# Relative Ranking of Facial Attractiveness

Hani Altwaijry and Serge Belongie
Department of Computer Science and Engineering
University of California, San Diego
{haltwaij,sjb}@cs.ucsd.edu

## Abstract

*Automatic evaluation of human facial attractiveness is a challenging problem that has received relatively little attention from the computer vision community. Previous work in this area have posed attractiveness as a classification problem. However, for applications that require fine-grained relationships between objects, learning to rank has been shown to be superior over the direct interpretation of classifier scores as ranks [27]. In this paper, we propose and implement a personalized relative beauty ranking system. Given training data of faces sorted based on a subject's personal taste, we learn how to rank novel faces according to that person's taste. Using a blend of Facial Geometric Relations, HOG, GIST, L*a*b* Color Histograms, and Dense-SIFT + PCA feature types, our system achieves an average accuracy of 63% on pairwise comparisons of novel test faces. We examine the effectiveness of our method through lesion testing and find that the most effective feature types for predicting beauty preferences are HOG, GIST, and Dense-SIFT + PCA features.*

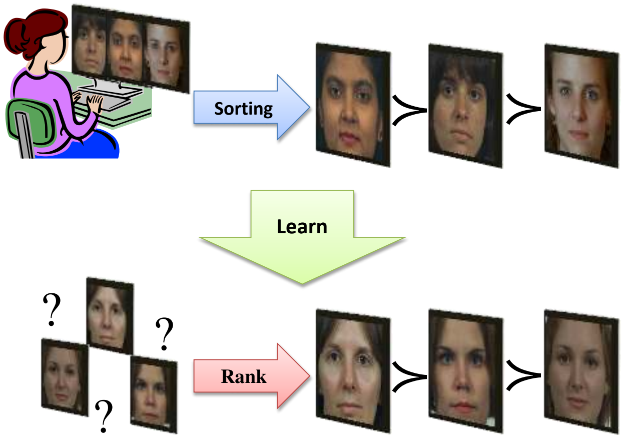

Figure 1. A user trains the system by sorting a set of faces according to perceived attractiveness. Our system learns from this ordering how to rank a novel set of faces according to that user's preferences. No two users perceive beauty in the same way; our system is designed to adapt to each user's taste.

## 1. Introduction

What makes a face beautiful is a hard question. Facial attractiveness is judged differently by individuals and many factors, such as symmetry or facial expressions, can contribute to why a face is attractive. Although many can agree on several beautiful faces, individual differences in taste exist. This makes *attractiveness* a complex feature on which different individuals might have different subjective opinions, unlike other simple objective facial features such as eye-size or mouth-size which are measurable.

Today, millions go on web-based dating services and browse a huge number of user profiles [34]. These dating services typically provide searching tools that leverage textual information in user profiles. We are currently not aware of any services that leverage profile pictures for automated beauty filtering based on user preferences. To accommodate such features, we are influenced by the literature of information retrieval and web-search [23] in adopting a ranking scheme. Such a ranking scheme should enable us to define fine-grained relationships among faces in terms of their attractiveness. This would allow dating services to present their users with profiles sorted according to their own taste in beauty.

As described in [37], recent research in this area poses the problem of attractiveness as a classification problem following a single universal notion of beauty [1, 19]. On the other hand, research in the area of determining "visible features" in faces, and images generally, is widespread [9, 10, 21], where usually the goal is to answer whether an attribute is present, *e.g.* is there a smile or not. In [27], the recognition of attributes in a relative manner was introduced through the use of a modified RankSVM [16, 18] approach. However, their work was limited to relations expressed on a small number of categories with multiple examples per category (8 categories representing the faces of 8 individuals).

978-1-4673-5052-5/12/$31.00 ©2012 IEEE

The granularity on which the relationships were expressed was limited to the category level, and not the image level.

In this paper, we tackle the problem of facial attractiveness from the perspective of the individual. We define our goal as: given a set of faces ordered by preference, learn how to rank novel faces using a criteria similar to that of the individual who presented that ordering. For example, in Figure 1, we see three faces that have been sorted by a user according to their perceived attractiveness. Our method's goal is to learn that user's taste, and then rank novel faces accordingly. Unlike previous work, this paper focuses on differentiating between any two faces, in terms of which face is more attractive based upon an individual's preference. We stress that in our work, we express relationships over the attractiveness of faces at the fine granularity of pairs of faces, and not at some coarse granularity of a categorical fashion.

We summarize this paper's contributions as follows: we introduce a system that predicts the relative ranking of facial attractiveness based on different individuals' tastes. The average accuracy achieved by our system is 63% as defined by the number of correctly ordered pairs among all possible pairs. This accuracy metric is based on the Kendall Tau [22] rank similarity measure. While we are not aware of any benchmarks to which we can compare this result, it is significantly better than random, for which the accuracy would be 50%. We believe our result has significance especially when considering the numerous factors affecting a given permutation, such as the intra-rater's consistency, i.e. the existence of a trend that the individual maintains throughout his preferences. Furthermore, we present a simple approach for combining Dense-SIFT [3] with PCA to represent faces, which delivered competitive results as to the other known feature types.

