Scalable Face Image Retrieval Using Attribute-Enhanced Sparse Codewords

Bor-Chun Chen, Yan-Ying Chen, Yin-Hsi Kuo, and Winston H. Hsu, *Senior Member, IEEE*

***Abstract*—Photos with people (e.g., family, friends, celebrities, etc.) are the major interest of users. Thus, with the exponentially growing photos, large-scale content-based face image retrieval is an enabling technology for many emerging applications. In this work, we aim to utilize automatically detected human attributes that contain semantic cues of the face photos to improve content-based face retrieval by constructing semantic codewords for efficient large-scale face retrieval. By leveraging human attributes in a scalable and systematic framework, we propose two orthogonal methods named attribute-enhanced sparse coding and attribute-embedded inverted indexing to improve the face retrieval in the offline and online stages. We investigate the effectiveness of different attributes and vital factors essential for face retrieval. Experimenting on two public datasets, the results show that the proposed methods can achieve up to 43.5% relative improvement in MAP compared to the existing methods.**

***Index Terms*—Content-based image retrieval, face image, human attributes.**

I. Introduction

DUE to the popularity of digital devices and the rise of social network/photo sharing services (e.g., Facebook, Flickr), there are largely growing consumer photos available in our life. Among all those photos, a big percentage of them are photos with human faces (estimated more than 60%). The importance and the sheer amount of human face photos make manipulations (e.g., search and mining) of large-scale human face images a really important research problem and enable many real world applications [1], [2].

Our goal in this paper is to address one of the important and challenging problems – large-scale content-based face image retrieval. Given a query face image, content-based face image retrieval tries to find similar face images from a large image database. It is an enabling technology for many applications including automatic face annotation [2], crime investigation [3], etc.

Manuscript received June 21, 2012; revised September 06, 2012 and November 18, 2012; accepted November 20, 2012. Date of publication January 24, 2013; date of current version July 15, 2013. This work was supported in part by grants from the National Science Council of Taiwan, under Contracts NSC 101-2628-E-002-027-MY2, and Excellent Research Projects of National Taiwan University, AE00-00-05. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Cees Snoek.

B.-C. Chen and Y.-Y. Chen are with the Department of Computer Science and Information Engineering, National Taiwan University, Taipei 10617, Taiwan (e-mail: sirius42@cmlab.csie.ntu.edu.tw; yanying@cmlab.csie.ntu.edu.tw).

Y.-H. Kuo is with the Graduate Institute of Networking and Multimedia, National Taiwan University, Taipei 10617, Taiwan (e-mail: kuonini@cmlab.csie.ntu.edu.tw).

W. H. Hsu is with the Graduate Institute of Networking and Multimedia and the Department of Computer Science and Information Engineering, National Taiwan University, Taipei 10617, Taiwan (e-mail: winston@csie.ntu.edu.tw).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2013.2242460

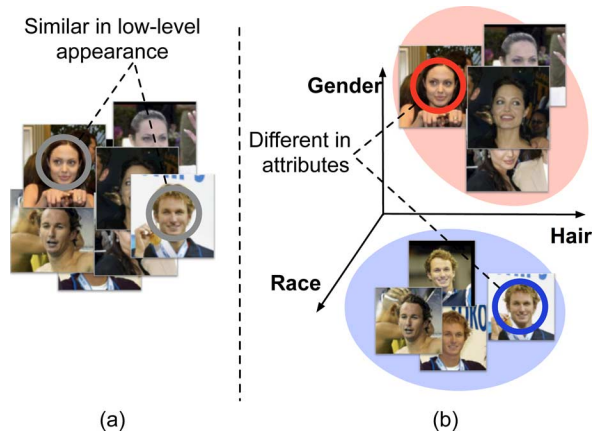

Fig. 1. (a) Because low-level features are lack of semantic meanings, face images of two different people might be close in the traditional low-level feature space. (b) By incorporating high-level human attributes (e.g., gender) into feature representations, we can provide better discriminability for face image retrieval. (Best seen in color).

