Project report

Advanced Machine Learning

Martim Silva 51304 and Alexandre Sobreira 59451

2022/2023

Contents

[Project Structure 3](#_Toc134816686)

[Introduction – Description of the problem 4](#_Toc134816687)

[Methodology 4](#_Toc134816688)

[Images Extraction 5](#_Toc134816689)

[Categories definition 5](#_Toc134816690)

[Feature Extractors 6](#_Toc134816691)

[HOG 6](#_Toc134816692)

[ORB 7](#_Toc134816693)

[VGG16 8](#_Toc134816694)

[AUTOENCODER 8](#_Toc134816695)

[Data Balancing 8](#_Toc134816696)

[Variables creation 9](#_Toc134816697)

[Models 9](#_Toc134816698)

[Balanced vs. Unbalanced 10](#_Toc134816699)

[Best Models 10](#_Toc134816700)

[Optimized Models 10](#_Toc134816701)

[**Results – Presentation and critical analysis of the results obtained** 10](#_Toc134816702)

[**Unbalanced vs Balanced and Best Model / Feature Extractor Selection** 10](#_Toc134816703)

[**Unbalanced Dataset Results** 11](#_Toc134816704)

[**Balanced Dataset Results** 11](#_Toc134816705)

[**Unbalanced vs Balanced** 12](#_Toc134816706)

[Feature Extractor 12](#_Toc134816707)

[Best Models 12](#_Toc134816708)

[Given the previous results, the Best models considered for further optimization can be seen in Figure 3. 12](#_Toc134816709)

[Optimized Models 13](#_Toc134816710)

[Final comments – Global assessment of the project: usefulness; what has been shown; limitations; possible improvements or extensions 15](#_Toc134816711)

[Bibliography 17](#_Toc134816712)

[Annexes 18](#_Toc134816713)

# Project Structure

Max: 6 pages, font 10 (not including cover page and annexes)

1. Introduction – Description of the problem
2. Approach – Description of the solution
3. Implementation – Detailed description of the developed prototype
4. Results – Presentation and critical analysis of the results obtained
5. Final comments – Global assessment of the project: usefulness; what has been shown; limitations; possible improvements or extensions
6. Bibliography

# 

# Introduction – Description of the problem

Image classification is a fundamental task in computer vision with numerous applications, such as object detection, face recognition, and content-based image retrieval. With the advent of deep learning, convolutional neural networks (CNNs) have become the most widely used approach for image classification due to their ability to automatically learn meaningful features from raw data. CNNs have achieved state-of-the-art performance on many benchmark datasets such as ImageNet, CIFAR-10, and MNIST (Krizhevsky et al., 2012; Russakovsky et al., 2015; LeCun et al., 1998). However, traditional machine learning algorithms such as Support Vector Machines (SVMs) are still commonly used for image classification tasks due to their simplicity, interpretability, and ability to handle high-dimensional data. SVMs have been used for various image classification tasks, such as face recognition (Zhang et al., 2016), object recognition (Zheng et al., 2018), and medical image analysis (Jiang et al., 2021), with promising results.

In this project, we will use both SVMs and a custom CNN architecture for image classification of human faces into different gender and age classes. Unlike previous studies that used pre-trained CNN architectures such as VGG, ResNet, and Inception (Simonyan and Zisserman, 2015; He et al., 2016; Szegedy et al., 2015), we will develop our own CNN from scratch, which allows us to customize the architecture to the specific characteristics of the UTK-FACE dataset. The UTK-FACE dataset is a challenging and diverse dataset, containing images of various resolutions, poses, and facial expressions, which makes it a suitable benchmark for evaluating the performance of our models.

The core problem of this project is to classify images of human faces into different gender and age classes. We will use the UTK-FACE dataset, which contains over 20,000 face images of varying resolutions, poses, facial expressions, illuminations, and occlusions. The dataset also includes ethnicity information, which will not be used in this project.

In the original work by Hiremath et al. (2012), the authors used an algorithm called SURF for feature extraction. However, this algorithm requires permission from its creators to be used, which is not feasible for this project. Therefore, we will use several feature extraction methods and compare their performance to choose the best one for our model. The feature extraction methods will be described in the implementation section.

The UTK-FACE dataset will be used in this project, which includes more than 20,000 face images of size 200 by 200 pixels. Each image is labelled with gender (0 for male, 1 for female) and age (from 0 to 116 years old) information, which will be used for classification. The dataset is publicly available on Kaggle and can be accessed through the following link: https://www.kaggle.com/datasets/jangedoo/utkface-new?datasetId=44109.

In summary, this project aims to replicate the work of Hiremath et al. (2012) on gender classification using SVM and extend it to age classification. Additionally, we will compare SVM's performance with CNN for both gender and age classification. The project will also explore various feature extraction methods, replacing SURF with a suitable alternative. The UTK-FACE dataset will be used, which is a large and diverse dataset of face images.

# Methodology

All the code was done using Python3, and the models were ran using

## Images Extraction

The dataset was downloaded to Google Collab using the Kaggle API to retrieve it directly from the source in the Kaggle website. The image files were read and transformed using OpenCV module functions in a loop where the 3 channels RGB colour format and 200 by 200-pixel dimensions were not tampered with to not lose any information that could be helpful for classification by the models later. The target classes (Age and Gender) of each image through their presence in the corresponding file name were stored alongside the images themselves which were processed by having each pixel’s 0 to 255 values normalized so to not cause conflicts with the requirements of all models used. This normalization also ensured that in the training of the models no single value would overtake the rest in terms of importance and result in an unequal optimization of the adjusting of weights (for the neural networks) mostly because of a few “standout” points which lowers the degree of overfitting in the fitting process.

## Categories definition

For the goals that are producing the best models for the classification of gender and of age some considerations must be defined. For a nominal variable like gender the model must produce a prediction between one of two classes: Male or Female. While for age it’s important to remember that the file names of the images are in themselves descriptions of target variables, for example the following image titled “26\_0\_2\_20170116182630634” is of a 26 year old male (second token where 0 is male and 1 is female) of Asian ethnicity (third token with 0 for white, 1 for black, 2 for Asian, 3 for Indian and 4 for any other ethnicities) with the ethnicity not being considered for the scope of this project.



Fig. X - Instance in the UTKFace data set with file title “26\_0\_2\_20170116182630634”

As Age is a numerical variable, a decision was made of keeping the model to predict on Age to be one of classification and not regression, this was supported by the notion that people’s facial features do not change at an even rate across their lives, at stages like growing from a newborn baby to an infant or when entering adolescence there are many more changes to the body, and more importantly in our context, to the set of features that compose one’s face, as such to classify Age the target classes were first decided according to guidelines provided in the US National Institute of Health (REF) as no longer represented as a numerical variable but as a categorical variable with all of the recorded age values (0 to 116) slotted into bins of classes, this separation is detailed in the table below.

|  |  |
| --- | --- |
| Age Classes | Intervals of values comprised |
| Baby | [0,1] |
| Infant | [2,13] |
| Adolescent | [14,16] |
| Young Adult | [17,29] |
| Middle-aged Adult | [30,59] |
| Old Adult | [60,116] |

Fig. X - Table with classes considered for predictions of Age target variable

* For the SVM normal class label were maintained for the age (0-5)
* For the CNN's these classes were transformed into a one hot encoding using to\_categorical from keras.
* The reason:
  + When using a CNN for multiclass classification, the output layer of the network typically has one neuron per class, and the activation function used is often softmax. Softmax activation ensures that the output for each class is between 0 and 1, and that the sum of the outputs for all classes is equal to 1. This makes it easy to interpret the output as probabilities for each class. In order to train the network to predict the correct class probabilities, we use a loss function such as categorical cross-entropy. In order to use this loss function, we need to convert the target labels into a one-hot encoded format, where each label is represented as a binary vector with a 1 in the index corresponding to the class and 0s elsewhere. The `to\_categorical` function in Keras provides a convenient way to perform this conversion. In contrast, SVMs do not require one-hot encoding of target labels because the optimization problem that SVMs solve involves maximizing the margin between classes rather than directly predicting class probabilities. SVMs can be used with integer-encoded labels directly, without the need for one-hot encoding. However, some implementations of SVMs, such as scikit-learn, do offer an option to use one-hot encoded labels.