### 1.1. Dataset

We used the widely known Color FERET dataset [29, 30] to extract 200 female faces that were cropped and affine transformed into frontal faces with a size of 250×250 pixels.[1] The transformations were based off hand-clicked points that registered the locations of several facial features. As the FERET faces are mostly frontal with slight rotations, the affine transforms did not damage the quality of the images and we believe had no effect on the perception of beauty. Figure 2 shows two examples of original faces and their affine transformed counterparts.

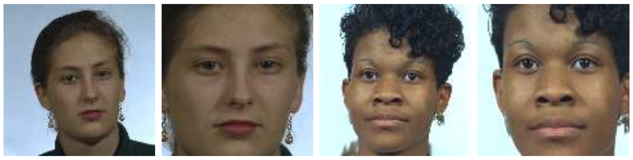

Figure 2. A sample of two faces from the dataset which had slight rotations and their respective cropped and affine transformed face.

### 1.2. Organization

This paper is organized in the following manner. Section 2 discusses some of the related work. Section 3 describes the approach of this paper in tackling this problem. Section 4 discusses our experiments and their results. Section 5 presents the current future direction of this work. Section 6 concludes.

## 2. Related Work

Relatively little work has been done in the area of personalized relative ranking, especially of facial attractiveness. This section reviews this related work.

Bottino and Laurentini [4] delivered a study of facial beauty as seen in the pattern analysis literature. The Color FERET dataset had been previously put in beauty-related research [11, 31], however unlike our work, beauty is dealt with as a universal notion.

Whitehill and Movellan [37] have tackled the beauty classification problem from the individual's perspective, and have employed various computer vision techniques that used feature sets such as Eigenface projections, Gabor filters, Edge orientation histograms (EOH), and Geometric relations with $\epsilon$-SVM. In their experiment, four classes of attractiveness were employed, and they were able to achieve 0.45 correlation in their prediction process. Finally, they also report on the best feature types used in their experiment, which were the Gabor features vs. PCA, EOH, and Geometry.

Sutic *et al.* [33] used features such as Eigenfaces and Geometric ratios with the algorithms: k-Nearest Neighbor (k-NN), Neural Networks, and AdaBoost. Their experiment performed two-class and four-class classifications. Their best results in the two-class case were obtained using Eigenfaces with k-NN. The correlation accuracy went up to 0.67 for Eigenfaces with k-NN, and 0.61 for Geometric ratios with k-NN. As for their four-class case, the best results were obtained through k-NN with accuracy of about 0.33.

Eisenthal *et al.* [8] used Eigenfaces and Geometric features with SVM, k-NN, and linear regression in performing beauty classification. They obtained their best results using Geometric features with SVM and with linear regression. The correlation accuracy reached 0.6.

Kumar *et al.* [21] used a number of binary classifiers to describe many facial features for face verification purposes.

[1] Due to the Color FERET Release Agreement, we are not allowed to show more than 15 images in this paper.
http://biometrics.nist.gov/cs_links/face/frvt/feret/colorferet/ColorFeret_release_agreement.pdf

Furthermore, they used "simile classifiers" that were able to describe novel faces as with respect to a limited set of reference faces. The inherent binary set-up of their work deems it inappropriate for our goal, for example, defining a "beauty" attribute would not yield a ranking of beauty degrees as we suggest here. Moreover, one might try to define beauty with respect to different celebrities and use simile classifiers to define a rank, however, this approach also lacks ranking aspects as interpreting a classifier score for ranking purposes was shown to be inaccurate by [27].

Parikh and Grauman [27] introduced *Relative Attributes*, a system for describing relative features between pairs of images. Their work was based on a small number of categories that contained many examples, with relationships expressed and generalized across the categories instead of the individual images, and moreover, the relatively described attributes were ones on which any individual would give the same answer for, i.e. the difference of individuals' opinions did not exist.

Ce Liu *et al*. [24] presented SIFT Flow, an approach that used Dense-SIFT for scene alignment and recognition. Jian-Gang Wang *et al*. [36] followed SIFT Flow in using Dense-SIFT to represent faces by placing 128 dimensional SIFT descriptors in place of pixels. AdaBoost was then used to identify the gender of test faces using that representation. The use of AdaBoost overcomes the high-dimensionality of the representation as Jian-Gang explains. In our approach, however, the use of Dense-SIFT was followed by dimensionality reduction using PCA. Our use of PCA on Dense-SIFT differs from [20].

