Searching for People through Textual and Visual Attributes

Junior Fabian, Ramon Pires, Anderson Rocha
Institute of Computing
University of Campinas (Unicamp)
Campinas-SP, Brazil

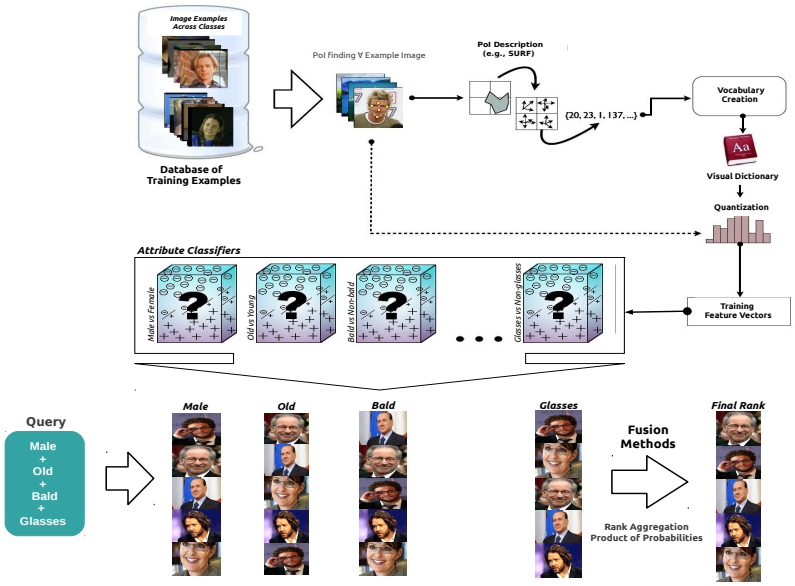


Fig. 1. The proposed approach aims at searching for people using textual and visual attributes. Given an image database of faces, we extract the points of interest (PoIs) to construct a visual dictionary that allow us to obtain the feature vectors by a quantization process (top). Then we train attribute classifiers to generate a score for each image (middle). Finally, given a textual query (e.g., male), we fusion obtained scores to return a unique final rank (bottom).

***Abstract*—Searching for people through their personal traits has been largely required for several areas and, consequently, has become the center of attention in the scientific community. Locating a suspect or finding missing people in a public space are some of the practical applications which take advantage of research conducted in this topic. In this paper, we propose the use of describable visual attributes (e.g, male, wear glasses, has beard), as labels that can be assigned to an image to describe its appearance. The approach is based on visual dictionaries to generate an intermediate representation for the face images. We train binary classifiers for the attributes which give to each image a score used to obtain its ranking. However, there are some attributes that have no immediate antagonistic (e.g., asian people). Then, we evaluate unary classifiers for such attributes. The method is easily extensible to new attributes. For queries consisting of more than one attribute, we use two approaches of the state-of-the-art to combine the rankings: product of probabilities and rank aggregation. Experimental results show that incorporating visual dictionaries improves the accuracy for some attributes. Furthermore, for many attributes, rank aggregation achieves better results than traditional methods of rank fusion. The proposed solution might be of interest in a forensic scenario for searching suspects in a database by means of textual descriptions provided by a victim.**

***Keywords*-Face Search; Rank Fusion; Visual Dictionaries**

I. Introduction

A large set of applications takes facial attributes to identify people. An example is in criminal investigation, when the police are interested in locating a suspect. In those cases, eyewitnesses typically fill out a suspect description form, where they indicate personal traits of the suspect as seen at the time when the crime was committed [1]. Based on that description, the police can manually scan the entire image and video archive looking for a person with similar characteristics. This search process has the disadvantage of being time consuming and often inaccurate.

Most state-of-the-art methods to date aim at solving the problem by extracting low-level features in the images [2], and applying such information to directly train classifiers for identification or detection [3]. In line with this, in this paper we propose to analyze the images with a unified intermediate representation for all associated textual descriptions.

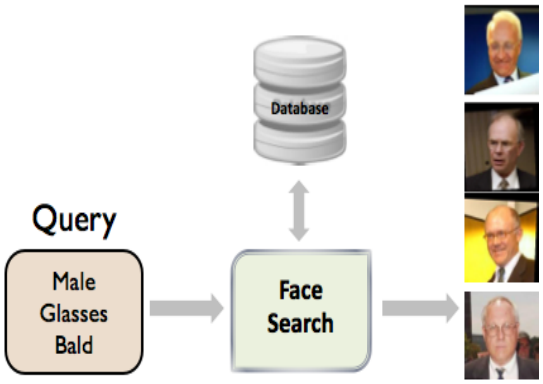


Fig. 2. Face Seach for a specific query $Q = \{male, glasses, bald\}$.