Alongside a “control data set” where the data was used as provided (we refer to this as the Base data set) , four different feature extraction methods were also applied in an attempt to shorten the amount of data used for the models while losing an acceptable amount of original information that would not contribute as much towards making predictions.

## Feature Extractors

### HOG

The first method, Histogram of Oriented Gradients, functions by resizing the image then calculating the gradient or the directional change in the intensity/colour of the image in sub-blocks of very few pixels each that compose the entire image, this processes involves calculating the matrices of magnitude and of angle for each of these blocks and with these parameters recorded, a feature vector is constructed from the matrices which is the histogram of oriented gradients of each sub-block. The values in these histograms are then normalised and the image is reconstructed with the values of each histogram, the result of applying this method is one that can be visible, with an example provided below.

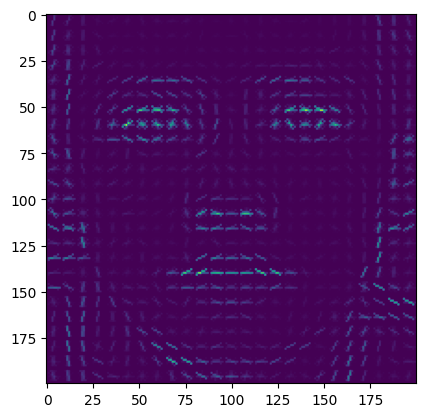
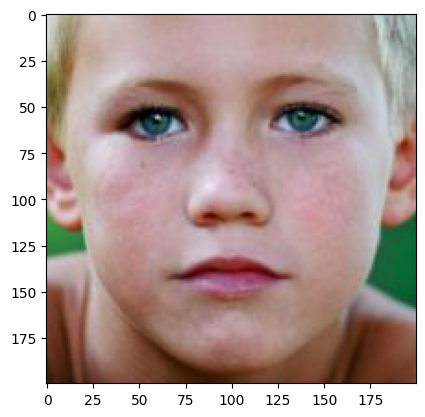


Fig. X - Instance of UTKFace data set before and after application of HOG extraction method

VGG16 is a pre-trained (more than a million images from *ImageNet* database) convolutional neural network with 16 layers

### ORB

The Oriented FAST and Rotated BRIEF (ORB) feature extraction method is a fast and efficient alternative to other popular feature extraction methods, such as Scale-Invariant Feature Transform (SIFT) or Speeded Up Robust Features (SURF) (Rublee et al., 2011). ORB uses a combination of the FAST keypoint detector and the BRIEF descriptor, which is binary and therefore requires less memory than other descriptor methods.

ORB works by first detecting keypoints using the FAST algorithm, which identifies points of interest in an image based on changes in intensity. Once keypoints have been detected, ORB uses the BRIEF descriptor to extract a binary feature vector that describes the local patch of the image surrounding the keypoint. The BRIEF descriptor is designed to be fast and efficient, and it uses a random pattern of tests to compare pairs of pixels within the local patch and produce a binary code.

One of the main advantages of ORB is its robustness to changes in scale and rotation. ORB achieves rotation invariance by first calculating the orientation of each keypoint using a method called centroid orientation, which computes the average gradient direction of the pixels within the keypoint region. The BRIEF descriptor is then rotated to align with the keypoint orientation, allowing it to capture the same information regardless of the image's rotation.

Overall, ORB is an efficient and effective feature extraction method that has become a popular alternative to other methods, such as SIFT and SURF.

#### ORB Adaptation:

During the process of using the output of a keypoint detection algorithm as inputs for models, a problem occurred because the algorithm did not detect the same keypoints in all images, which resulted in differences in the shapes of the images. This created conflicts, particularly for the SVM model.

To address this issue, several adaptations were made to the process. First, a minimum number of keypoints was defined, which was set at 100 after a superficial analysis suggested that this value provided the most images. Then, all images containing that number of keypoints were selected. Next, keypoints were randomly removed until the number of keypoints in each image was reduced to 100. The resulting images, along with their corresponding labels, were then appended to a new variable to maintain the correspondence between the images and their labels.

It should be noted, however, that this process may lead to significant losses in the quality of the models. This is because the data will be reduced greatly, as well as the information from the faces, which may affect the ability of the models to accurately classify new, unseen data.

### VGG16

VGG-16 or VGGNet (Simonyan et al.,2014) is a pre-built convolutional neural network model that uses 16 deep layers with weights that are pre-trained with the *imageNet* dataset, having obtained a test accuracy of 92.7% on predictions of that data set which is composed of over a million images.

It is an improvement on another similar model AlexNet by using smaller sized filters (3 by 3) which contributes to reducing the chance of the model overfitting and it is very used for object detection. It originally is designed to work on images that are 224 by 224 pixels but it can work on different resolutions and so it was adapted to take in UTKFace data set images which are 200 by 200 pixels without having any need to change the colour channels. It is a very large network with millions of parameters, yet its components are only input, convolutional, hidden, pooling layers and output layers with Rectified Linear Unit activation functions within and softmax activation for the output layer. For our project this CNN was used as a feature extractor where the images of the data set used as the input and the output are high-level features of said images that are passed on to the models that will enact the gender and age classification tasks with the knowledge that VGG-16 generated earlier.

### AUTOENCODER

The final method of feature extraction used on the data set of this project is the application of the “first half” of an autoencoder. An autoencoder is a neural network very much employed in reproducing data such as images or text. It is composed of three parts, the encoder that includes layers with filters decreasing in filter count (bottlenecking) that reduce or compress the information coming in through several convolutions and then flattens the “reduced image” (the representation of the information in smaller dimension space) into a feature vector called the code. The second part takes this vector with the general properties of the training data as input and generates the image again based on those properties and this part is named the decoder. Feature extraction was performed by using the encoder part to only supply classification models with the data compressed in the previously mentioned “code”.

There is an issue with sampling of the data where we were not able to generally evenly extract values from all classes in respect to the age classification.

## Data Balancing

Our analysis of the distribution of the data regarding each of the target classes revealed that Male and Female instances were present in differing quantities, an even greater degree of disparity was observed regarding the ages of each image which had the highest densities around the young adults (17 to 29 years old) where the count was almost three times higher than in all the remaining age groups. The difference for genders was smaller at a X%.

The reason for this unbalance in the categories of ages was due to the way the intervals were decided, initially according to the guidelines proposed by the American Medical Associations’ age designations [1], but with an extra degree of separation in the Adults interval in an attempt to lessen the effects of having such massive differences in the number of images comprised in each interval of ages where Adolescents or teenagers span just 5 years, and Adults contain images of all people with ages between 18 and 65 years old (spans 47 years instead).

* After this sub-task of categorization of the continuous variable of “Age”, the classes were visualized again, and we observed that the Adolescent class had only 581 elements being the one with the smallest size so we decided to retrieve this smallest quantity of elements from all other classes in a random fashion so as to have a balanced version of the data to study alongside the unbalanced one. For this random retrieval a preliminary shuffling of elements was performed since the data was originally sorted in relation to age. The resulting data set had 581 elements for all six classes for a total of 2900 images for the classification models to be trained and tested with. For genders the same procedure was undertaken with each class having the length equal to the length of the smallest class at a total of 20000 images in equal amounts of Male and Female representations. The effects of balancing the dataset (balanced vs unbalanced) will be evaluated using statistical hypothesis tests after the models are applied.

A picture containing screenshot, plot, line, diagram

Description automatically generatedFig. X – Bar charts respectively detailing the distribution of images per age values numerically, categorically and of genders

## Variables creation

* From this, the variables of interest for the models were created and saved using the numpy.savez so this heavy computationally process didn't need to be repeated, only loading of the variables would be required.
  + The variables were: X\_age\_classes(array) (shape) and y\_age\_classes (array)(lenght); X\_genders (array)(shape) and y\_genders(array)(lenght)

## Models

* Both svm’s and CNN's were ran twice, one for balanced and one for unbalanced dataset. Also, the best svm and best CNN will be ran using the best performing feature extractor for both gender and age classes
* The metrics generated by the models were carefully chosen. The Matheus correlation coefficient was used as the main indicator of the model's quality given the unbalanced datasets were present. The classification report from sklearn metrics was also used and given that it is a classification problem, the confusion matrices were outputted as well to see the distribution of true vs predicted labels.
* On a first phase, where the goal was to find the best feature extractor, for the gender and ages only 10000 images were used (unbalanced dataset) and for the balanced dataset, 10000 images for genders and full images for age classes, 2900.