Ranking and relative ordering has been thoroughly investigated within the machine learning literature especially in applications pertaining to information and document retrieval [5, 18, 23], and in applications of collaborative filtering for movie recommendations such as those used in TiVo and Netflix [2, 28, 39]. We also find image search related applications for image retrieval using similar techniques that were used for document retrieval [13, 15, 32]. Our work differs significantly from those as we learn a ranking function to specifically differentiate between individuals' tastes expressed over image pairs.

# 3. Our Approach

In this section we present our main learning technique (Section 3.1), and how we obtain the relative orders and the sorting of faces (Section 3.2). We conclude by discussing our Dense-SIFT facial representation (Section 3.3).

## 3.1. Learning to Relatively Rank

Our learning method builds upon the one used by Parikh and Grauman [27]. In this work, we do not require the constraint of equivalence although for generality we will be including it. The faces in the dataset are represented as a set $F = \{F_1, F_2, ..., F_n\}$. The sorted faces list is given by a subject as an ordered tuple $O = \langle F'_1, F'_2, ..., F'_n \rangle$ and it is interpreted as $F'_1 \succ F'_2 \succ ... \succ F'_n$, where "$F'_i \succ F'_j$" denotes that $F'_i$ is more attractive than $F'_j$. Moreover, "$F'_i \sim F'_j$" denotes that both $F'_i$ and $F'_j$ are equivalent in terms of attractiveness. We use $F'$ here to distinguish the sorted faces from the unsorted ones.

Let $x_i$ represent the feature vector of $F_i$, our goal is to learn the function:

$$g(x_i) = w^\top x_i \tag{1}$$

subject to the constraints:

$$\forall F'_i, F'_j, i \neq j, F'_i \succ F'_j \rightarrow g(x_i) > g(x_j) \tag{2}$$

$$\forall F'_i, F'_j, i \neq j, F'_i \sim F'_j \rightarrow g(x_i) = g(x_j) \tag{3}$$

The problem as described by [27] can be solved by the introduction of slack variables and is modeled as the following optimization problem:

$$\hat{w} = \arg\min_{w} \left( \frac{1}{2} ||w||_2^2 + C \left( \sum \gamma_{ij}^2 + \sum \xi_{ij}^2 \right) \right) \tag{4}$$

such that:

$$\forall F'_i, F'_j, i \neq j, F'_i \succ F'_j \rightarrow w^\top (x_i - x_j) \geq 1 - \xi_{ij}^2 \tag{5}$$

$$\forall F'_i, F'_j, i \neq j, F'_i \sim F'_j \rightarrow |w^\top (x_i - x_j)| \leq \gamma_{ij}^2 \tag{6}$$

$$\gamma_{ij} \geq 0, \xi_{ij} \geq 0 \tag{7}$$

where $C$ controls the satisfaction of strict relative order vs. establishing a greater margin between the examples, and $\gamma_{ij}, \xi_{ij}$ are slack variables.

In our implementation, we restricted the problem to strict "more attractive than" relationships for reasons related to our sorting mechanism which produced a fully ordered set which corresponds to the tuple $O$. Moreover, this setting enabled us to use ranking metrics, as in [22], to establish an accuracy measure for our system. Our implementation is based on the RankSVM [16, 18] code by O. Chapelle [6] which originally solves the primal SVM problem without the restriction imposed by equation (6).

## 3.2. Sorting and Relative Order

Obtaining the sorted orders of individuals' tastes is a crucial component of the system. The discouraging aspect of having a fully ordered list is the high cost of sorting that is magnified when performed by each individual. A perfectly ordered list would require the individual to perform a large number of comparisons, where in the average case the number of comparisons for a data of size $n$ is $\Theta(n\, log_2\, n)$, *e.g.* if we had 100 faces, we would require over 600 comparisons on average.

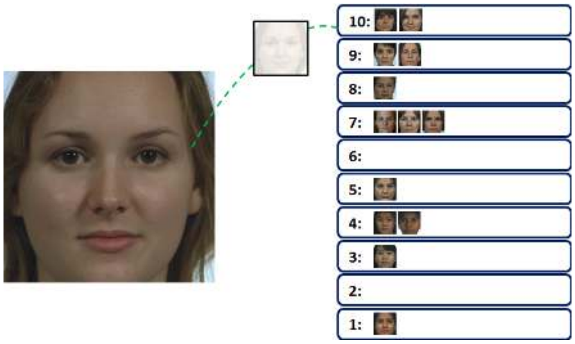

Figure 3. Phase one of the binned-sort method: the user is asked to drag the face placing it within the desired bin. The relationship between all the faces within a bin are predetermined with respect to all other faces in other bins.