Traditional methods for face image retrieval usually use low-level features to represent faces [2], [4], [5], but low-level features are lack of semantic meanings and face images usually have high intra-class variance (e.g., expression, posing), so the retrieval results are unsatisfactory (cf. Fig. 1(a)). To tackle this problem, Wu *et al.* [4] propose to use identity based quantization and Chen *et al.* [5] propose to use identity-constrained sparse coding, but these methods might require clean training data and massive human annotations.

In this work, we provide a new perspective on content-based face image retrieval by incorporating high-level human attributes into face image representation and index structure. As shown in Fig. 1, face images of different people might be very close in the low-level feature space. By combining low-level features with high-level human attributes, we are able to find better feature representations and achieve better retrieval results. The similar idea is proposed in [6] using fisher vectors with attributes for large-scale image retrieval, but they use early fusion to combine the attribute scores. Also, they do not take advantages of human attributes because their target is general image retrieval.

Human attributes (e.g., gender, race, hair style) are high-level semantic descriptions about a person. Some examples of human attributes can be found in Fig. 2(a). The recent work shows automatic attribute detection has adequate quality (more than 80% accuracy) [7] on many different human attributes. Using these human attributes, many researchers have achieved promising results in different applications such as face verification [7], face identification [8], keyword-based face image retrieval [9], and similar attribute search [10]. These results indicate the power of

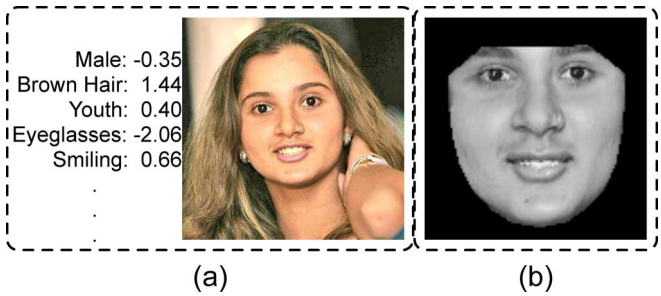

Fig. 2. (a) A face image contains rich context information (hair color, skin color, race, gender, etc.). Using automatic human attribute detection, we can describe them in high-level semantics; for example, *Male:* -0.35 suggests the person is unlikely a male, and *Brown Hair: 1.44* implies the hair color tends to be brown. (b) The same image after preprocessing steps in prior works for face image retrieval or recognition. They normalize the position and illumination differences between the faces and exclude background contents. Such common approaches sacrifice the important context information. Using automatically detected human attributes can compensate the information loss.

TABLE I
ENTROPY AND MUTUAL INFORMATION COMPUTED FROM TWO DIFFERENT DATASETS. X IS A RANDOM VARIABLE FOR THE IDENTITY OF A PERSON. Y IS THE ATTRIBUTE. THE CONDITIONAL ENTROPY, GIVEN THE ATTRIBUTE (E.G., GENDER), DROPS. IT SUGGESTS THAT USING HUMAN ATTRIBUTES CAN HELP IDENTIFY A PERSON

Dataset	$H(X)$	$H(X\|Y)$	$I(X;Y)$
LFW	11.21	10.45	0.77
Pubfig	5.43	4.46	0.97

the human attributes on face images. In Table I, we also show that human attributes can be helpful for identifying a person by the information-theoretic measures.

Although human attributes have been shown useful on applications related to face images, it is non-trivial to apply it in content-based face image retrieval task due to several reasons. First, human attributes only contain limited dimensions. When there are too many people in the dataset, it loses discriminability because certain people might have similar attributes. Second, human attributes are represented as a vector of floating points. It does not work well with developing large-scale indexing methods, and therefore it suffers from slow response and scalability issue when the data size is huge.

To leverage promising human attributes automatically detected by attribute detectors for improving content-based face image retrieval, we propose two orthogonal methods named *attribute-enhanced sparse coding* and *attribute-embedded inverted indexing*. Attribute-enhanced sparse coding exploits the global structure of feature space and uses several important human attributes combined with low-level features to construct semantic codewords in the offline stage. On the other hand, attribute-embedded inverted indexing locally considers human attributes of the designated query image in a binary signature and provides efficient retrieval in the online stage. By incorporating these two methods, we build a large-scale content-based face image retrieval system by taking advantages of both low-level (appearance) features and high-level (facial) semantics.