### Balanced vs. Unbalanced

* In order for us to have a more precise notion of the impact of the balancing of the classes a Wilcoxon statistical test was performed to evaluate if the MCC obtained from the various models using the balanced dataset was statistically higher than the MCC obtained from the various models using the unbalanced dataset, the same was done for the ACC, for both age and gender.

### Best Models

#### Gender

* According to best metrics…

#### Age

* According to best metrics…

### Optimized Models

After selecting the best models for both gender and age classes some otimizations will be performed on both SVM and CNN.

#### SVM

Gender – 20k

Age – 2.9k

#### CNN

Regarding the optimization of the CNN, it will be improved by using the full dataset for both gender (20,000) and age classes (2,900) as well as implementing the following adaptations: (1) Increase the number of convolutional layers from 3 to 6; (2) Add batch normalization after each layer; (3) Change the number of neurons in the fully connected layer to 512; (4) Add an extra fully connected layer with 256 neurons. The summary of the network can be seen in Figure X in annexes.

Besides these alterations, other techniques will be applied: (1) Early stopping, which is used to stop the model training if the validation loss (i.e., the loss on a held-out validation set) does not improve for a certain number of epochs (10); (2) Reduce Learning Rate Palteau to reduce the learning rate of the optimizer if the validation loss does not improve for a certain number of epochs (3), the learning rate is reduced by a factor (0.2) until it reaches a minimum value (0.0001); (3) Testing of 3 different optimization algorithms (Adam, SGD and RMSprop).

# **Results – Presentation and critical analysis of the results obtained**

After all the method was applied the following results were obtained. Below can be seen the results for both gender and age classes obtained by the SVM and CNN on the various steps of the methodology.

## **Unbalanced vs Balanced and Best Model / Feature Extractor Selection**

For this phase, given that the goal was to observe which combination model-feature extractor provided the best results for classifying both gender and age classes, only half of the data set was used (n = 10000).

## **Unbalanced Dataset Results**

In the first phase of the methodology, all feature extractors were tested using SVM and CNN to obtain the best combination regarding the unbalanced dataset, the results obtained from the unbalanced dataset can be seen below in Table 1 where they were sorted by MCC values given the unbalanced nature of the datasets.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Model** | **Feature Extract** | **ACC** | **MCC** |  | **Model** | **Feature Extract** | **ACC** | **MCC** |
| SVM | VGG 16 | 0.89 | 0.78 |  | CNN | BASE | 0.66 | 0.51 |
| CNN | BASE | 0.89 | 0.78 |  | SVM | VGG 16 | 0.54 | 0.42 |
| SVM | HOG | 0.88 | 0.75 |  | CNN | Autoenconder | 0.59 | 0.42 |
| CNN | VGG 16 | 0.87 | 0.75 |  | SVM | BASE | 0.7 | 0.41 |
| SVM | BASE | 0.87 | 0.73 |  | SVM | HOG | 0.52 | 0.4 |
| SVM | Autoenconder | 0.86 | 0.73 |  | CNN | HOG | 0.56 | 0.39 |
| CNN | Autoenconder | 0.84 | 0.68 |  | CNN | ORB | 0.37 | 0.09 |
| CNN | HOG | 0.82 | 0.63 |  | SVM | ORB | 0.12 | 0.08 |
| CNN | ORB | 0.22 | 0.61 |  | SVM | Autoenconder | 0.21 | 0.02 |
| SVM | ORB | 0.66 | 0.32 |  | CNN | VGG 16 | 0.27 | 0.01 |

Table 1 Accuracy (ACC) and Mathews Correlation Coefficient (MCC) regarding the Unbalanced Classes Dataset for Genders (left) Age Classes (right) and using the various feature extractors with SVM and CNN.

Looking at Table 1 is possible to conclude that when using the unbalanced datasets, for the classification of Age Classes the best combination is CNN with no feature extractor (ACC = 0.66; MCC = 0.51) and for the classification of Gender the best combination is SVM using VGG16 as feature extractor (ACC = 0.89; MCC = 0.78)

## **Balanced Dataset Results**

In the second phase of the methodology the previous process was repeated but now using the balanced dataset which meant that for the age classes instead a n = 10000 it was n = 2900, the results obtained from the balanced dataset can be seen below in Table 2 where they were sorted by MCC values to compare with the models obtained with the unbalanced datasets.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Model** | **Feature Extract** | **ACC** | **MCC** |  | **Model** | **Feature Extract** | **ACC** | **MCC** |
| SVM | HOG | 0.9 | 0.8 |  | SVM | HOG | 0.63 | 0.56 |
| CNN | BASE | 0.89 | 0.79 |  | CNN | BASE | 0.6 | 0.51 |
| CNN | VGG 16 | 0.88 | 0.76 |  | SVM | VGG 16 | 0.58 | 0.51 |
| SVM | Autoenconder | 0.88 | 0.76 |  | SVM | Autoenconder | 0.56 | 0.47 |
| SVM | BASE | 0.87 | 0.73 |  | SVM | BASE | 0.55 | 0.46 |
| SVM | VGG 16 | 0.86 | 0.73 |  | CNN | HOG | 0.52 | 0.43 |
| CNN | HOG | 0.85 | 0.71 |  | CNN | Autoenconder | 0.52 | 0.41 |
| CNN | Autoenconder | 0.84 | 0.68 |  | SVM | ORB | 0.3 | 0.15 |
| SVM | ORB | 0.67 | 0.35 |  | CNN | ORB | 0.28 | 0.13 |
| CNN | ORB | 0.62 | 0.23 |  | CNN | VGG 16 | 0.19 | 0.02 |

Table 2 Accuracy (ACC) and Mathews Correlation Coefficient (MCC) regarding the Balanced Classes Dataset for Genders (left) and Age Classes (right) using the various feature extractors with SVM and CNN.

Looking at Table 2 is possible to conclude that when using the balanced datasets, for the classification of Genders the best combination is SVM with the HOG feature extractor (ACC = 0.9; MCC = 0.8) and for the classification of Age Classes the best combination is also SVM using the HOG feature extractor (ACC = 0.63; MCC = 0.56)

## **Unbalanced vs Balanced**

To have a more precise understanding of the impact of unbalanced datasets 4 statistical tests were performed with the following hypotheses: H0: Balanced “Class” “Metric” = Unbalanced “Class” “Metric” vs. H1: Balanced “Class” “Metric” MCC > Unbalanced “Class” “Metric” MCC. This hypothesis was repeated four times for “Class” Age and Gender and “Metric” ACC and MCC. The statistical test applies was the non-parametric Wilcoxon given the low sample size (n<30) and the absence of Gaussian distribution of ACC and MCC in both samples (Balanced and Unbalanced).

The results indicate that for age, where the biggest transformation was performed, the MCC obtained from the various models using the balanced dataset was statistically higher than the MCC obtained from the various models using the unbalanced dataset (alpha of 0.01). For the gender, the same pattern was observed but it was not significant for an alpha of 0.05. For the ACC for both gender and age, it showed to be higher in the balanced dataset, but it wasn't statistically significant in any case for an alpha of 0.05.

The fact that only MCC for Age Classes was significant is rather interesting and logical given that first, the MCC is much more sensitive to dataset unbalances than ACC and second because the biggest transformation was performed on the Age Classes, therefore, impacting the MCC and revelling statistically significant differences with a low p-value (>0.001).

Given these results, the models obtained from the balanced dataset were considered for the next steps which were to choose the best models for both gender and age classes.

