Abstract – In this paper we present an approach of applying machine learning algorithms to the task of predicting human attractiveness. We have collected human beauty ratings of female facial images. We have chosen eigenfaces and ratio-based features as face representations. Along with k-nearest neighbors, we have used neural network and AdaBoost algorithms, which had not been used for this task before. Our analysis shows that machine learning algorithms have a preference towards facial symmetry, but also that a wider set of features needs to be included. We validate our results with a survey of four participants, which shows that facial attractiveness is a highly subjective judgement.

I. INTRODUCTION

The subject of visual processing of human faces has received attention from philosophers and scientists for centuries. Various experiments have shown a great influence of physical attractiveness on our lives, both as individuals and as a part of society. Its impact is obvious by the amounts of money spent on fashion trends and plastic surgery. And although a beautiful face is something we can recognize in an instant, it is still difficult difficult to define what exactly constitutes a beautiful face. Several face characteristics, such as symmetry, proportions and averageness, are however known to be positively related to facial attractiveness. However, the common notion is that the individual attractiveness is not generally predictable without the knowledge of person's cultural and behavioral background.

In this paper we explore the notion of facial attractiveness through the application of machine learning techniques. We present empirical results comparing the performance of two image representations — eigenfaces and ratio-based face representation — and different machine learning methods.

II. RELATED WORK

Relatively little research has been conducted on the specific task of machine learning approaches to attractiveness prediction. Existing tools mostly differ in methods used for face representation and machine learning techniques, and therefore in precision accomplished. Hefner and Lindsay [1] describe a system similar to ours in that eigenfaces method and ratio-based face representation are use for face representation with K-nearest neighbors as the learning algorithm. Additionally, they implemented SVM algorithm. The results of the project are comparable to our results, with ratio-based features outperforming eigenfaces. The dataset used in this project is the same as the one in this paper.

Whitehill et al. [3] used linear filters for edge detection (Gabor filter) and edge orientation histograms (EOH) for face representation. Using ε-SVM method they analyzed over 2000 photographs of male and female persons.

Eisenthal et al. [4] built an attractiveness prediction system that uses face representation methods as described in this paper, i.e., eigenfaces method and feature-based face representation. Best results were accomplished using linear regression methods, while other methods, such as SVM and KNN, gave slightly weaker results.

White et al. [2] used ridge regression, a Gaussian RBF kernel, and textural facial features to predict the mean attractiveness scores assigned to 4000 face images. After experimenting with several textural features, the best performance score was found to be the one achieved using kernel PCA on the face pixels.

III. THE DATASET

The whole dataset consists of 2250 images of female persons extracted from the website www.hotornot.com [2]. In our work we have used a subset of this dataset, as will be explained later. HotOrNot is a rating web-page that allows users to rate the attractiveness of photos submitted voluntarily by others. Each person in the dataset has an assigned beauty rating in the range of 1 to 10, which is an average score from www.hotornot.com ratings.

Only images with more than 50 voters and adequate face visibility were taken for the dataset. The rectified images were all downsampled to 86 by 86 pixels. Example images are given in Figure 1.

To simplify the attractiveness prediction problem, we categorized the dataset into two and four classes. When using two classes, the boundary value was chosen so to split dataset in half. That boundary is equal to median of all scores (score 7.9), thus dividing the dataset into more and less beautiful people than average. The boundaries for four classes are quartiles: 3.0, 7.9 and 9.0 of maximum 10.
III. FEATURES

We have chosen to use two different types of features as input to rating models. The first are facial ratios, e.g., mouth width divided by lip-to-forehead distance. Analyzing such ratios enabled us to test the significance of symmetry and other attributes, such as full lips or big eyes. The other face representation method used were eigenfaces. Whereas the first method is based on our subjective choice of a limited set of ratios to work with, eigenfaces provide an objective data conversion tool based on principal component analysis. A practical advantage of ratios over eigenfaces is that results obtained from ratios are more amenable to interpretation - it is easier to reason about results in terms of important ratios than with eigenfaces.