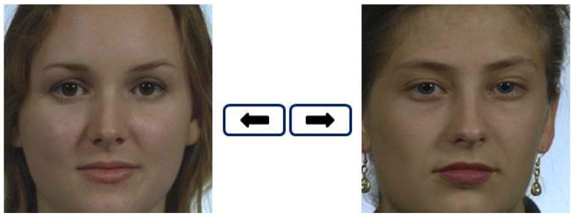

Figure 4. Phase two of the binned-sort method: the user chooses the more attractive face between the two shown faces.

To overcome this, we use a sorting method that includes binning. Thus, the sorting is split into two phases. In phase one, faces are placed in bins, where placing a face in a bin, say $A$, established that it is better than all faces in another bin, say $B$. For concreteness, the binned-sort method establishes that:

$$\forall F_i \in A, F_j \in B \rightarrow F_i \succ F_j \tag{8}$$

where A and B are bins of faces. We used ten bins to emulate a score from 1 to 10. However, we emphasize that this scoring emulation has no connection to the actual ranking, it is only meant to reduce the number of sorting operations. The first phase of this binned-sort is shown in Figure 3.

The second phase of the binned-sort begins by showing the user two faces, as shown in Figure 4, and the user is asked to choose the more attractive face. This phase actually implements a normal comparison sort algorithm with faces instead of numbers. At the end of this phase, we obtain the ordered tuple $O = \langle F'_1, F'_2, ..., F'_n \rangle$.

### 3.3. Dense-SIFT + PCA

Influenced by [36] we represented each face by computing a dense grid of SIFT descriptors [25] at each pixel with a spatial-bin size of 3×3 pixels. The resulting facial representation was a 241×241×128-dimensional matrix per face. The resulting representation is immensely larger in data-size as compared to the original faces. To reduce this high-dimensionality, we used PCA on each one of the 241×241 slices in the 241×241×128-dimensional facial representation. We then projected each slice onto a 20-dimensional vector, which size was chosen empirically based on the eigenvalues' decay. The resulting facial representation required a 2560-dimensional vector per face. This representation can be thought of as performing Eigenfaces on each dimension of the SIFT descriptors, separately. Our goal was to capture local gradient information in a compact form at the global level of the face.

## 4. Experiments

### 4.1. Features

To accomplish our ranking goal, we have extracted and combined several feature types together. We used these features in a usefulness testing scenario in our experiments as we will see shortly. Our set of features is composed by six feature types which were automatically extracted except for the Facial geometric features which involved some hand-clicked points.

**Facial Geometric Features:** The facial geometry features were derived by calculating the pixel distances within the eyes, the nose, and the mouth. The values were obtained through hand-clicking which we believe serves a near ideal measurement that gives some room for small jitter. The facial geometric feature vector was then formed by calculating different height-width ratios of the facial features mentioned.

**Eigenfaces:** [35] features were extracted from training and testing images based on Eigenface projections on the training data eigenvectors' space. The number of dimensions to project upon was chosen empirically based on the observation of the decay of the eigenvalues, in our case that was 20. These features were appealing as they were used in many of the earlier classification approaches for attractiveness and showed promising results.

**SIFT – Bag of Words:** SIFT [25] features were used as Geng and Jiang demonstrated in [12] their effectiveness in non-holistic face recognition and their feasibility in conditions where faces are not aligned. In the facial attractiveness domain, these were appealing as they were never employed beforehand. We imagined SIFT capturing specific attractiveness features that could only be found by local descriptors. We employed a technique from [38] that is based on the *bag-of-words* unsupervised model, where all the SIFT features from the training set are initially clustered using K-means and then used to create a K-dimensional *codeword* for each face by following a voting scheme. In our implementation, the choice of K was made empirically based on the average number of SIFT features per face.

**Dense-SIFT + PCA:** Our facial representation, as discussed in Section 3.3, took part in our experiments. Each face was represented by a 2560-dimensional vector storing local gradient information.

**GIST:** In our experiments we use a 960-dimensional gist [26] feature descriptor. Our use of this feature-type was influenced by its success in Relative Attributes [27].

**Histogram of Oriented Gradients:** We use a simple and straightforward 900-dimensional histogram of oriented gradients (HOG) [7] that was calculated on the whole image.

**Color Histograms:** To capture the effects of color on attractiveness, we use a 45-dimensional L*a*b* color histogram, with 15 dimensions per color channel.

### 4.2. Measuring Accuracy

To measure the accuracy of our method we turned towards tools for comparing ranked orders: Kendall Tau [22] and Gamma Test [14].

The Kendall Tau measures the number of pairwise inversions between two ordered lists $L_1, L_2$ as follows:

$$\tau(L_1, L_2) = \sum_{\forall (i,j) \in L_1} I((j,i) \in L_2) \tag{9}$$

where $I(\cdot)$ is an indicator function.