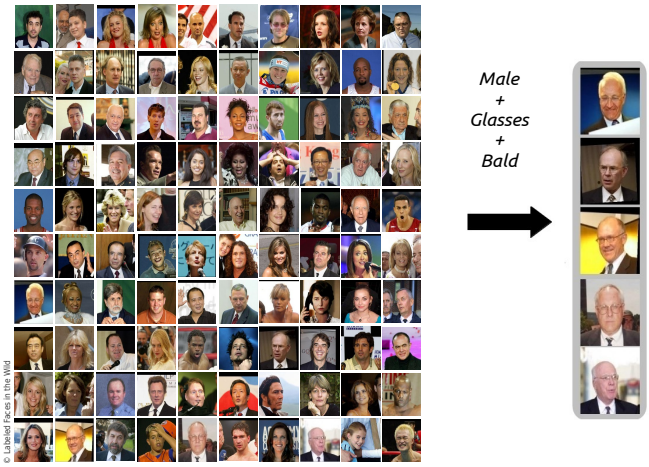


Fig. 3. Reducing the amount of images for a given query.

Our approach constructs visual dictionaries to represent important features of each facial attribute, an approach inspired in the current computer vision and image processing literature.

Most image search engines in the web are dependent on textual metadata. However, for the vast majority of images on the internet and in private collections, this attached data is often ambiguous, incorrect, or simply not present [3]. For a convenient face search, it is important to deploy a method able to automatically label the images with no need for any associated metadata.

Given a database of face images and a query composed of a set of attributes, which represent presence or absence of a visual trait (e.g., male with glasses and bald), our aim is at retrieving a subset of images from the database that satisfy each facial attribute contained in the query. The main challenge is to combine the ranking of different visual attributes for a final ranking which complies with the required attributes. Figure 2 depicts an example of the proposed approach.

Contributions: The novelty and contribution of this paper is in the new representation of low-level features for face characterization based on points of interest and in a common intermediary representation of such discriminative features using the concept of visual dictionaries. We also evaluate the use of unary classifiers to model visual attributes that have no immediate antagonistic (e.g., asian people) as opposed to features with direct antagonistic such as male/female. In addition, we investigate the sparse features characterization process before building the visual dictionaries. Finally, given a query with multiple attributes we use some fusion methods to combine the outputs of the classifiers generating a reduced list of people as Figure 3 depicts. By introducing visual dictionaries we achieve significant improvements on the results in comparison to the results obtained in the state-of-the-art [3].

II. Related work

Our work can be viewed as a form of Content-based Image Retrieval (CBIR), where the content is limited to face images and the queries are visual descriptions of the face (keywords). This section presents an overview of the relevant work in the literature.

Several works have been done regarding face characterization. Early work on appearance-based face verification [4] looked at the distance between pairs of images in a low dimensional subspace obtained using Principal Components Analysis (PCA). Variations in pose, expression, and lighting cause significant difficulties in the face verification task. To solve these problems, sometimes alignment, especially in 3D are used. Unfortunately, in a real-world scenario, 3D alignment is difficult (expensive) [3].

Prior work on attribute classification has focused mainly on ethnicity and on gender classification [5], [6]. First proposed in the computer vision community by Ferrari and Zisserman [7], visual attributes are text labels that can be automatically assigned to scenes, categories, or objects using standard machine learning techniques. Regarding combination of textual and visual features, the most similar work to ours is [3] and [8]. In [3], the authors explore direct image pixels intensity, edge magnitude, and edge orientation features with and without normalization for searching faces based on textual descriptions. For fusion, the authors use product of probabilities. In [8], the authors use soft biometric traits (scars, marks, and tattoos) to speed up face matching and assist in individualizing face searching tasks. The use of attributes has also been the object of research in the computer vision community [9], [10], [11].

Our method is similar to [3] and [8] in objectives but different in the sense we use a visual dictionary-based solution for characterizing the faces. To date, only binary classifiers are used to model the visual attributes. However, there are attributes that cannot be modeled using binary classifiers (e.g., asian people). To this end, we evaluate an unary classifier to such attributes and discuss if it is better to use one-class classifiers in such cases or a sampling of some negatives even though they do not represent direct and complete negatives of the trait in analysis.

Recently in [12], the authors propose techniques based on the statistical Extreme Value Theory to construct normalized

"multi-attribute spaces" from raw classifier outputs. Furthermore, they calibrate each raw score to a probability that the given attribute is present in the image. In this paper, we also normalize the classifier output and calibrate each score to a probability similar to [12].

We also use an alternative solution for the fusion called rank aggregation from the information retrieval literature [13] and compare it with the product of probabilities normally used in the literature of face attributes.

Regarding combination of output of classifiers, some works in the literature have used sum of scores [1], Borda Count [14], Rank Position [15] and the Condorcet Method [14]. Bayesian networks have also been explored for intelligent decision [16] as well as basic AND, OR, majority voting, and behavior knowledge space [17].

III. Proposed Technique

In order to solve the proposed problem, the first step of our approach is to extract "low-level" features related to the attribute of interest from different face regions. The features we use provide the representation of visual content of a given image through a set of Points of Interest (PoIs) in the image.