A. Ratios

Key points for defining distances were some of the prominent facial points, like edges of the eyes or tip of the nose. Fig. 2 a) illustrates some of the points used. Finding these points by a computer algorithm is a difficult task, so we decided to mark them manually. Each image used for training and test was marked with 40 points (136 pictures in total). Connecting some of the points yields a facial mask shown in Figure 2 b). Ratios were obtained by dividing important facial distances. Figure 2 c) illustrates a ratio obtained by dividing the distance A with the distance B. In total 25 such ratios were defined, often in such a way that it was possible to compare left and right side of a face (symmetry). Example could be a ratio that measures width of a smile to the left and to the right of the mouth centre. With this method, every face was represented with ratios as an input to the machine learning algorithms.

B. Eigenfaces

Eigenface (ger. eigen means inherent, characteristic, own) is a result of transforming a face with Principal Component Analysis (PCA). PCA (sometimes known as Karhunen–Loève transform or the Hotelling transform) is a mathematical transformation that transforms the data so that its output is sorted by a decreasing order of varying components. The first component of the output contains the most variation, and is possibly the most helpful in data variation; the second component contains the second most variation, and so on. This transformation is often used to reduce the dimensionality of the dataset. We have based our transformation to eigenface space on work of Serrano [14]. Sirovich and Kirby were the first to use the ideas of PCA to create eigenfaces [5], later improved by Turk [6]. A good introduction into the field can be found in [12]. Figure 3 illustrates faces from Figure 1 transformed into the eigenface space.

IV. CLASSIFICATION

To classify the images into two or four classes according to attractiveness, we have used three separate algorithms. We have implemented these algorithms in MATLAB computing environment. The first of these was k-nearest neighbors (k-NN): choosing the class according to local neighborhood. Distance between two pictures was calculated as Euclidean distance. Along with baseline k-NN, weighted version was also implemented, where the classification decision is influenced by the distance of K nearest neighbors and the picture to be classified.

The second implemented classifier was artificial neural network. Neural network is a robust classifier that models biological neural structures. We have used neural network primarily because it takes little assumptions on input data and has good generalization abilities. Their drawback is that it is hard to extract human understandable rules that explain the classification from a trained neural network. A survey into the field can be found in an article by Zhang [13]. We have used multilayered network with a sigmoid activation function and back propagation learning algorithm.
The third implemented classifier was trained by AdaBoost algorithm. The main idea of AdaBoost is to train many weak classifiers (i.e., those that classify the data barely above random guessing) in such a way that combined they make a strong classifier. Given enough strategies that work just barely better than random guessing, AdaBoost’s theoretical properties guarantee to make a classifier of arbitrary precision. A short introduction to AdaBoost and its properties can be found in Freund and Schapire [7]. Weak classifier in our case is based on only one ratio. E.g., if the ratio from Fig. 2 c) is close to 0.4, person is attractive. In the training process, the weights are assigned to the weak classifiers, so that the final decision is a weighted voting of week classifiers. Another reason why we used AdaBoost, besides favorable theoretical properties, is because it is easily possible to interpret its classification process – the intervals chosen in the training process for ratios provide insight into attractiveness cues. Also, the weight given to a ratio directly shows its importance.

V. RESULTS AND DISCUSSION

We have used 136 pictures to calculate ratios, of which 70 were used for training, 30 as validation set and 36 as test set. For the eigenfaces method we have used 200 picture dataset, of which 100 were used as a reference set for k-NN, and the other 100 for testing.

A. Two-class classification

In two-class classification, we divided the dataset into two equally-sized sets: more and less attractive than average. The threshold set to was median of all scores, 7.9 out of 10.

The graphical presentation of the results achieved by the two-class classification is depicted in Fig. 4. When classifying into two classes, k-NN had the best results. When the ratios were used as the input data, it had 61% correct classification for the baseline version, and 64% correct classification for the weighted version. When working with the distances in the eigenface space, k-NN achieved 67% correct classification. The best number of neighbors K was always between five and seven.