Based on the Kendall Tau we construct our accuracy measurement to account for correct pairs divided by the total number of pairs to reach a notion of correctness. If $N$ is the total number of pairs, then our accuracy measurement for a list $L_1$ matching $L_2$ is:

$$\alpha(L_1, L_2) = 1 - \frac{\tau(L_1, L_2)}{N} \tag{10}$$

The Gamma Test measures the difference between the number of correct pairs and the number of pairwise inversions and divides by their sum to yield a value from $-1$ (full negative association) to $+1$ (full positive association). A value of 0.0 indicates no association. It is calculated for two lists $L_1, L_2$ as follows:

$$A = \sum_{\forall (i,j) \in L_1} I((j,i) \in L_2) \tag{11}$$

$$B = \sum_{\forall (i,j) \in L_1} I((i,j) \in L_2) \tag{12}$$

$$\gamma(L_1, L_2) = \frac{A - B}{A + B} \tag{13}$$

The Gamma Test gives a result within the interval $[-1, 1]$ where our $\alpha$ measurement gives us a percentage. Later, we use the Gamma Test to show the dissimilarity between the sorted orders provided by the different subjects we had, while we use $\alpha$ to determine the accuracy of our attractiveness ranking method.

### 4.3. Collecting User Preferences

To test our method we presented 60 Amazon Mechanical Turk[2] Workers (MTurkers) with the 200 female faces which they sorted according to the binned-sort method described in Section 3.2. Initially, we requested a number of 30 MTurkers to perform the sorting *two consecutive times* to measure for *intra-rater consistency* and to prevent *adversarial* MTurkers from earning money by providing random data. Each sorting session took approximately four hours on average. We have discovered that the MTurkers struggled to keep their preferences consistent. We used our $\gamma$ measure to test for repeated sorting accuracy. The highest achieved consistency was $0.61$ by a single sample, the mean was $0.34$, and the minimum was $-0.04$. That minimum indicated that the two preference lists were “random” with respect to each other.

The execution of our ranking system showed slight accuracy changes for that sample versus the $\gamma$ measurement of the users’ two preference lists, with higher variance statistics for preference list pairs that showed disassociation, i.e. lists with $\gamma$ values closer to zero. Figure 5 shows the results of this comparison. The small variance value, for highly associated pairs, indicates that good user consistency verifies the existence of a “beauty-trend” in the data. Here, a “beauty-trend” is a set of visual features which the user preferred over other visual features consistently across his decisions. However, higher variance values, in pairs that showed disassociation, indicates the unreliability of using the Gamma Test as a measure for the existence of a beauty-trend.

We have therefore concluded that intra-rater consistency within the provided data does not reveal whether a beauty-trend exists, and have continued our experiment asking the remaining 30 MTurkers to sort only once.

### 4.4. Facial Attractiveness Ranking

We designed our experiments creating different feature combinations by gradually introducing features and removing some. We ran our tests on the preference list of each MTurker by splitting the list into 160 training examples, and 40 testing examples. The splitting was performed through uniform sampling of the preference lists.

Running the experiment with the 60 preference lists provided by the MTurkers showed mixed results as the ranking accuracy plot in Figure 6 shows. The highest achieved average accuracy was about 61%. The different feature combinations showed mixed results especially as indicated by the error bars. We believed this was due to some randomly sorted lists provided by *adversarial* MTurkers that could not be detected. Therefore, we decided to observe the method’s behavior given random permutations generated by

[2] https://www.mturk.com/mturk/welcome

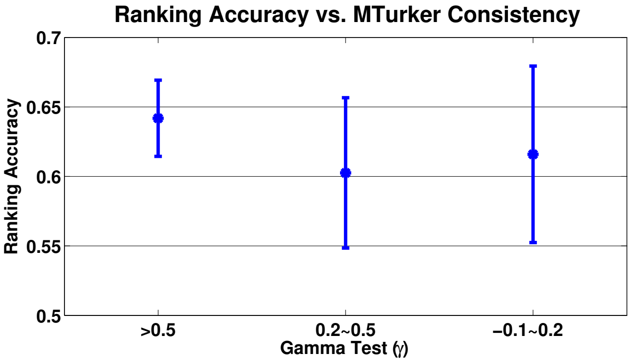

Figure 5. A comparison of ranking accuracy vs. the $\gamma$ measurement of the MTurkers pair of preference lists is shown. A $\gamma$ value of 0.0 indicates no association, i.e. lists are random with respect to each other, while a $\gamma$ value above 0.5 indicates good high association. This indicates the unreliability of the Gamma Test as a beauty consistency metric.

| **Feature Identifier** | **Description** |
|---|---|
| **(a)** | None/Random |
| **(b)** | Facial Geometric Features |
| **(c)** | Histogram of Oriented Gradients |
| **(d)** | GIST |
| **(e)** | L*a*b* Color Histograms |
| **(f)** | Eigenfaces |
| **(g)** | SIFT - Bag of Words |
| **(h)** | Features (c)+(d) |
| **(i)** | Features (b)+(c)+(d)+(e) |
| **(j)** | (b)+(c)+(d)+(f)+(g) |
| **(k)** | Features (b)+(c)+(d)+(e)+(f)+(g) |
| **(l)** | Dense-SIFT+PCA |
| **(m)** | Features (l)+(b)+(c)+(d)+(e) |

Table 1. The feature identifiers used in Figures 6, 7 , and 8 are shown here, and their respective descriptions.

a pseudo-random generator. By construction, those permutations should lack a beauty-trend. This showed as an average accuracy of less than 50% which matches how a random prediction approach would preform. Figure 7 shows how a sample of random permutations performs.

