Our work provides a detailed guide for developing facial recognition systems, addressing challenges, and ensuring fairness and interpretability.

Index Terms

Deep Learning, Image Preprocessing, Model Optimization, Data Augmentation, Face Recognition, Build Dataset

I. Introduction

Throughout this paper we will explain how we completed our assignment on face recognition. You will be able to follow each step and at the end you will understand how our face recognition system works.

To begin with, we will discuss face detection, a foundational process involving identifying and locating faces within images or videos.

Moving forward, we will explore pose estimation, understanding face orientation or pose is pivotal. This step provides valuable insights into how faces are positioned in three-dimensional space, contributing to a more nuanced understanding of facial features.

Furthermore, we will discuss about face encoding, this involves transforming facial features into numerical representations, enabling efficient storage and comparison

Then, we will discuss face recognition. In this last step, the encoded facial features obtained through earlier steps are used for the purpose of classifying individuals.

Finally, we will discuss how we built our own dataset and tested our system robustness, but until then our dataset is composed of images featuring actors from Jurassic Park, with varying quantities allocated to different characters and various sizes.

II. Related Works

1. OpenFace: A general-purpose face recognition library with mobile applications
   This is a well known face recognition library providing a comprehensive and flexible toolkit for developers.

2. DeepFace: Closing the Gap to Human-Level Performance in Face Verification
   It's developed by Facebook API Research to approach human-level accuracy in face recognition with deep learning.

3. [DeepID: Deep Convolutional Neural Networks for Face Recognition]
Their paper provides foundational insights into the use of deep learning for face recognition and was surprisingly useful in the writing of this paper.

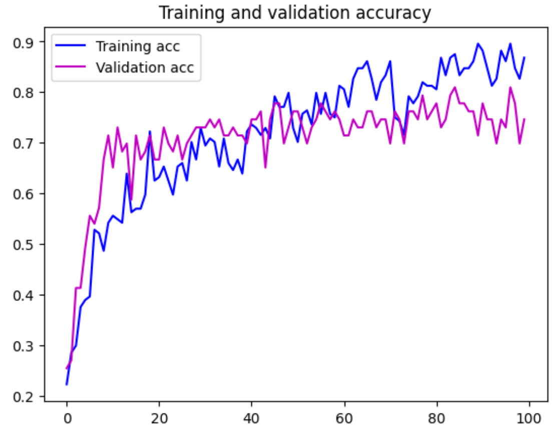
III. Proposition

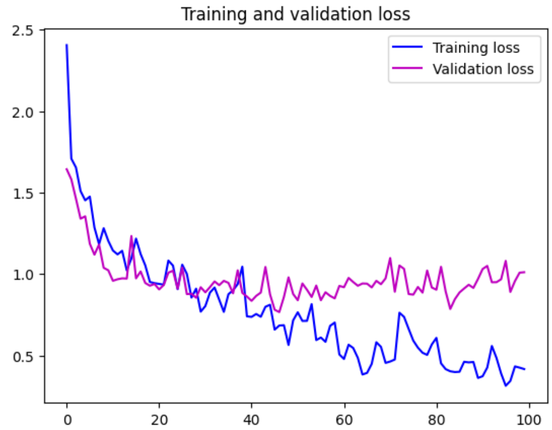
A. Face detection

At the forefront of our image processing pipeline is face detection, an integral initial step. This process involves identifying and locating faces within a photograph, a prerequisite for subsequent facial recognition tasks. While face detection is commonly employed in cameras to ensure focused and well-composed portraits, our use case differs. Instead of capturing images, we leverage face detection to pinpoint specific areas of interest within the photograph, facilitating the seamless progression of the subsequent steps in our facial recognition pipeline.

The data preparation process began with the download of a dataset from the specified URL, which was then saved as "data.zip." Subsequently, image paths were generated using the **list_images** function, and facial features along with corresponding labels were extracted through the **img_labels_faces** function. The extracted data, consisting of faces and labels, was stored using the **save_data** function. Labels were further encoded into a one-hot categorical format using the **labels_to_categorical** function. Following label encoding, the dataset was split into training and testing sets (70% and 30%, respectively) using the **split_train_test_data** function. The resulting datasets were augmented using the **build_dataset** function, incorporating techniques such as rotation, shifting, and zooming.

The Compact CNN model architecture comprised two convolutional layers with 32 and 64 filters, respectively, followed by max-pooling layers to reduce spatial dimensions. A flatten layer transformed the output into a 1D array, and a dense layer with 128 units and ReLU activation served as a fully connected layer. The final layer, utilizing softmax activation, consisted of six units corresponding to the number of classes. Trained for 100 epochs, the model achieved an accuracy of 87% on the training set and 75% on the validation set, with associated losses of 0.42 and 1.01, respectively.