We have randomly initialized the neural network weights and trained it in multiple runs over the training data until the performance over the validation set started to drop, indicating overfitting. The final results on the test set were different from run to run, staying in the 45%-55% interval of correct classification. This indicates that in our case neural network did not generalize well, because at two-class classification, 50% is the success of random guessing.

Finally, AdaBoost algorithm had a success rate of 55% when working with ratios, which is slightly better than random guessing. However, weak classifiers produced during the training phase were instructive, showing strong preference towards symmetry. For example, the weak classifier with the biggest weight was based on the ratio of left to right eye, with the form:

\[0.93 < \frac{\text{left eye width}}{\text{right eye width}} < 1.06\] (1)

Another interesting result showed that four participants significantly differed in classification between themselves. Namely, only 48% of images were classified the same by all four of them, when having to choose between two classes. When using four-class classification, only 16% of images were classified equally by all four participants.

To further measure the annotator agreement, we have used Fleiss’ kappa measure. Fleiss’ kappa is a variant of Cohen’s kappa, a statistical measure of inter-annotator reliability. Fleiss’ kappa can be calculated for any constant number of annotators giving categorical ratings, to a fixed number of items. Fleiss’ kappa can be interpreted as expressing the extent to which the observed amount of agreement among annotators exceeds what would be expected if all annotators made their ratings randomly.

In our case, we had four annotators and 50 items. Fleiss’ kappa was calculated for two and for four class classification. If we interpret results by the Landis and Koch’s table, it would be as follows: in case with two classes there is poor agreement (κ = -0.0775) between annotators and with four classes there is slight agreement (κ = 0.1675). While it is usual to have less agreement for fewer classes, in our case it is the exact opposite.

In our case, we had four annotators and 50 items. Fleiss’ kappa was calculated for two and for four class classification. If we interpret results by the Landis and Koch’s table, it would be as follows: in case with two classes there is poor agreement (κ = -0.0775) between annotators and with four classes there is slight agreement (κ = 0.1675). While it is usual to have less agreement for fewer classes, in our case it is the exact opposite.

In our case, we had four annotators and 50 items. Fleiss’ kappa was calculated for two and for four class classification. If we interpret results by the Landis and Koch’s table, it would be as follows: in case with two classes there is poor agreement (κ = -0.0775) between annotators and with four classes there is slight agreement (κ = 0.1675). While it is usual to have less agreement for fewer classes, in our case it is the exact opposite.

We have also directly compared the classification of k-NN with human classification, but the results were not better than random guessing, which is understandable because machine learning algorithm was trained on the
original dataset, unlike training it to match the taste of survey participants.

These results show that our machine learning approach did better (67%) on the dataset than any of the annotators (60%). The variance of the classification made by humans raises a question of whether an objective measure of facial attractiveness can truly be made.

D. Comparison with related work

The results presented in this paper are comparable to the previous work most similar to ours, done by Hefner and Lindsay [1]. We achieved 67% two-class classification, while their best result was 70%. This indicates that current attempts to evaluate attractiveness by machine learning are still somewhat limited.

This, however, is not surprising from a biological point of view. In his classical work in evolutionary biology, Ridley [10] emphasizes the importance of facial attractiveness in mate selection. Therefore, it should not be surprising that humans have evolved to be able to spot facial nuances better than a machine learning algorithm can.

The accuracy of classification deteriorates with the increase in the number of classes, because it becomes harder to separate the classes when they are closer.

VI. CONCLUSION AND FUTURE WORK

To evaluate facial attractiveness, along with KNN algorithm we have applied neural network and AdaBoost algorithms. These algorithms haven’t been used before in related research, but have also not proven to be effective in our work.

k-NN had the best results in two-class classification with 67% accuracy, similar to related work [1].

Our research is unique in that we have validated our results with a survey taken by four participants. All four had less than 60% agreement with our dataset for two-class classification, worse than our machine learning algorithm. Participants also varied significantly within themselves, which indicates that evaluation of facial attractiveness is inherently subjective.

For further work, we suggest adding a new set of higher level features such as the color of the hair, skin texture and even combining expression recognition algorithms [15] with our current work.

VII. REFERENCES