By observing how random permutations behaved, we decided to filter samples based on how the system responded. We removed the samples that gave accuracies close to random. We perceived this as our only measure to reject randomly permuted lists as provided by adversarial MTurkers. After filtration our sample was reduced to 44/60 preference lists. Figure 8 shows an average accuracy of 63% for feature types **(m)**, a slight difference of about 2% due to the removal of those randomly behaving samples.

The average accuracy of 63% was achieved by using a combination of Facial geometric features, GIST, HOG, L*a*b* color histograms, and Dense-SIFT + PCA features. The behavior of sole feature types appeared to vary as Figure 8 shows. The L*a*b* color histograms **(e)** alone did not carry any information that helps over random. SIFT alone,

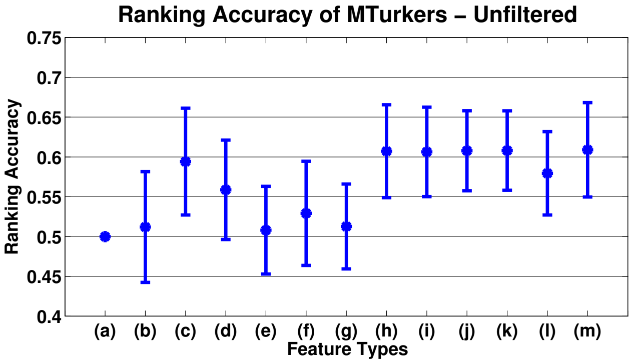

Figure 6. The average accuracies of running our ranking approach with the different feature combinations on the list of 60 preference lists.The feature combinations are those described in Table 1.

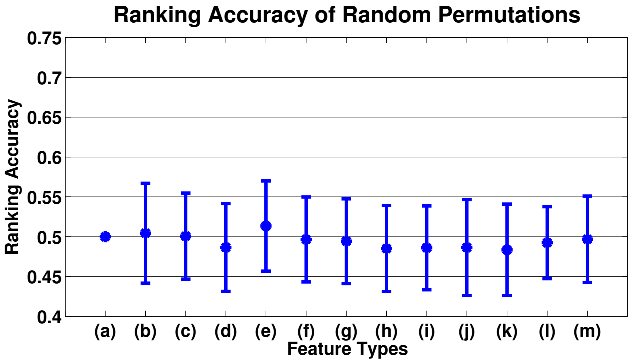

Figure 7. Feeding randomly permuted preference lists to the ranking system yielded random accuracy. This reveals that randomly permuted preferences lack beauty-trends. The feature combinations are those described in Table 1.

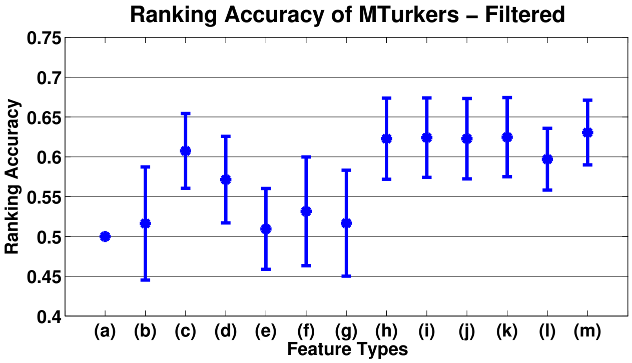

Figure 8. After filtering our sample to 44 preference lists by removing preference lists that exhibited random behavior, the average accuracies of running our ranking approach shows a slight increase with tighter error margins. The feature combinations are those described in Table 1.

as well, was not very successful, as it was influenced by the dimensionality of the codebook. Moreover, its combination with the other features did not offer any boosts **(k)**. It appears that the best predictors were HOG **(c)**, GIST **(d)** and Dense-SIFT + PCA **(l)**. The combination of HOG and GIST **(h)** added a small boost to the average accuracy as given by each of them separately. Dense-SIFT + PCA came in between HOG and GIST **(l)**, and its addition to the combination showed the best results with the narrowest error margin **(m)**. It appears that the additions of Facial geometric features, Color histograms, and Eigenfaces did not add any contribution to the mixture. This can be seen with **(i)** and **(k)**.