After extracting points of interest in the image, we compute an "intermediate-level" representation using visual dictionaries to preserve the distinctiveness power of the descriptors while increasing their generalization [18].

Given the set of 'words' of the visual dictionary, we summarize each image of the collection analyzing and assigning each of its PoIs to the closest visual word in the dictionary. This representation is used, in a third step, to train two-class classifiers which provide scores to each test image. Finally, to perform queries with several attributes, we implement two fusion approaches computed over the output scores of each individual attribute classifier. Our steps are formalized bellow and depicted in the teasing figure of our method in Figure 1.

1) **Low-level Feature Extraction:** For each face image in the database, extract the points of interest using SURF [2].
2) **Compute Intermediate-level Features:** Use a non-supervised learning technique over the PoIs to obtain k visual words ($\frac{k}{2}$ are positive and $\frac{k}{2}$ are negative) and create a visual dictionary representing each attribute of interest. This configuration is also a contribution with respect to the literature in which visual dictionaries are normally computed without considering the class information.
3) **Obtain the Attribute Classifier Scores:** Use a supervised learning technique to define the score for each test image I_i given a particular visual attribute of interest. Sort the images in decreasing order of relevance defined by such scores.
4) **Rank Fusion:** Given the set of attributes $A = \{a_1, a_2, ..., a_n\}$, their classifiers $C = \{c_1, c_2, ..., c_n\}$, and their outputs represented by ranks $R = \{r_1, r_2, ..., r_n\}$, where $r_k = \{i_1, i_2, ..., i_j\}$, $i_{f=1...j} \in$ set of images. Use the combining functions $F : Q \rightarrow R$ as described below to define the final rank R for a query $Q = \{a_p, ..., a_q\}$.

A. Low-level Features Extraction

Assuming the facial images are, at least, roughly aligned, we use an algorithm for extraction of points of interest to represent their visual content and to characterize their surrounding regions. It is desired to choose scale-invariant interest points in order to achieve a representation robust to some possible image transformations (e.g., rotations, scale, and partial occlusions). For this task, we use the well-known Speeded Up Robust Features (SURF) algorithm [2]. SURF algorithm has four major stages:

1) **Feature Point Detection:** In this stage, SURF uses an Hessian detector approximation and integral images [19] to speed up the involved operations.
2) **Feature Point Localization:** SURF uses the determinant of the Hessian for both location and scale. To localize the interest points in the image across different scales, the method performs nonmaximum suppression in a $3 \times 3 \times 3$ neighborhood. The determinant's maxima of the Hessian matrix are then interpolated in scale and image space.
3) **Orientation Assignment:** In order to be invariant to rotation, SURF calculates the Haar-wavelet responses for both x and y directions within a circular neighborhood of radius *6s* around the interest point, with $s = \sigma$ the scale at which the interest point was detected. The dominant orientation is estimated by calculating the sum of all responses within a sliding orientation window covering an angle of $\frac{\pi}{3}$ and the interest point gets the orientation of the longest calculated vector [2].
4) **PoI Characterization:** For the extraction of the descriptor, SURF constructs a square region centered around the interest point and oriented along the orientation selected in the previous stage. The region is split up regularly into smaller 4×4 square sub-regions and, for each sub-region, the method computes a Haar wavelet responses at 5×5 regularly-spaced sample points. Finally, the wavelet responses are summed up over each subregion and form a first set of entries to the feature vector [2].

We constrain the extraction of the points of interest according to regions of interest closely related to the attributes under consideration. Figure 4 depicts the regions of interest used for the feature selection on affine-aligned face images.

B. Intermediate-level Features

As we mentioned earlier, low-level features are not enough to represent facial images. When searching for a specific target, this discriminative power is extremely important.

Fig. 4. Regions of interest for each attribute. R_1: glasses. R_2: male and asian. R_3: bald. R_4: mustache. R_5: beard.

Notwithstanding, when searching for complex categories, it is a problem since the ability to generalize becomes paramount. As these solutions are often designed for exact matching, they do not translate directly into good results for image classification. In this sense, we can use the concept of visual vocabularies [18] to increase the descriptor generalization.

In the visual vocabulary construction, each set of PoIs becomes a visual 'word' of a 'dictionary'. Searching for *"people with mustache"*, for instance, in a database of images with faces, consists of selecting and creating a database of training examples comprising training positive examples (i.e., faces of people with mustache) and negative images (i.e., faces of people without mustache). The points of interest are calculated within the region of interest for the attribute 'mustache' (R4) as Figure 4 depicts.

After filtering the PoIs, we create a visual dictionary representing distinctive features of images for each specific attribute with K-Means. For this, we set $\frac{k}{2}$ words to represent the presence of the attribute and $\frac{k}{2}$ for the absence of such attribute.