## Feature Extractor

Discuss Feature Extractor Performance: Orb adaptation effect and other visible behaviours

## Best Models

## Given the previous results, the best models considered for further optimization can be seen in Figure 3.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Model** | **Feature Extract** | **ACC** | **MCC** |  | **Model** | **Feature Extract** | **ACC** | **MCC** |
| SVM | HOG | 0.9 | 0.8 |  | SVM | HOG | 0.63 | 0.56 |
| CNN | BASE | 0.89 | 0.79 |  | CNN | BASE | 0.6 | 0.51 |

Table 3 Best Combination Model - Feature Extractor using the Balanced Classes Dataset for Genders (left) Age Classes (right).

Even though the SVM using the HOG method for feature extraction yielded better results than the CNN using no feature extraction, this was still considered for further optimization given its large manoeuvre area in terms of architecture improvement and mainly due to its immensely larger capability of handling larger datasets when compared to the SVM.

The fact that the best CNN in some studies was without a feature extractor may be due to the intrinsic characteristics of this type of architecture, where the convolution layers act as a type of feature extractor themselves. This is supported by authors such as Zhang et al. (2018) who compared the performance of a CNN with and without a feature extractor for recognizing handwritten digits and found that the CNN without the feature extractor achieved better accuracy, which may be due to the convolution layers in the CNN acting as a feature extractor in themselves. Similarly, in a study by Wang et al. (2019), the authors compared several CNN architectures with and without feature extractors for facial expression recognition. They found that the CNNs without feature extractors outperformed the CNNs with feature extractors, suggesting that the convolution layers in CNNs can effectively extract discriminative features from facial images.

The fact the best CNN was without feature extractor may be due to the intrinsic characteristic of this type of architecture which belong to it convolution layers which act as a type of feature extractor in themselves reducing the image to smaller and smaller dimensions, maintaining it “fundamental” characteristics.

## Optimized Models

After applying all the optimizations referred to in the methodology presentation, the following results were obtained (Table 4):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Model** | **Feature Extract** | **ACC** |  | **Model** | **Feature Extract** | **ACC** |
| CNN | BASE | 0.9 |  | CNN | BASE | 0.65 |
| SVM | HOG | 0.9 |  | SVM | HOG | 0.63 |

Table 4 Optimized models with ACC values for Gender (left) and Age Classes (right)

Starting with the CNN’s, it obtained an ACC of 0.65 for the Age classification problem, this model was able to improve compared with the previous stage architecture obtaining the overall best ACC value for the Age classification problem (0.65). This was attained using the SGD optimizer. Its learning curves (loss and acc) can be seen in Figure x.

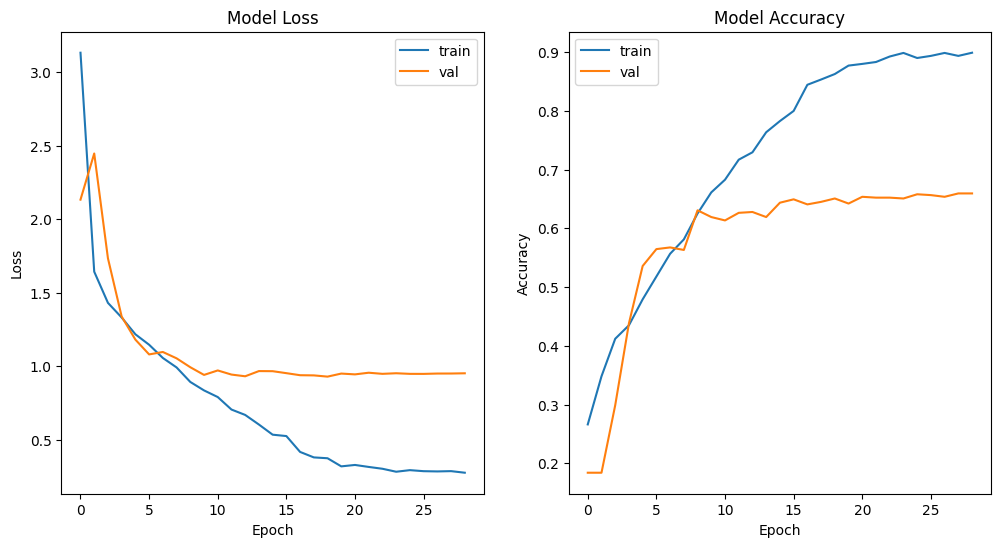


Figure 1 Loss (left) and Accuracy (right) curves for train and validation sets regarding the Otimized CNN for the balanced Age Classes

Observing the plots in Figure x it is apparent that the model started to show signs of overfitting from epoch 10 given that the validation set accuracy stabilized while the training set accuracy continued to increase which means that even if it continues to increase it lost its ability to generalize to “unseen” faces. The confusion matrix can be seen in Figure X in the annexes.

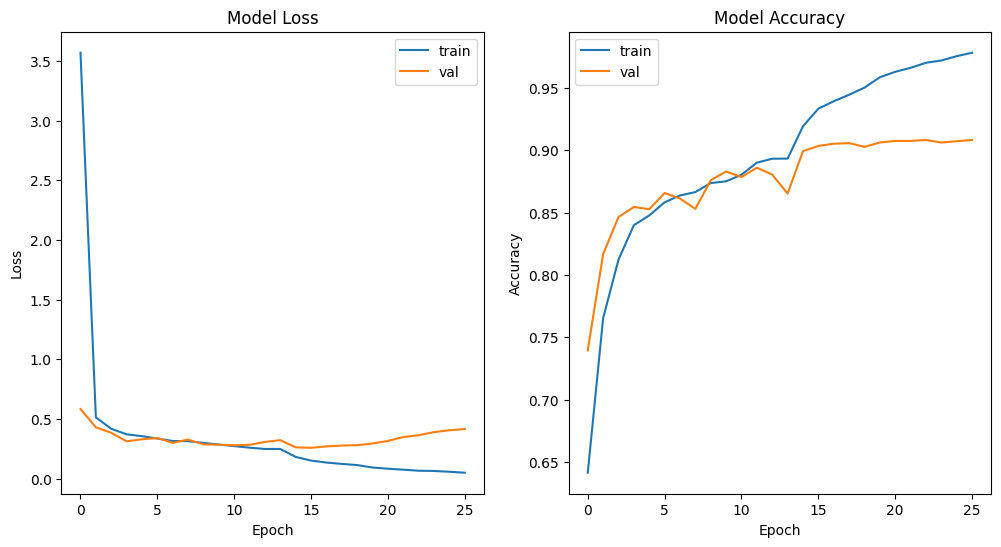
Regarding the Optimized CNN for the gender classification problem a 0.9 of ACC was obtained which was an improvement compared to the previous stage CNN for the same problem but still only managed to equalize to the SVM – HOG. This ACC value was obtained using the ADAM optimizer. Its learning curves (loss and acc) can be seen in Figure x.

Figure 2 Loss (left) and Accuracy (right) curves for train and validation sets regarding the Optimized CNN for the balanced Gender Classes

Observing the plots in Figure x it is apparent that the model started to show signs of overfitting close to epoch 15 given that the validation set accuracy stabilized while the training set accuracy continued to increase which means that even if it continues to increase it lost its ability to generalize to “unseen” faces. The confusion matrix can be seen in Figure X in the annexes.

For the SVM’s, no optimization was applied, only the sample size increased for the gender given that for the age classes it got greatly reduced when balancing. Still, it attained an ACC value as good as the CNN (0.90) for classifying genders and minimum difference when of 0.02 acc for the age classes. The confusion matrixes for the SVM using HOG for both gender and Age can be seen below in Figure x

A close-up of a graph

Description automatically generated with low confidence

Figure 3 Confusion matrixes for Optimized SVM - HOG regarding both Gender (left) and Age (right) Balanced Classes

For the

# Final comments – Global assessment of the project: usefulness; what has been shown; limitations; possible improvements or extensions

* Points to cover
  + Was the SVM of the original paper effectively replicated? (ACC)
  + CNN is the proposed Model for both classification problems given its higher computational capability
  + In what classification problem did we create a better model?
  + Balanced Datasets importance
  + Computational demand: GPU importance for running DL models; RAM for SVM’s; Overall resources availability
  + The use of IVS would be a good form of validation

# Bibliography

Cortes, C., & Vapnik, V. (1995). Support vector machines. Machine learning, 20(3), 273-297. From: https://courses.cs.duke.edu/fall15/compsci527/notes/hog.pdf

Tomasi, C. (2012). Histograms of oriented gradients. *Computer Vision Sampler*, 1-6.