In order to evaluate the performance of the proposed methods, we conduct extensive experiments on two separate public datasets named LFW [11] and Pubfig [12]. These two datasets contain faces taken in unconstrained environment and are really challenging for content-based face image retrieval. Some examples of the datasets can be found in Fig. 6. During the experiments, we show that the proposed methods can leverage the context information from human attributes to achieve relative improvement up to 43.55% in mean average precision on face retrieval task compared to the existing methods using local binary pattern (LBP) [13] and sparse coding [5]. We also analyze the effectiveness of different human attributes across datasets and find informative human attributes.

To sum up, the contributions of this paper include:

- We combine automatically detected high-level human attributes and low-level features to construct semantic codewords. To the best of our knowledge, this is the first proposal of such combination for content-based face image retrieval.
- To balance global representations in image collections and locally embedded facial characteristics, we propose two orthogonal methods to utilize automatically detected human attributes to improve content-based face image retrieval under a scalable framework.
- We conduct extensive experiments and demonstrate the performances of the proposed methods on two separate public datasets and still ensure real time response.
- We further identify informative and generic human attributes for face image retrieval across different datasets. The selected descriptors are promising for other applications (e.g., face verification) as well.

The rest of the paper is organized as follows. Section II discusses related work. Section III describes our observations on the face image retrieval problem and the promising utilities of human attributes. Section IV introduces the proposed methods including attribute-enhanced sparse coding and attribute-embedded inverted indexing. Section V gives the experimental results, and Section VI concludes this paper.

II. RELATED WORK

This work is closely related to several different research topics, including content-based image retrieval (CBIR), human attribute detection, and content-based face image retrieval.

Traditional CBIR techniques use image content like color, texture and gradient to represent images. To deal with large-scale data, mainly two kinds of indexing systems are used. Many studies have leveraged inverted indexing [14] or hash-based indexing [15] combined with bag-of-word model (BoW) [16] and local features like SIFT [17], to achieve efficient similarity search. Although these methods can achieve high precision on rigid object retrieval, they suffer from low recall problem due to the semantic gap[18]. Recently, some researchers have focused on bridging the semantic gap by finding semantic image representations to improve the CBIR performance. [19] and [20] propose to use extra textual information to construct semantic codewords; [21] uses class labels for semantic hashing. The idea of this work is similar to the aforementioned methods, but instead of using extra information that might require intensive human annotations (and tagging), we try to exploit automatically detected human attributes to construct semantic codewords for the face image retrieval task.

Automatically detected human attributes have been shown promising in different applications recently. Kumar *et al.* propose a learning framework to automatically find describable

visual attributes [7]. Using automatically detected human attributes, they achieve excellent performance on keyword-based face image retrieval and face verification. Siddiquie *et al.* [9] further extend the framework to deal with multi-attribute queries for keyword-based face image retrieval. Scheirer *et al.* [8] propose a Bayesian network approach to utilize the human attributes for face identification. To further improve the quality of attributes, Parikh *et al.* propose to use relative attributes [22] and Scheirer *et al.* propose multi-attribute space [10] to normalize the confidence scores from different attribute detectors for similar attribute search. The works demonstrate the emerging opportunities for the human attributes but are not exploited to generate more semantic (scalable) codewords. Although these works achieve salient performance on keyword-based face image retrieval and face recognition, we propose to exploit effective ways to combine low-level features and automatically detected facial attributes for scalable face image retrieval. To the best of our knowledge, very few works aim to deal with this problem.