C. Attribute Classifiers Scores

To perform the final classification procedure, we select a two-class machine learning classifier such as Support Vector Machines (SVMs) for all attributes. Furthermore, we evaluate an one-class machine learning classifier for attributes with no direct antagonistic. For training the two-class classifiers, we feed it with the signatures of the training images containing positive (e.g., images containing a specific attribute) and negative (images without the attribute) examples. And to train the one-class classifiers we use the signatures of the training images containing only positive examples.

After the learning stage, all of the images, except the ones contained in the training set, are classified yielding a classification score. Search results are ranked by confidence, so that the most relevant images are shown first. We use the computed distance to the classifier margin decision boundary as a measure of confidence similar to [3]. The images are then sorted in a decreasing order.

D. Rank Fusion

For searching with multiple query terms, we combine the confidence of different attribute classifiers such that the final ranking refers to images in decreasing order of relevance regarding the search terms. For instance, to solve a query such as "give me faces depicting a *male*, with *glasses* and *non-bald*", we fuse the scores given by each attribute classifier (a rank for *male*, a rank for *glasses* and a rank for *non-bald*) to produce a ranking based on the combination of the attributes. We considered two different fusion methods:

1) **Product of Probabilities:** Given a query $Q = \{a_p, ..., a_q\}$, this method consists in finding, for each image I_i in the database, its scores in each rank $r_p, ..., r_q$, and multiply the values [3]. The I_i's resulting scores are then sorted in decreasing order. To prevent high confidence for one attribute from dominating the search results, it is necessary to convert the confidences in probabilities. We transform each score s_i in a new score s'_i ensuring that the difference between s'_i and s'_{i+1} is equal to the difference between s'_{i+1} and s'_{i+2}. Figure 5 depicts an example of rank fusion using product of probabilities.

2) **Rank Aggregation:** Rank aggregation consists of taking m different rankings of n candidates (possibly given by different voters) and aggregating them in a single ranking. Kemeny [13] proposed an aggregation mechanism that produces the global ranking that minimizes the number of inverted pairs with the input rankings. The algorithm produces a *Footrule-optimal aggregation* that minimizes the sum of the differences of the ranks.

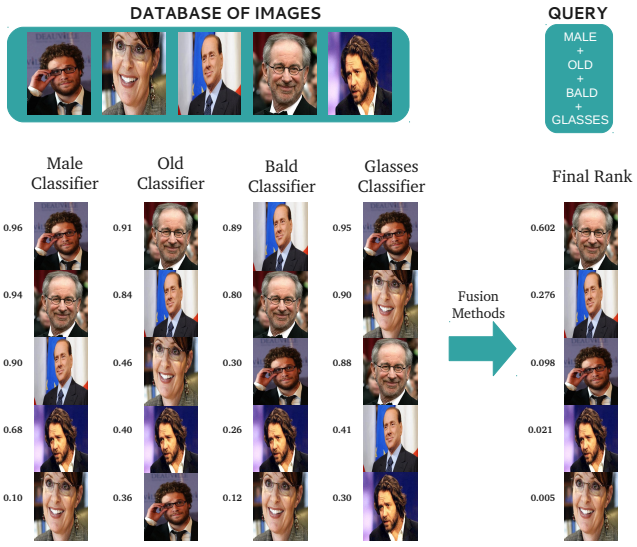


Fig. 5. Combing scores obtained by different classifiers for a given query using product of probabilities.

IV. Experiments and Discussion

In this section, we present the experiments we performed to validate the approach we discuss in this paper. We used the Labeled Faces in the Wild, a dataset of photographs with faces designed for unconstrained face recognition [20]. The dataset comprises 13,000+ face images. Here we used an LFW version whose images were aligned with funneling since it is not our purpose here to validate any face registration algorithm.

To validate the discussed method, we performed several searches with a subset of all possible combinations of the attributes we selected, and assessed the precision of retrieved images given a fixed recall. We designed the experiments in two rounds:

1) **Round 1:** evaluates individual attribute classifiers. We train two-class classifiers for all attributes and evaluate a one-class classifier for the attribute *asian* with training sets equally distributed (number of positive and negative examples for training is equal).
2) **Round 2:** evaluates the rank fusion for the selected queries by means of the fusion techniques previously explained.

In this work, we represent a query as a set of attributes $Q = \{a_p, ..., a_q\}$. We consider six attributes: *male*, *glasses*, *beard*, *mustache*, *asian* and *bald* and represent them respectively as *ma*, *gl*, *be*, *mu*, *as* and *ba*. The absence of an attribute is shown with an overline (e.g., $\overline{ma}$). For example, a query that contains *male*, non-*beard*, and *mustache* is represented by $Q = \{ma, \overline{be}, mu\}$. Finally, the fusion functions $F : Q \rightarrow R$ product of probabilities and rank aggregation are denoted respectively as $F_{product}$ and $F_{aggregation}$.