Li, F.-F., Johnson, J., & Yeung, S. (2018). Convolutional neural networks for visual recognition. Retrieved from <http://cs231n.stanford.edu/reports/2017/pdfs/300.pdf>

Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). ORB: An efficient alternative to SIFT or SURF. In 2011 International Conference on Computer Vision (pp. 2564-2571). IEEE.

<https://arxiv.org/abs/1409.1556>

Hiremath, J. S., Hiremath, S. S., Kumar, S., Chincholi, M. S., Patil, S. B., & Hiremath, M. S. (2022). Gender Detection using Facial Features with Support Vector Machine. *2022 IEEE 2nd International Conference on Mobile Networks and Wireless Communications (ICMNWC)*, 1–6. <https://doi.org/10.1109/ICMNWC56175.2022.10031856>

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Fei-Fei, L. (2015). Imagenet large scale visual recognition challenge. International Journal of Computer Vision, 115(3), 211-252.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

Wang, H., Liu, Z., Zhang, Y., & He, D. (2019). A comparative study of deep learning models for facial expression recognition. Journal of Computational Science, 30, 80-91. https://doi.org/10.1016/j.jocs.2018.11.012

Zhang, X., Wu, P., Zhang, Y., Wang, H., & Li, H. (2016). Face recognition based on support vector machine and principal component analysis. In Proceedings of the 8th International Conference on Advanced Computer Control (pp. 372-377).

Zheng, H., Pan, X., Wang, S., & Wang, X. (2018). Object recognition based on support vector machine. In Proceedings of the 2018 3rd International Conference on Image, Vision and Computing (ICIVC) (pp. 118-121).

Zhang, X., Zhang, Z., Zhang, C., & Sun, M. (2018). Comparison of deep learning methods for handwritten digit recognition. In Proceedings of the 2018 International Conference on Artificial Intelligence and Industrial Engineering (pp. 118-122). ACM.

Jiang, Y., Li, S., Zhang, J., Liu, J., & Xiang, G. (2021). Medical image classification with feature extraction based on Support Vector Machine. Multimedia Tools and

## 

# Annexes

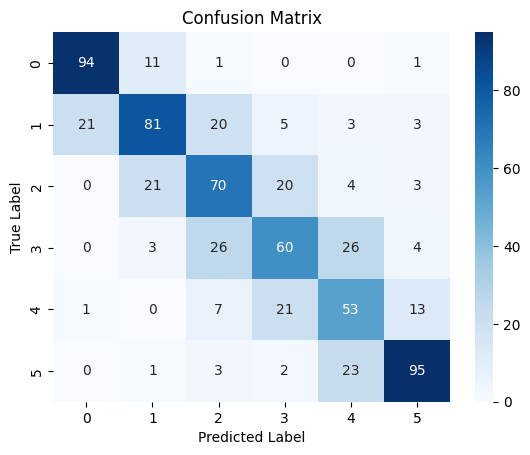


Figure 4 CNN otimized confusion matrix for Age Classes

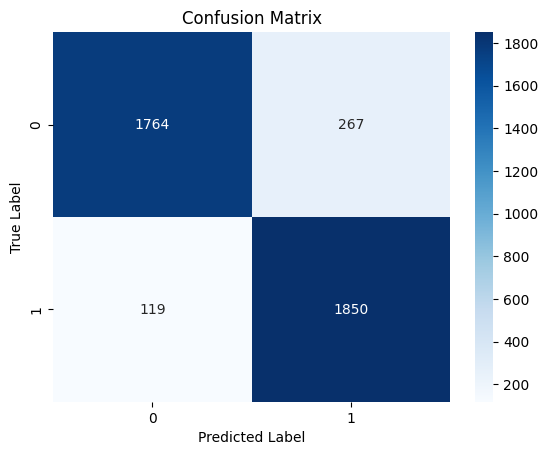
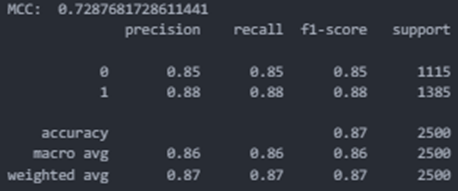
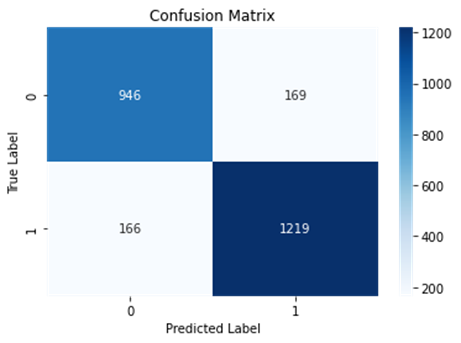