We believe that gradient data and Gabor filter responses served well in capturing the differences on which the individuals based their preferences. Moreover, the small difference in performance between **(i)** vs. **(m)** indicates that HOG might be a better and cheaper alternative to the computation of a Dense-SIFT across the whole image. Note that in **(m)**, we only added Dense-SIFT + PCA to the combination over **(i)**.

In order to verify significant dissimilarity between the different preference lists we conducted Gamma Test correlation measurements between all pairs of preference lists. We saw some positive correlation among them, 0.2 on average, which we interpreted as a general notion of beauty shared between the different individuals. This positive but not complete correlation suggests that our positive ranking accuracy is achieved by some understanding of what each individual found as attractive personally and differently.

The tests included 29, of which we only showed 13, combinations of features. The running time took approximately 40 seconds starting with data splitting through training and finally prediction. The experiments were performed in Matlab running on an Intel i5 2.67 GHz 64-bit processor with 4GB of memory.

## 5. Future Work

Jamieson and Nowak [17] proposed a sorting method that is optimized for objects that lie within a $d$-dimensional space. In their framework, the ranking reflects the object's distance from some reference point in that $d$-dimensional space. Now, since faces are viewed as high-dimensional objects, their method of $\Omega(d\, log_2\, n)$ comparisons might prove useful for an implementation of this paper on a large-scale website where the experience is delivered to the users.

An evaluation of the method's results with the MTurkers is an appealing experiment which we have not pursued. For example, some results could be returned to the MTurker to verify subjectively whether he/she approves of our automatically generated preference list of novel faces, and to which degree.

We also believe that there is a large margin for improvements on our current results. Attributes and high-level semantic information extracted from the faces might be more meaningful features for complex attributes such as *attractiveness*.

## 6. Conclusion

We introduced a ranking system for facial attractiveness that is able to rank novel faces according to personalized preferences, showing an average accuracy of 63%. To the best of our knowledge, our system is the first to establish a baseline on personalized facial attractiveness ranking. We showed that our system was able to identify certain beauty-trends which subjects revealed in their preferences as a form of consistency. On the other hand, random preference permutation failed to deliver such consistencies. The preferences we used were provided by different individuals and our Gamma Tests showed how their preferences were different, although some slight correlation existed, which further attests to our method's success.

## Acknowledgments

We would like to thank Kai Wang and Ana Murillo for their valuable comments and discussions. This work was supported by ONR MURI Grant #N00014-08-1-0638 and by the KACST Graduate Studies Scholarship. Portions of the research in this paper use the FERET database of facial images collected under the FERET program, sponsored by the DOD Counterdrug Technology Development Program Office.

## References

[1] P. Aarabi, D. Hughes, K. Mohajer, and M. Emami. The automatic measurement of facial beauty. In *Proceedings of ICSMC '01*, volume 4, pages 2644 –2647, 2001. 1

[2] K. Ali and W. van Stam. TiVo: making show recommendations using a distributed collaborative filtering architecture. In *Proceedings of SIGKDD '04*, pages 394–401, New York, NY, USA, 2004. ACM. 3

[3] A. Bosch, A. Zisserman, and X. Muñoz. Scene classification via pLSA. In *Proceedings of ECCV '06*, pages 517–530, 2006. 2

[4] A. Bottino and A. Laurentini. The analysis of facial beauty: an emerging area of research in pattern analysis. In *Proceedings of ICIAR '10*, pages 425–435, Berlin, Heidelberg, 2010. Springer-Verlag. 2

[5] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of ICML '07*, pages 129–136, New York, NY, USA, 2007. ACM. 3

[6] O. Chapelle. Training a support vector machine in the primal. *Neural Computing*, 19:1155–1178, May 2007. 3

[7] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings of CVPR '05*, pages 886–

893, Washington, DC, USA, 2005. IEEE Computer Society. 5

[8] Y. Eisenthal, G. Dror, and E. Ruppin. Facial attractiveness: Beauty and the machine. *Neural Computation*, 18(1):119–142, 2006. 2

[9] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *Proceedings of CVPR '09*, 2009. 1

[10] V. Ferrari and A. Zisserman. Learning visual attributes. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in NIPS 20*, pages 433–440. MIT Press, Cambridge, MA, 2008. 1

[11] R. Franklin-Jr. and R. Adams-Jr. The two sides of beauty: Laterality and the duality of facial attractiveness. *Brain and Cognition*, 72(2):300 – 305, 2010. 2

[12] C. Geng and X. Jiang. Face recognition using sift features. In *Proceedings of ICIP '09*, pages 3313 –3316, November 2009. 4

[13] T. Gevers and A. W. M. Smeulders. Pictoseek: combining color and shape invariant features for image retrieval. *IEEE Transactions on Image Processing*, 2000. 3