A. Round 1

In this round of experiments, we explore the importance of the number of words in the dictionary creation for each considered attribute classifier. We use three vocabulary sizes: 100, 500 and 1,000, where half of the words refers to the presence of the attribute and half to its absence. We select the best-performing dictionary for each attribute. For all attributes we consider, we use 1,000 training images and 500 testing images. We used SVM classifier with a radial basis kernel. The SVM parameters were calculated for each training set, using the standard LibSVM's grid search fine-tuning algorithm.

Figure 7 depicts the results with different dictionary sizes for each attribute. Additionally, Figure 7 shows the best result for the attribute *asian* using unary classifiers, while Table I shows the classification accuracy (#correctly classifications / #misclassifications) and the area under the Receiver Operating Curve (ROC) for each case. The scores are obtained by the distance to the SVM hyperplane.

Furthermore, for the attribute *asian* we evaluated an one-class SVM classifier with different kernels with 500 training images and 500 testing images. Figure 6 depicts the results obtained using a vocabulary size of 1,000 words for the attribute *asian*. Figure 6 also shows that unary classifiers achieves the best-performing when the kernel used is not linear.

TABLE I
Accuracy and AUC for each facial attribute.

Attribute	Accuracy	AUC	Number of Words
Male	81.6%	90.82%	1000
Glasses	80.4%	88.35%	500
Beard	79.0%	87.40%	500
Mustache	84.8%	91.65%	500
Asian	73.8%	81.37%	100
Unary-Asian	61.0%	66.33%	1000
Bald	83.4%	90.23%	100

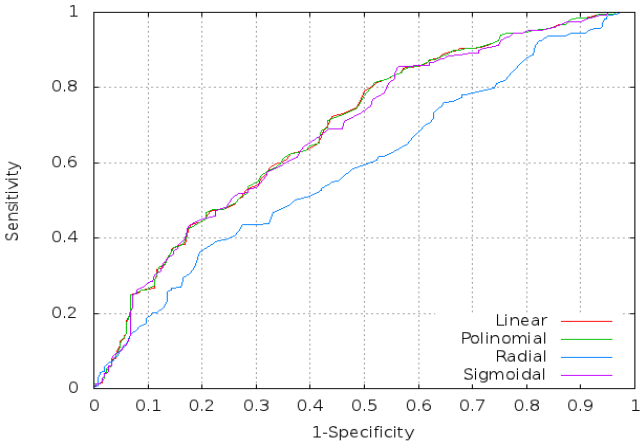


Fig. 6. ROC curves obtained for the attribute classifier of *asian* using a vocabulary size of 1,000 words and different kernels.

Figure 7 shows that by introducing visual dictionaries we achieve significant improvements on the results in comparison to the result obtained in the state-of-the-art[3]. Furthermore, we can note that the best-performing dictionary for each attribute has different vocabulary sizes. This is because the size of the regions in some attributes are different as Figure 4 depicts. Then attributes with large regions (e.g., male) has large variations, so it need more visual words to be represented. The attribute *asian* has best-performing using binary classifiers than using unary classifiers as Figure 7 depicts.

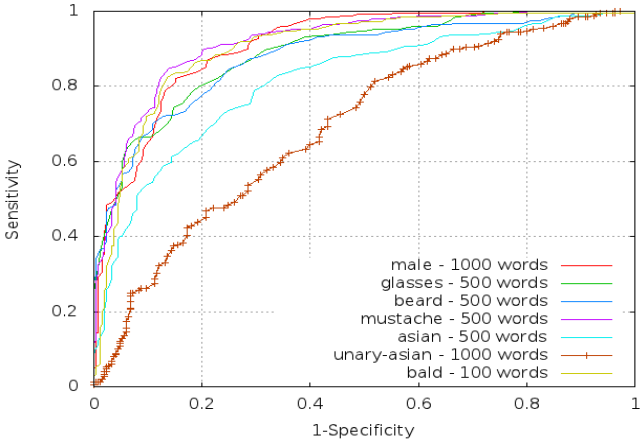


Fig. 7. ROC curves obtained for the attribute classifiers.

B. Round 2

This section shows results for rank fusion techniques. With k attributes, we have 2^k possible queries. Here, we present the results for a subset of all possible queries.

Before fusion, score normalization is a fundamental task when dealing with discrepant values related to different attribute classifiers. To overcome this problem, we normalize the scores using the traditional *z-norm* (subtract the overall mean score and divide by the overall standard deviation). To measure the effectiveness of each original rank and each rank resulting of a fusion, we assess the number of relevant images within the retrieved images for a fixed recall. Table II shows the precision for a subset of queries.

Simple queries (only one attribute), normally have high precision results as Table II shows. An exception is noted for the attribute *asian*, due to the fact that LFW does not have a significant number of images of asian people for training. Although the database contains an acceptable number of asian images, the number of asian people is small when we remove the training set part. However, as we would expect, the non-*asian* precision is 100%.