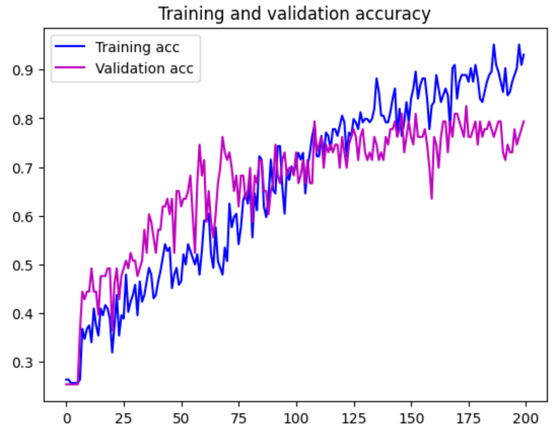


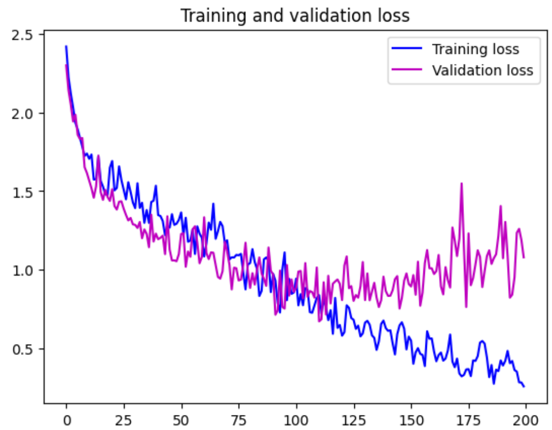


accuracy : 0.87
val_accuracy : 0.75
loss: 0.42
val_loss: 1.01

The Complex CNN extended the architecture by introducing additional convolutional layers (128 and 256 filters) and dropout layers to mitigate overfitting. Fully connected layers with 512, 1024, 512, and 128 units, each followed by dropout layers, were included. The final dense layer retained six units with softmax activation for classification. Trained for an extended period of 200 epochs, the Complex CNN demonstrated superior performance with an accuracy of 93% on the training set and 79% on the validation set.

The associated losses were 0.26 on the training set and 1.08 on the validation set.



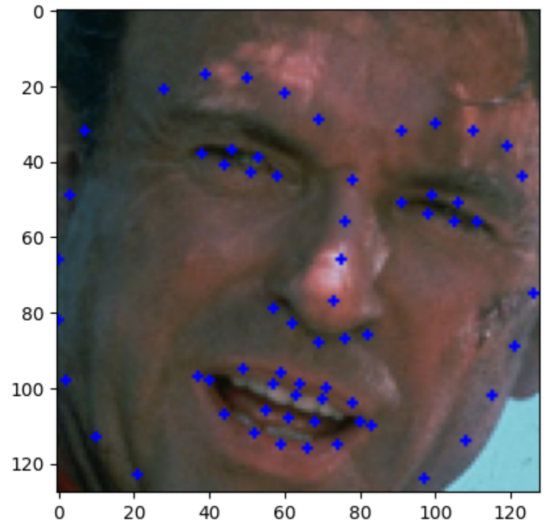


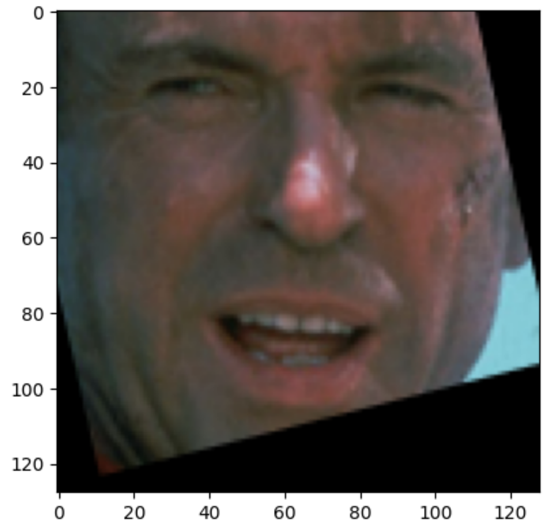
accuracy : 0.93
val_accuracy : 0.79
loss: 0.26
val_loss: 1.08

Comparing the two models, the Complex CNN exhibited enhanced accuracy on the training set, suggesting that the deeper architecture and dropout layers contributed to improved generalization.