Figure 5 CNN otimized confusion matrix for Gender Classes

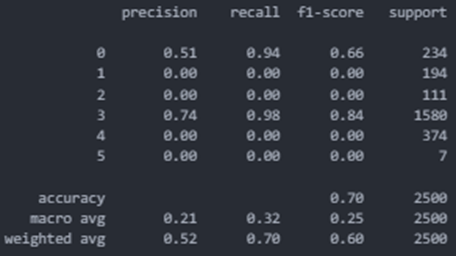
* SVM
  + Base:
    - Gender

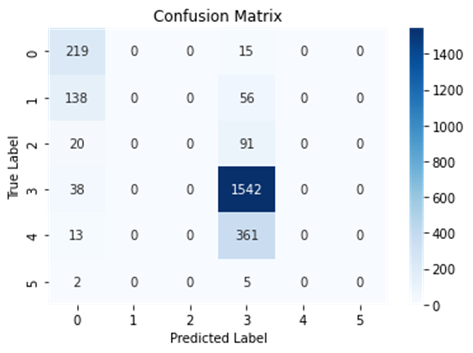




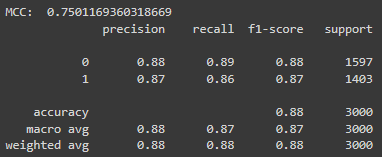
* + - Age

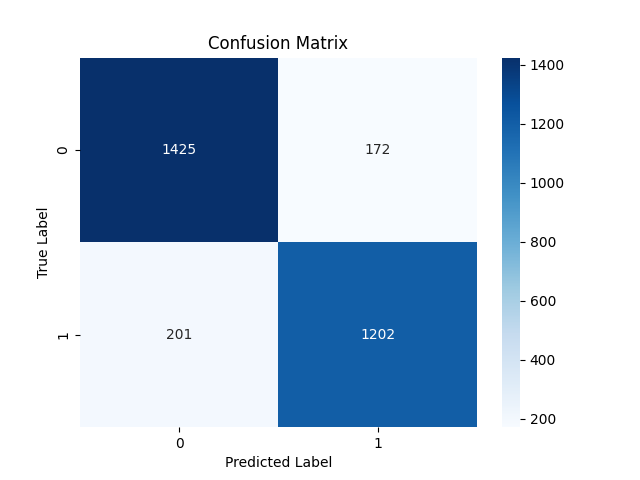
~<



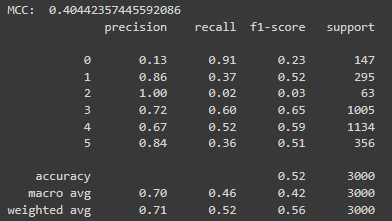


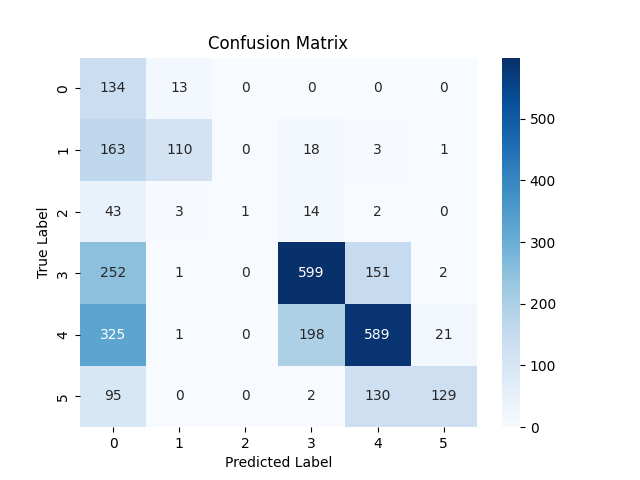
* + HOG
    - Gender



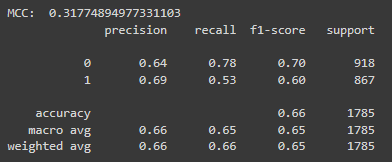


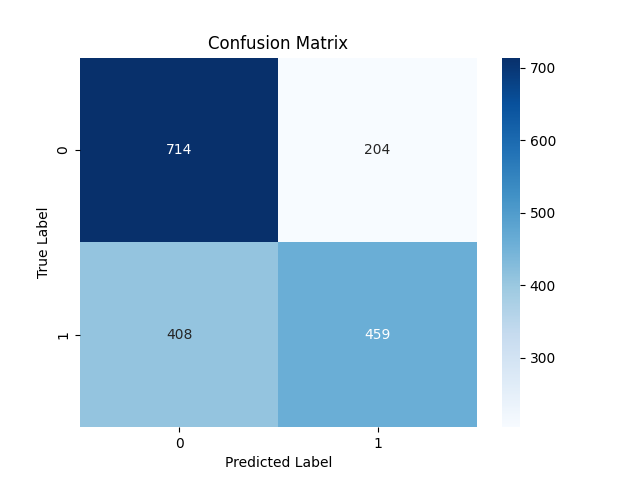
* + - Age



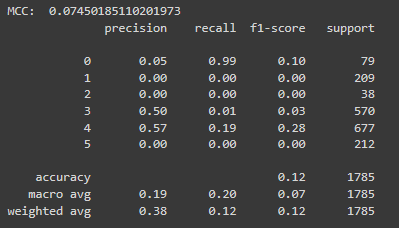


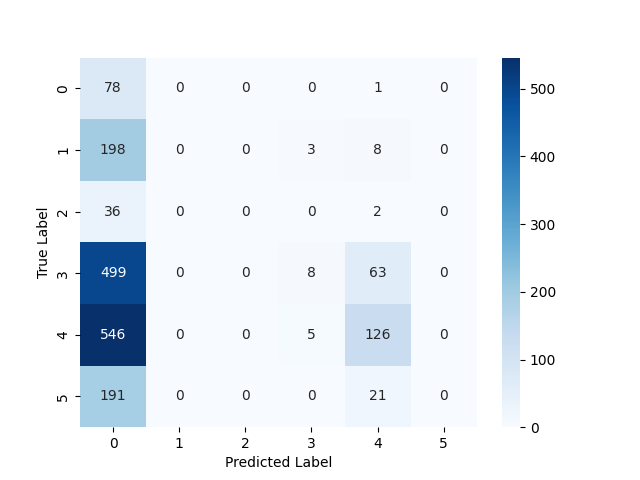
* + ORB
    - Gender



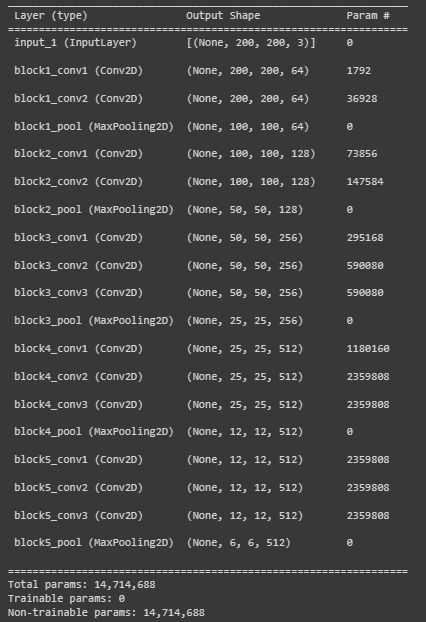


* + - Age

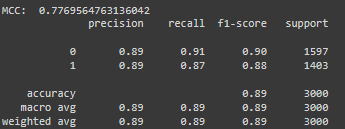




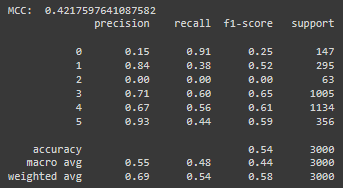
* + VGG16

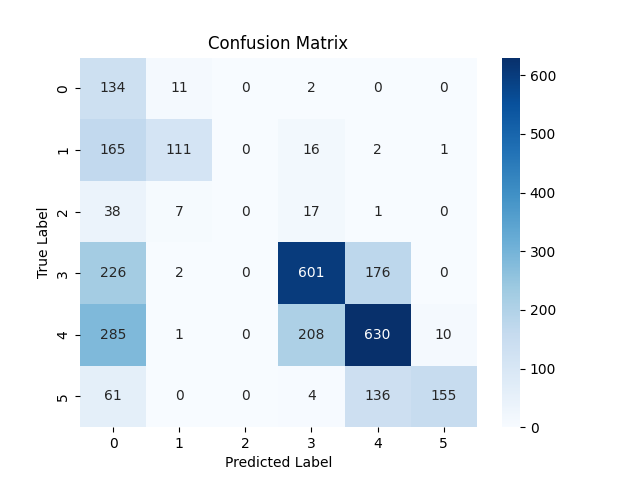