The rank aggregation method is not able to yield good results for searches with two attributes. This is because only one vote is enough to put an image in the resulting rank. The precision of rank aggregation for most of the queries with two attributes, is approximately half of the precision of the product of probabilities approach. In searches with more than two attributes, however, the results are promising. In some cases, like $\{\overline{gl}, \overline{mu}, \overline{be}, \overline{ba}\}$, rank aggregation presents a huge difference in comparison with the product of probabilities as Figure 8 depicts along with other complex queries.

In order to measure the precision of our approach we have analyzed the results returned in the top positions as can be seen in Figure 8. As a result of such analysis we have realized that the higher precision is obtained in the top 25 positions and it decreases as we analyze the next top positions. Maximizing the number of relevant results in the first positions represent an important advantage to ensures the quality of the retrieved results.

TABLE II
PRECISION OF SOME SELECTED QUERIES.

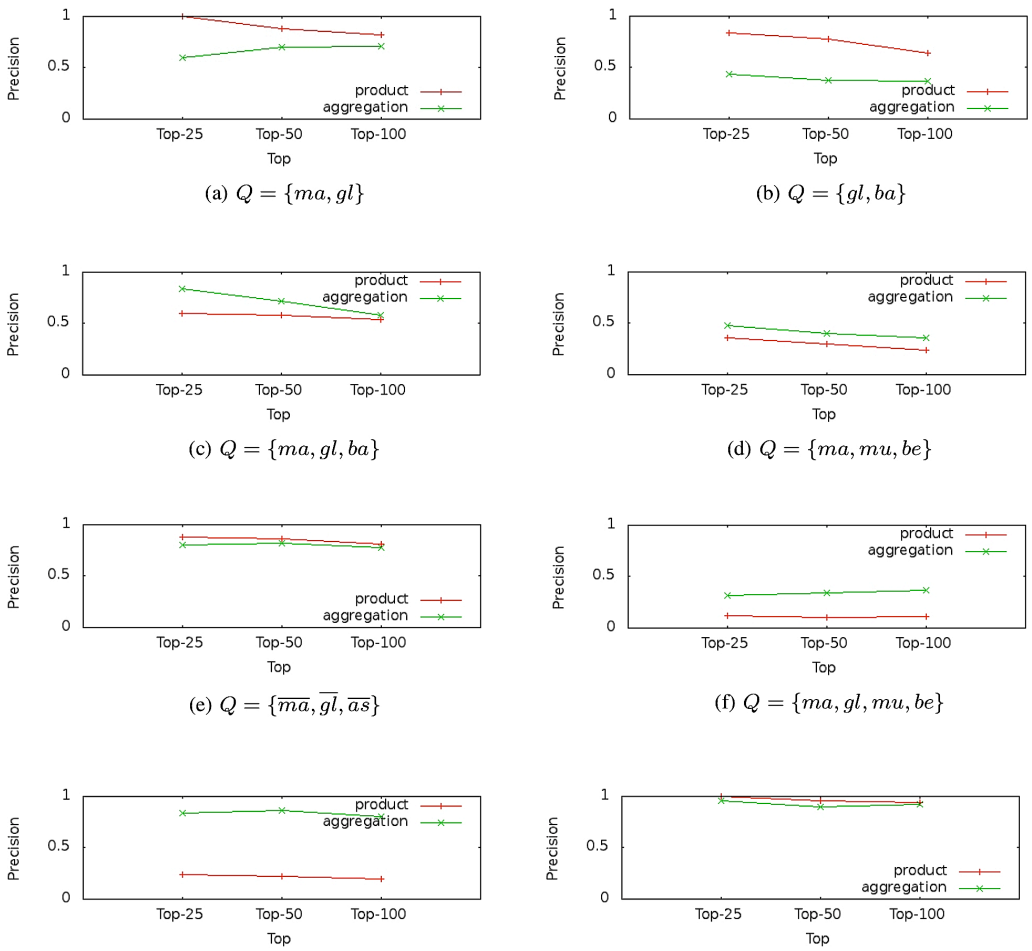
Q	Top-25		Top-50		Top-100	
	$F_{product}$	$F_{aggregation}$	$F_{product}$	$F_{aggregation}$	$F_{product}$	$F_{aggregation}$
$\{ma\}$	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
$\{gl\}$	100.0%	100.0%	100.0%	100.0%	99.0%	99.0%
$\{be\}$	72.0%	72.0%	56.0%	56.0%	52.0%	52.0%
$\{mu\}$	92.0%	92.0%	82.0%	82.0%	70.0%	70.0%
$\{as\}$	48.0%	48.0%	44.0%	44.0%	37.0%	37.0%
$\{ba\}$	84.0%	84.0%	74.0%	74.0%	75.0%	75.0%
$\{\overline{ma}\}$	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
$\{\overline{gl}\}$	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
$\{\overline{be}\}$	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
$\{\overline{mu}\}$	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
$\{\overline{as}\}$	100.0%	100.0%	98.0%	98.0%	99.0%	99.0%
$\{\overline{ba}\}$	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
$\{gl, as\}$	**48.0%**	32.0%	**40.0%**	34.0%	37.0%	37.0%
$\{gl, ba\}$	**84.0%**	44.0%	**78.0%**	38.0%	**64.0%**	37.0%
$\{ma, as\}$	**48.0%**	32.0%	**30.0%**	22.0%	**23.0%**	20.0%
$\{gl, be\}$	**44.0%**	24.0%	**38.0%**	18.0%	**25.0%**	14.0%
$\{ma, gl\}$	**100.0%**	60.0%	**88.0%**	70.0%	**82.0%**	71.0%
$\{mu, be\}$	64.0%	64.0%	56.0%	**64.0%**	50.0%	**53.0%**
$\{\overline{ma}, as\}$	**28.0%**	24.0%	**20.0%**	**18.0%**	**25.0%**	**14.0%**
$\{\overline{ma}, gl\}$	**40.0%**	4.0%	**28.0%**	2.0%	**22.0%**	2.0%
$\{\overline{ma}, \overline{gl}\}$	**100.0%**	92.0%	**98.0%**	84.0%	**95.0%**	82.0%