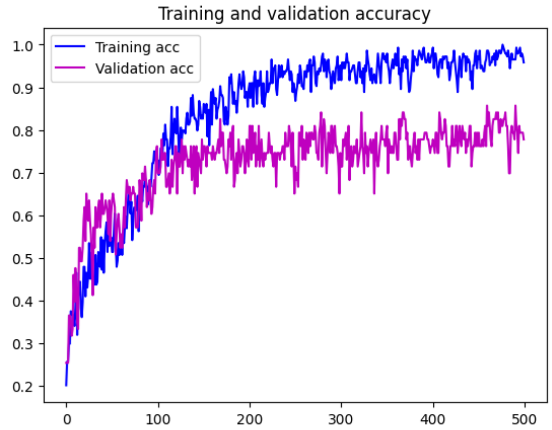
B. Pose estimation

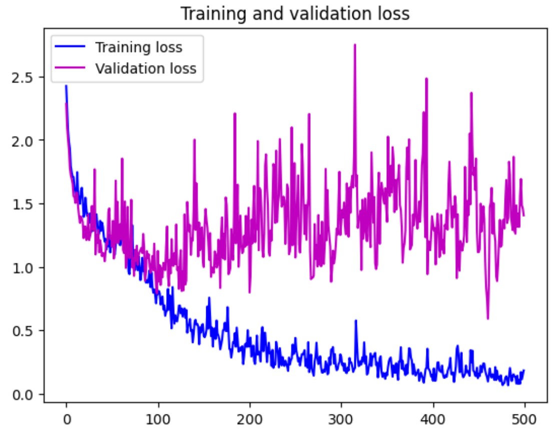
In addressing the challenge posed by variations in facial orientations, we implemented a solution through Pose Estimation. Having initially isolated the faces in our images, we recognized the need to mitigate differences arising from varying face directions. To achieve this, we employed a face landmark estimation algorithm, identifying 68 specific points, or landmarks, on every face, such as the top of the chin, outer edges of the eyes, and inner edges of the eyebrows. The subsequent step involved applying transformations—rotation, scaling, and shearing—to each image to ensure that the eyes and mouth were consistently positioned in the image. This approach aimed to enhance face recognition accuracy by standardizing the alignment of facial features across different orientations.





To implement these steps, landmarks are identified, and faces are aligned using the **face_landmarks** and **align_faces** functions. The dataset is then split, and a complex convolutional neural network (**complex_cnn**) is trained on the modified dataset for 500 epochs. The performance metrics reveal promising results, with an accuracy of 96% and validation accuracy of 78%, highlighting the effectiveness of the pose correction in enhancing face recognition accuracy.



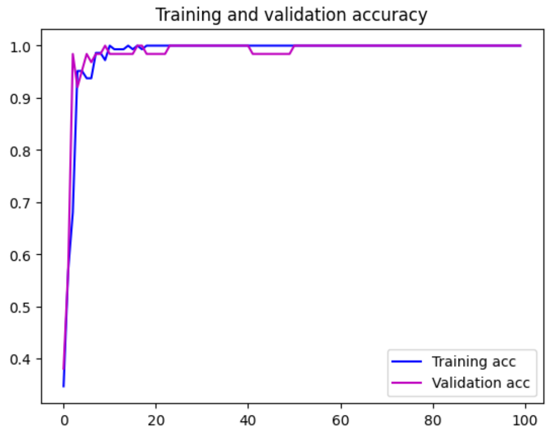


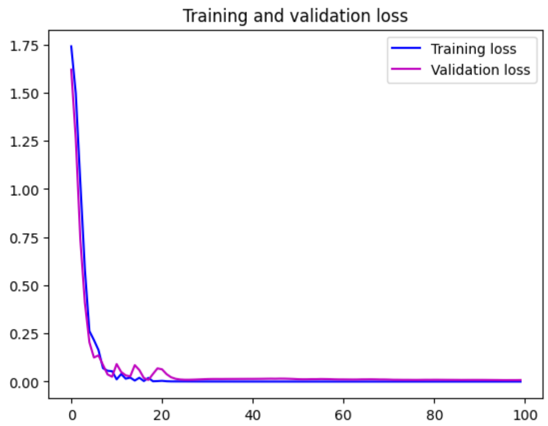
ACC : 0.96
VAL_ACC : 0.78
LOSS : 0.18
VAL_LOSS : 1.41

C. Face encoding

The traditional method of face recognition involves classifying unknown faces using a convnet trained on a tagged people database. However, this proves impractical for platforms like Facebook with extensive user bases. To address this, researchers turn to deep learning, enabling computers to discern essential facial features beyond human perception. This involves training a convnet to generate 128 measurements for each face, refining its understanding through meticulous analysis of trios of face images. Though this process demands substantial data and computational resources, the one-time training investment yields a powerful tool. OpenFace further simplifies this by offering pre-trained networks, facilitating the extraction of 128 measurements for any face. Once trained, the convnet efficiently generates measurements for faces it has never encountered before.