* + - Gender

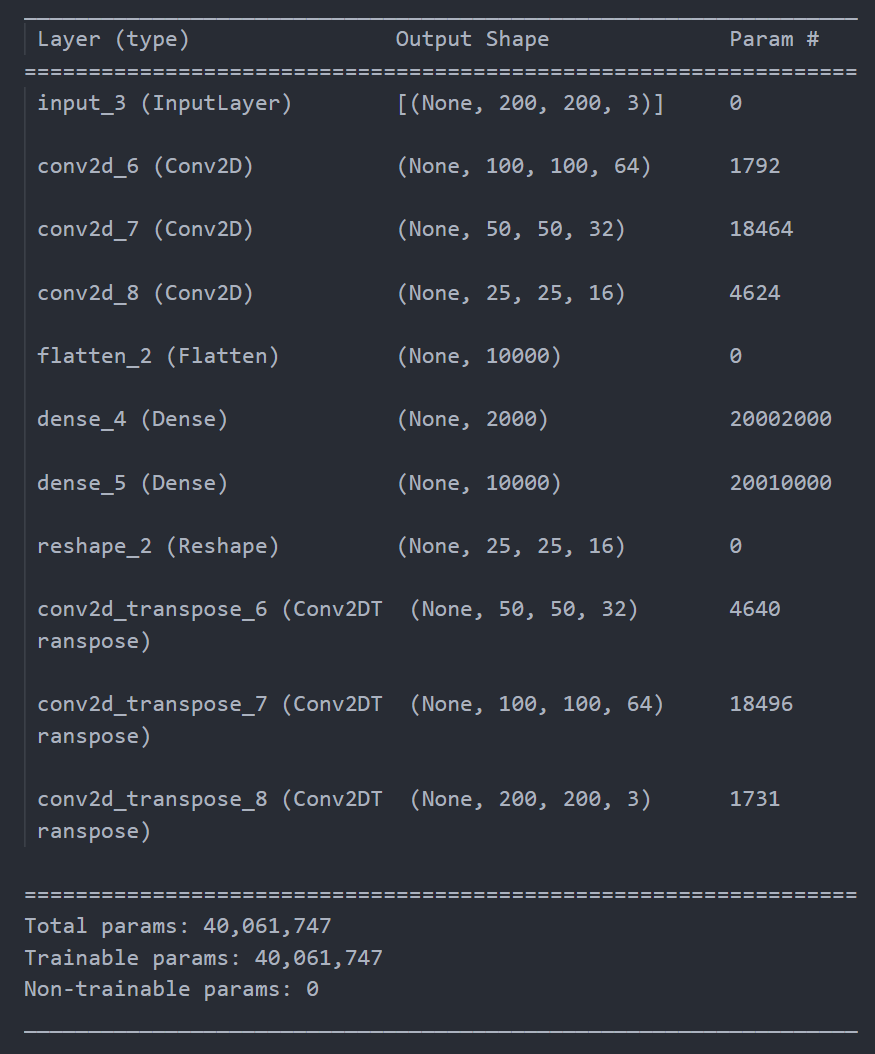


* + - Age

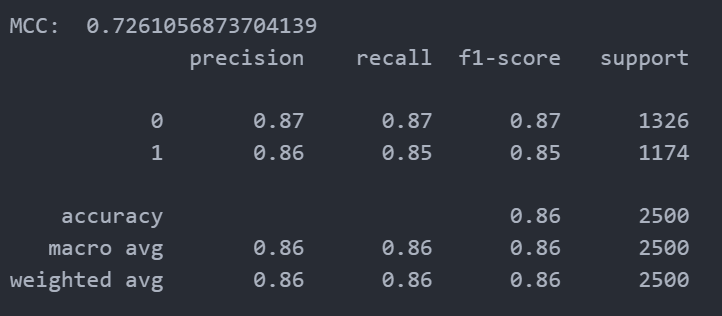


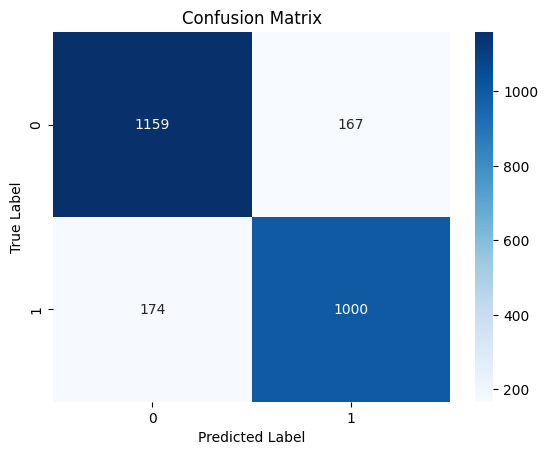


* + Autoencoder

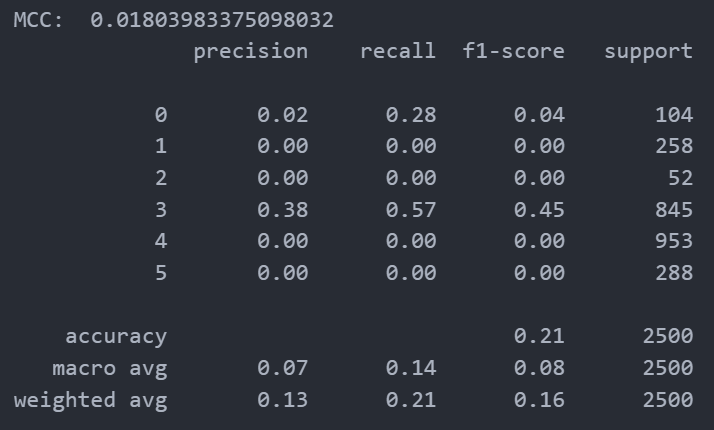


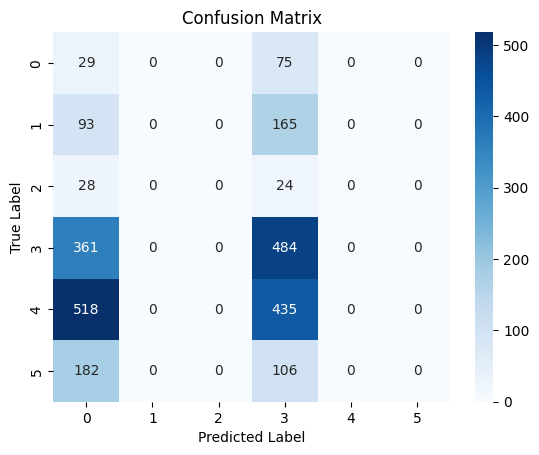
* + - Gender





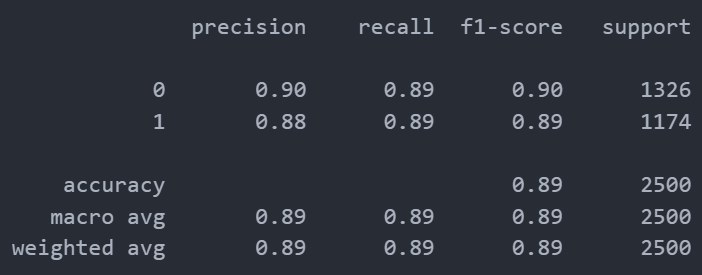
* + - Age

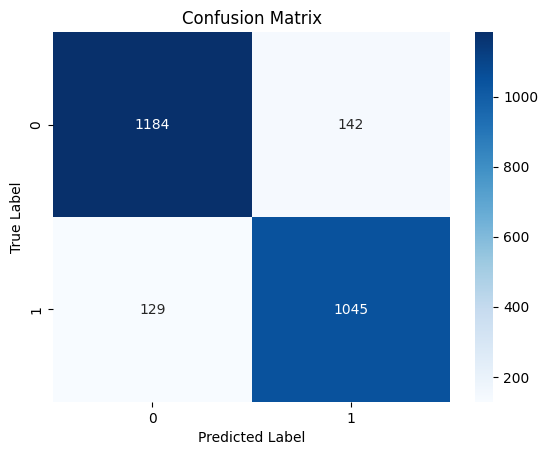




* CNN
  + Base:
    - Gender

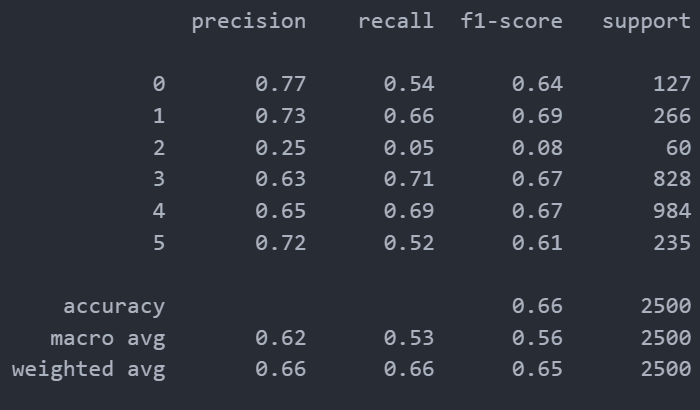


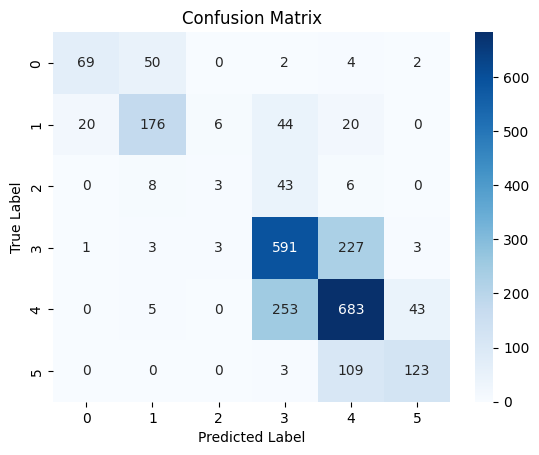




* + - Age

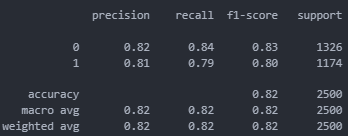


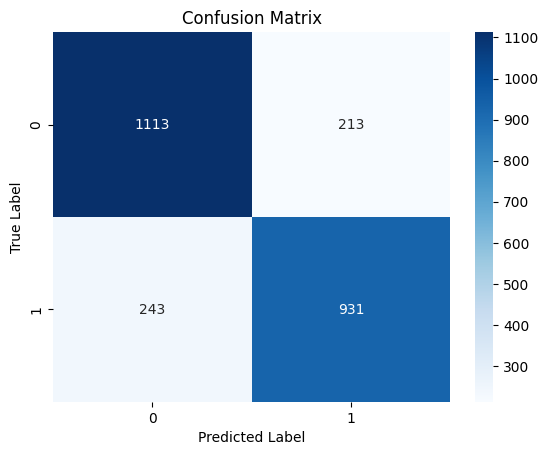




* + HOG
    - Gender

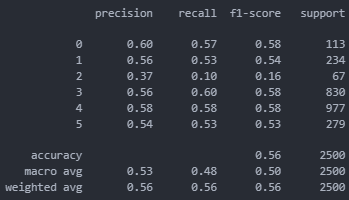


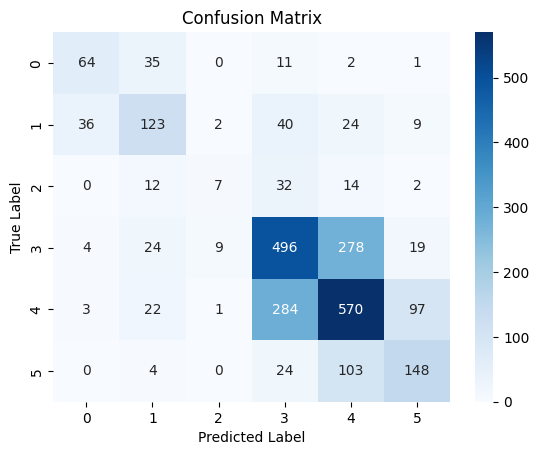




* + - Age

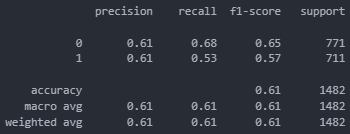