$\{mu, \overline{be}\}$	12.0%	**28.0%**	14.0%	**24.0%**	9.0%	**18.0%**
$\{ma, gl, ba\}$	60.0%	**84.0%**	58.0%	**72.0%**	54.0%	**58.0%**
$\{\overline{ma}, gl, as\}$	4.0%	4.0%	4.0%	**6.0%**	**4.0%**	3.0%
$\{\overline{ma}, gl, \overline{as}\}$	**16.0%**	4.0%	**14.0%**	4.0%	**14.0%**	6.0%
$\{\overline{ma}, \overline{gl}, as\}$	16.0%	16.0%	**14.0%**	12.0%	9.0%	9.0%
$\{\overline{ma}, \overline{gl}, \overline{as}\}$	**88.0%**	80.0%	**86.0%**	82.0%	**81.0%**	78.0%
$\{ma, gl, be\}$	8.0%	**20.0%**	10.0%	**14.0%**	10.0%	**15.0%**
$\{ma, gl, mu\}$	**28.0%**	24.0%	**26.0%**	22.0%	**24.0%**	20.0%
$\{ma, mu, as\}$	20.0%	20.0%	12.0%	**20.0%**	10.0%	**17.0%**
$\{ma, mu, be\}$	36.0%	**48.0%**	30.0%	**40.0%**	24.0%	**36.0%**
$\{ma, mu, \overline{be}\}$	**16.0%**	12.0%	**16.0%**	10.0%	**11.0%**	9.0%
$\{\overline{mu}, \overline{be}, as, \overline{ba}\}$	**20.0%**	8.0%	14.0%	**16.0%**	12.0%	**14.0%**
$\{gl, mu, be, \overline{as}\}$	24.0%	24.0%	16.0%	16.0%	**15.0%**	13.0%
$\{gl, \overline{mu}, \overline{be}, as\}$	**16.0%**	8.0%	**12.0%**	10.0%	13.0%	13.0%
$\{ma, gl, mu, be\}$	12.0%	12.0%	10.0%	**12.0%**	**11.0%**	7.0%
$\{ma, \overline{gl}, as, \overline{ba}\}$	**12.0%**	8.0%	8.0%	8.0%	6.0%	**8.0%**
$\{ma, gl, mu, \overline{ba}\}$	**24.0%**	12.0%	**20.0%**	12.0%	**17.0%**	8.0%
$\{ma, mu, \overline{be}, \overline{ba}\}$	4.0%	4.0%	6.0%	**8.0%**	5.0%	**9.0%**
$\{\overline{gl}, \overline{mu}, \overline{be}, \overline{ba}\}$	24.0%	**84.0%**	22.0%	**86.0%**	20.0%	**80.0%**
$\{ma, gl, mu, as, \overline{ba}\}$	8.0%	8.0%	8.0%	8.0%	9.0%	**10.0%**
$\{ma, gl, mu, be, \overline{ba}\}$	4.0%	**16.0%**	8.0%	**12.0%**	4.0%	**10.0%**
$\{gl, \overline{mu}, \overline{be}, as, \overline{ba}\}$	**20.0%**	4.0%	**14.0%**	4.0%	**10.0%**	7.0%
$\{\overline{gl}, \overline{mu}, \overline{be}, \overline{as}, \overline{ba}\}$	**100.0%**	96.0%	**96.0%**	90.0%	**94.0%**	92.0%
$\{ma, gl, mu, be, as, ba\}$	4.0%	4.0%	2.0%	2.0%	2.0%	2.0%
$\{ma, gl, \overline{mu}, \overline{be}, as, \overline{ba}\}$	**16.0%**	4.0%	**10.0%**	2.0%	**8.0%**	6.0%
$\{ma, \overline{gl}, mu, \overline{be}, \overline{as}, \overline{ba}\}$	4.0%	**8.0%**	6.0%	**8.0%**	5.0%	**7.0%**
$\{ma, \overline{gl}, \overline{mu}, \overline{be}, as, \overline{ba}\}$	**16.0%**	8.0%	**10.0%**	8.0%	7.0%	**8.0%**

V. CONCLUSION

In this work, we have shown how to automatically train visual feature classifiers and associate these features to text attributes allowing one to perform high-level queries to a database of images without using text annotations. These classifiers are learned using images from LFW database. We showed performance in line with recent attribute classifiers of the literature [3].