[14] L. A. Goodman and W. H. Kruskal. Measures of association for cross classifications. 49(268):732–764, December 1954. 5

[15] J. He, M. Li, H.-J. Zhang, H. Tong, and C. Zhang. Manifold-ranking based image retrieval. In *Proceedings of MULTIMEDIA '04*, pages 9–16, New York, NY, USA, 2004. ACM. 3

[16] R. Herbrich, T. Graepel, and K. Obermayer. *Large Margin Rank Boundaries for Ordinal Regression*. MIT Press, March 2000. 1, 3

[17] K. G. Jamieson and R. Nowak. Active ranking using pairwise comparisons. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, editors, *Advances in NIPS 24*, pages 2240–2248. 2011. 7

[18] T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of KDD '02*, pages 133–142, New York, NY, USA, 2002. ACM. 1, 3

[19] A. Kagian, G. Dror, T. Leyvand, D. Cohen-Or, and E. Ruppin. A humanlike predictor of facial attractiveness. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in NIPS 19*, pages 649–656. MIT Press, Cambridge, MA, 2007. 1

[20] Y. Ke and R. Sukthankar. PCA-SIFT: a more distinctive representation for local image descriptors. In *Proceedings of CVPR '04*, pages 506–513, Washington, DC, USA, 2004. IEEE Computer Society. 3

[21] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *Proceedings of ICCV '09*, 2009. 1, 2

[22] R. Kumar and S. Vassilvitskii. Generalized distances between rankings. In *Proceedings of WWW '10*, pages 571–580, New York, NY, USA, 2010. ACM. 2, 3, 5

[23] H. Li. A short introduction to learning to rank. *IEICE Transactions*, 94-D(10):1854–1862, 2011. 1, 3

[24] C. Liu, J. Yuen, A. Torralba, J. Sivic, and W. T. Freeman. Sift flow: Dense correspondence across different scenes. In *Proceedings of ECCV '08*, pages 28–42, Berlin, Heidelberg, 2008. Springer-Verlag. 3

[25] D. G. Lowe. Object recognition from local scale-invariant features. *Proceedings of ICCV '99*, 2:1150, 1999. 4

[26] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *IJCV '01*, 42:145–175, May 2001. 5

[27] D. Parikh and K. Grauman. Relative attributes. In *Proceedings of ICCV '11*, pages 503 –510, November 2011. 1, 3, 5

[28] S.-T. Park and D. M. Pennock. Applying collaborative filtering techniques to movie search for better ranking and browsing. In *Proceedings of SIGKDD '07*, pages 550–559, New York, NY, USA, 2007. ACM. 3

[29] P. Phillips, H. Moon, S. Rizvi, and P. Rauss. The feret evaluation methodology for face recognition algorithms. *IEEE Transactions on PAMI '00*, 22:1090–1104, 2000. 2

[30] P. Phillips, H. Wechsler, J. Huang, and P. Rauss. The feret database and evaluation procedure for face recognition algorithms. *Image and Vision Computing Journal*, 16(5):295–306, 1998. 2

[31] K. Schmid, D. Marx, and A. Samal. Computation of a face attractiveness index based on neoclassical canons, symmetry, and golden ratios. *Pattern Recognition*, 41(8):2710 – 2717, 2008. 2

[32] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on PAMI '00*, 22:1349–1380, December 2000. 3

[33] D. Sutic, I. Breskovic, R. Huic, and I. Jukic. Automatic evaluation of facial attractiveness. In *Proceedings of MIPRO '10*, pages 1339 –1342, May 2010. 2

[34] M. Thompson, P. Zimbardo, and G. Hutchinson. Consumers are having second thoughts about online dating (white paper). April 2005. Published by weAttract.com, `http://www.singleboersen-vergleich.de/dossier-partnervermittlung/us-stanford.pdf`. 1

[35] M. Turk and A. Pentland. Face recognition using eigenfaces. In *Proceedings of CVPR '91*, pages 586 –591, June 1991. 4

[36] J.-G. Wang, J. Li, C. Y. Lee, and W.-Y. Yau. Dense sift and gabor descriptors-based face representation with applications to gender recognition. In *Proceedings of ICARCV '10*, pages 1860–1864. IEEE, 2010. 3, 4

[37] J. Whitehill and J. Movellan. Personalized facial attractiveness prediction. In *Proceedings of FG '08*, pages 1 –7, September 2008. 1, 2

[38] X. Zhao, Y. Fu, H. Ning, Y. Liu, and T. Huang. Human pose regression through multiview visual fusion. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(7):957 –966, July 2010. 4

[39] Y. Zhou, D. Wilkinson, R. Schreiber, and R. Pan. Large-scale parallel collaborative filtering for the netflix prize. In *Proceedings of the 4th International Conference on Algorithmic Aspects in Information and Management, LNCS 5034*, pages 337–348. Springer, 2008. 3