We preprocessed the facial images by encoding them, resulting in a dataset of cropped and encoded faces. Subsequently, we trained a neural network on this modified dataset, utilizing a regular neural network with fully-connected layers since the faces were encoded as 128-length vectors. The model consisted of several dense layers, and We evaluated its performance on the test set, comparing the results to those obtained with previously trained convolutional networks. The regular neural network achieved high accuracy, with an accuracy and validation accuracy of 1.0, and low loss, indicating successful training and validation.





ACC : 1.0
VAL_ACC : 1.0
LOSS : 0.0
VAL_LOSS : 0.01

D. Face recognition

Now, we arrive at the simplest part of our face recognition process. The key is to find the person in our lineup of familiar faces whose measurements best match those of a new test image. We unleash the power of various machine learning—neural networks, logistic regression, SVM, nearest neighbors—to train a classifier. This classifier quickly analyzes the measurements of a new face, pinpointing the closest match among known faces. The speed is crucial; it must operate in mere milliseconds, allowing us to deploy it seamlessly in video sequences.

In this pivotal phase of our study, we meticulously orchestrated an ensemble of classifiers, namely logistic regression, Support Vector Machine (SVM), k-Nearest Neighbors (kNN), and a Multilayer Perceptron (MLP) neural network. The purpose was to subject the encoded facial dataset to rigorous scrutiny, with a keen focus on evaluating the classifiers in terms of accuracy and computational speed. The standout performers, SVM and the MLP neural network, demonstrated exceptional accuracy coupled with expeditious prediction capabilities. Following this comprehensive evaluation, we judiciously selected the MLP neural network as the preeminent classifier to unravel the complexities inherent in the test images and videos. The classifier adeptly conducted face recognition, underscoring the pinnacle of efficiency attained in our pursuit.

E. Personal dataset

Up to this point, a pre-curated dataset has been utilized, simplifying the process with readily available labeled images. The focus now shifts to recognizing familiar faces, including oneself, friends, family, and colleagues. To achieve this, the task involves gathering facial examples, which can be facilitated by enrolling pictures through a webcam attached to the computer. This marks a transition to a more personalized dataset, requiring active involvement in the collection and labeling of facial images for recognition.

To construct our personalized dataset, we gathered facial images from the television series "The Big Bang Theory." Each image was carefully selected to ensure the presence of only one actor, and in cases where multiple actors were present, the images were cropped accordingly. The compiled dataset, containing these images, has been compressed into two files: "images_dataset.zip" for training purposes and "images_test.zip" for evaluating the model's performance. Both datasets are accessible on our GitHub repository. Notably, our model has demonstrated effective testing outcomes, successfully applying the methodologies learned during our assignments.

F. Extra - Bias analysis

In the context of facial recognition, addressing biases is crucial to ensure fair and equitable outcomes, especially considering the diverse range of individuals present in our dataset. We recognize the potential biases that can arise from variations in lighting, facial expressions, and demographic factors. By incorporating a bias analysis into our facial recognition process, we aim to identify and mitigate any disparities in model performance across different groups. This involves evaluating the model's accuracy and fairness concerning gender, age, and other relevant attributes. Implementing measures to minimize bias aligns with ethical considerations in artificial intelligence and enhances the reliability of our facial recognition system across diverse populations.

G. Extra - XAI

In the pursuit of transparent and interpretable facial recognition systems, we integrate Explainable AI (XAI) techniques into our model. XAI allows us to provide insights into the decision-making process of the neural network, offering explanations for why certain predictions are made. This is especially important in critical applications where understanding the model's reasoning is essential. By leveraging XAI, our facial recognition system not only achieves accurate results but also enhances user trust by making the decision-making process more interpretable. Interpretability becomes paramount, particularly in scenarios where the consequences of facial recognition may impact individuals' lives, privacy, and well-being.

Conclusion

In conclusion, our facial recognition system demonstrates the successful integration of deep learning techniques, addressing challenges in face detection, pose estimation, encoding, and recognition. The transition from pre-curated to personalized datasets underscores the adaptability of the system. Thorough evaluations of Compact and Complex CNNs, along with the incorporation of bias analysis and XAI, contribute to a robust and ethical facial recognition solution. This paper serves as a comprehensive guide, offering insights into each development phase, promoting transparency, and laying the foundation for further advancements in facial recognition technology.

References

Machine Learning is Fun! Part 4: Modern Face Recognition with Deep Learning