We demonstrated that the use of visual dictionaries is worthwhile to learn and represent features in a common and standard form. We have used two approaches from the state-of-the-art for rank fusion (product of probabilities and rank aggregation) using the attribute classifiers' outputs. We have built six attribute classifiers but the incorporation of classifiers for new attributes is straightforward.

Finally, we now aim at investigating other classifier fusion techniques to improve the results for even more complex queries. Furthermore, other normalization techniques may be used to reduce the effects of noise, improving the performance achieved by the visual dictionaries. Another future direction is to investigate techniques to measure the level of presence or absence of an attribute and be able to perform queries such as *"white male, partially bald with a high mustache"*.



(a) $Q = \{ma, gl\}$ (b) $Q = \{gl, ba\}$

(c) $Q = \{ma, gl, ba\}$ (d) $Q = \{ma, mu, be\}$

(e) $Q = \{\overline{ma}, \overline{gl}, \overline{as}\}$ (f) $Q = \{ma, gl, mu, be\}$

(g) $Q = \{\overline{gl}, \overline{mu}, \overline{be}, \overline{ba}\}$ (h) $Q = \{\overline{gl}, \overline{mu}, \overline{be}, \overline{as}, \overline{ba}\}$

Fig. 8. Product of Probabilities vs. Rank Aggregation approaches. Some selected queries.

ACKNOWLEDGMENT

We would like to thank Microsoft Research and the São Paulo Research Foundation (FAPESP) for the financial support. We also thank E. Valle for valuable suggestions.

REFERENCES

[1] A. Datta, R. Feris, and D. Vaquero, "Hierarchical ranking of facial attributes," in *F&G*, 2011.

[2] H. Bay, T. Tuytelaars, and L. V. Gool, "Surf: Speeded up robust features," in *ECCV*, 2006.

[3] N. Kumar, A. C. Berg, P. Belhumeur, and S. Nayar, "Describable visual attributes for face verification and image search," in *IEEE TPAMI*, October 2011.

[4] M. Turk and A. Pentland, "Face recognition using eigenfaces," in *IEEE CVPR*, 1991.

[5] G. W. Cottrell and J. Metcalfe, "Empath: face, emotion, and gender recognition using holons," in *NIPS*, 1990.

[6] B. Golomb, D. Lawrence, and T. Sejnowski, "Sexnet: a neural network identifies sex from human faces," in *NIPS*, 1990.

[7] V. Ferrari and A. Zisserman, "Learning visual attributes," in *NIPS*, December 2007.

[8] U. Park, S. Liao, B. Klare, J. Voss, and A. K. Jain, "Face finder: Filtering a large face database using scars, marks and tattoos," Michigan State Univ., Tech. Rep. TR11, 2011.

[9] N. Kumar, P. Belhumeur, and S. Nayar, "Facetracer: A search engine for collections of images with faces," in *ECCV*, 2008.

[10] N. Kumar, A. C. Berg, P. Belhumeur, and S. Nayar, "Attribute and simile classifiers for face verification," in *ICCV*, 2009.

[11] C. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *CVPR*, 2009.

[12] W. Scheirer, N. Kumar, P. N. Belhumeur, and T. E. Boult, "Multi-attribute spaces: Calibration for attribute fusion and similarity search," in *IEEE CVPR*, June 2012.

[13] J. Kemeny, "Mathematics without numbers," *Daedalus*, vol. 88, no. 4, pp. 577–591, 1959.

[14] F. Roberts, *Discrete Mathematical Models with Applications to Social, Biological, and Environmental Problems.* Prentice Hall, 1976.

[15] R. Nuray and F. Can, "Automatic ranking of information retrieval systems using data fusion," *Information Processing & Management*, vol. 42, no. 3, pp. 595–614, 2006.

[16] W. Scheirer, N. Kumar, K. Ricanek, T. Boult, and P. Belhumeur, "Fusing with context: a bayesian approach to combining descriptive attributes," in *IJCB*, 2011.

[17] L. Lam and C. Y. Suen, "Optimal combinations of pattern classifiers," *PRL*, vol. 16, no. 9, pp. 945–954, 1995.

[18] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *WSLCV*, 2004.

[19] P. Viola and M. Jones, "Robust real-time face detection," *IJCV*, vol. 57, pp. 137–154, 2004.

[20] G. Huang, M. Ramesh, T. Berg, and E. Learned-miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," 2007.