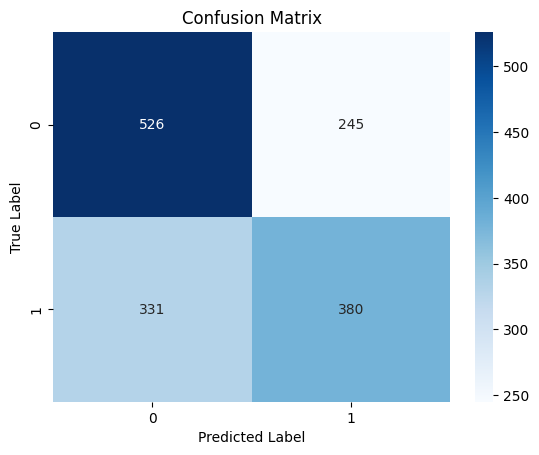




* + ORB
    - Gender

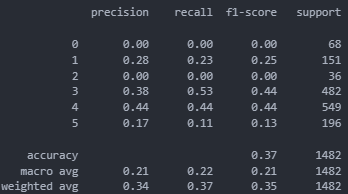


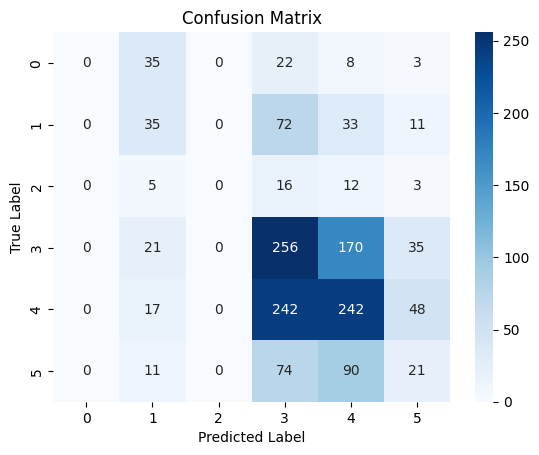




* + - Age

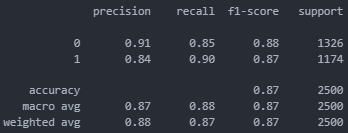


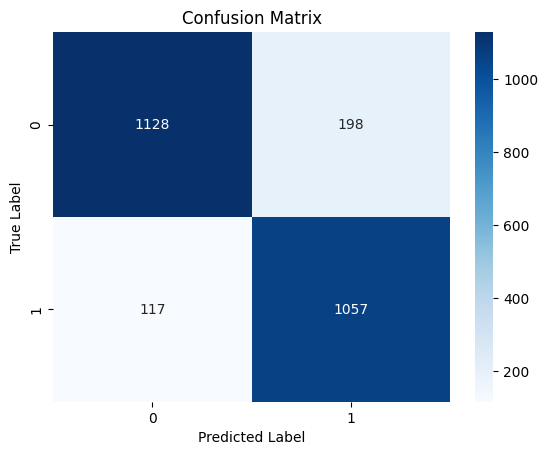




* + VGG16
    - Gender

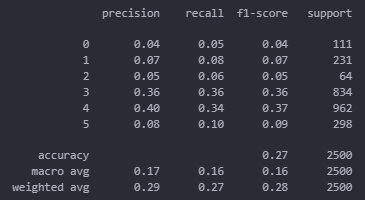


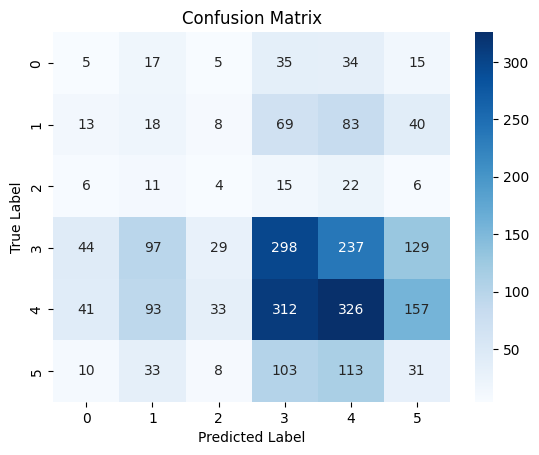




* + - Age

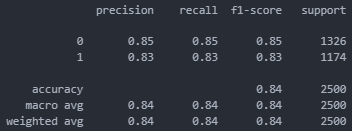


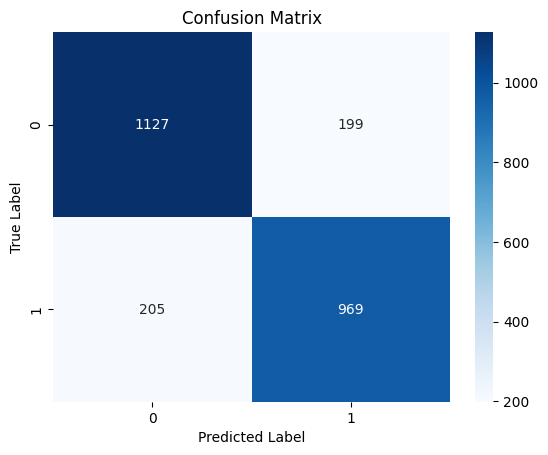




* + Autoencoder
    - Gender

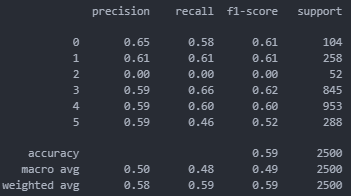


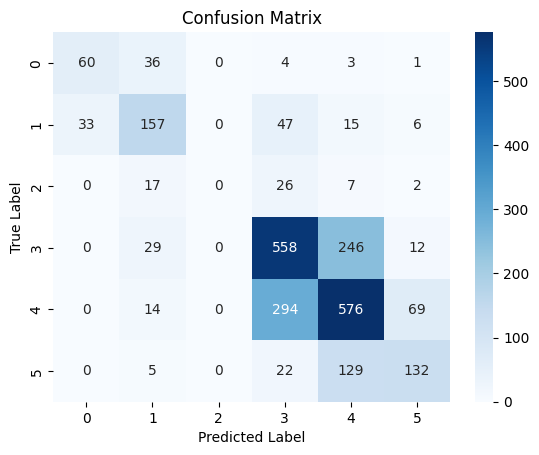




* + - Age







* Best Model with best feature extraction method

- Caso o CNN seja um deles obter plots para LOSS e ACCURACY

* + Gender
  + Age

Annexes

Bibliography

[1] https://www.nih.gov/nih-style-guide/age