Due to the rise of photo sharing/social network services, there rises the strong needs for large-scale content-based face image retrieval. Content-based face image retrieval is closely related to face recognition problems but they focus on finding suitable feature representations for scalable indexing systems. Because face recognition usually requires substantial computation cost for dealing with high dimensional features and generating explicit classification models, it is non-trivial to directly apply it to face retrieval tasks. Meanwhile, the photo quality in consumer photos is more diverse and poses more visual variances. Wu *et al.* [4] propose a face retrieval framework using component-based local features with identity-based quantization to deal with scalability issues. To compensate the quantization loss, they further propose to use a state-of-the-art features [23] with principal component analysis for re-ranking. Wang *et al.* [2] propose an automatic face annotation framework based on content-based face image retrieval. In their framework, they adopt GIST [24] feature with locality sensitive hashing [15] for face image retrieval. Chen *et al.* [5] propose to use component-based local binary pattern (LBP) [13], a well known feature for face recognition, combined with sparse coding and partial identity information to construct semantic codewords for content-based face image retrieval.

Although images naturally have very high dimensional representations, those within the same class usually lie on a low dimensional subspace. Sparse coding can exploit the semantics of the data and achieve promising results in many different applications such as image classification and face recognition. Raina *et al.* propose a machine learning framework using unlabeled data with sparse coding for classification tasks [25], Yang *et al.* [26] apply the framework on SIFT descriptors along with spatial pyramid matching [27] and maximum pooling to improve classification results. Wright *et al.* [28] propose to use sparse representation for face recognition and achieve state-of-the-art performance.

Taking advantages of the effectiveness and simplicity of LBP feature with the superior characteristics of sparse coding on face images, we adopt a similar approach as Chen *et al.* used in [5]—using component-based LBP combined with sparse coding to construct sparse codewords for efficient content-based face image retrieval. However, instead of using identity information that might need manual annotations, we focus on utilizing automatically detected human attributes to construct semantic-aware sparse codewords using attribute-enhanced sparse coding. In addition, we propose another orthogonal approach to further leverage attribute information by constructing attribute-embedded inverted index in the online ranking stage. Note that the proposed methods can be easily combined with the method proposed in [5] to take advantage of both identity information and automatically detected human attributes. Also, low-level feature (i.e., LBP) can be replaced by other features such as T3HS2 descriptor used in [4].

III. Observations

When dealing with face images, prior works [2], [4], [5] usually crop only the facial region and normalize the face into the same position and illumination to reduce intra-class variance caused by poses and lighting variations. Doing these preprocessing steps, they ignore the rich semantic cues for a designated face such as skin color, gender, hair style. To illustrate, Fig. 2 shows the face image before and after the common preprocessing steps. After preprocessing steps, the information loss causes difficulty in identifying attributes (e.g., gender) of the face. In [7], the authors conducted human experiments to support similar points. When using a cropped version of face images, the face verification performance will drop comparing with using the original uncropped version. Interestingly, their experiments also show that human can achieve salient verification performance using only the surrounding context of face images. The experiments suggest that the surrounding context indeed contains important information for identifying a person. Therefore, we propose to use automatically detected human attributes to compensate the information loss.

Given a face image, let X be a random variable for the identity of a person, and Y is the attribute (e.g., gender). In information-theoretic perspective, knowing attributes can reduce the entropy for identifying a person and the information gain can be computed as,

$$I(X;Y) = H(X) - 284H(X|Y), \tag{1}$$

where $H(X)$ denotes the Shannon entropy of X, which is used to measure the uncertainty of the random variable X. $H(X|Y)$ is the conditional entropy of X given Y and shows the uncertainty of X after knowing the value of Y. Intuitively, the larger mutual information indicates more help coming from Y for predicting X. Table I shows the entropy and mutual information computed from two different public datasets using only gender as the human attribute. The detailed statistics of the datasets can be found in Table II. The probability of X is computed by the frequency of the person in the dataset. Genders of the people are manually labeled. We only consider gender in Y for simplicity. As a result, we gain up to 0.97 bit information, that is, considering the gender attribute allows us to skip nearly half of the database if the database contains 50% females and 50% males. Hence we hypothesize that using human attributes can help the face retrieval task.

IV. Proposed Method

In this section, we first describe the overview of our scalable content-based face image retrieval system, and then we ex-

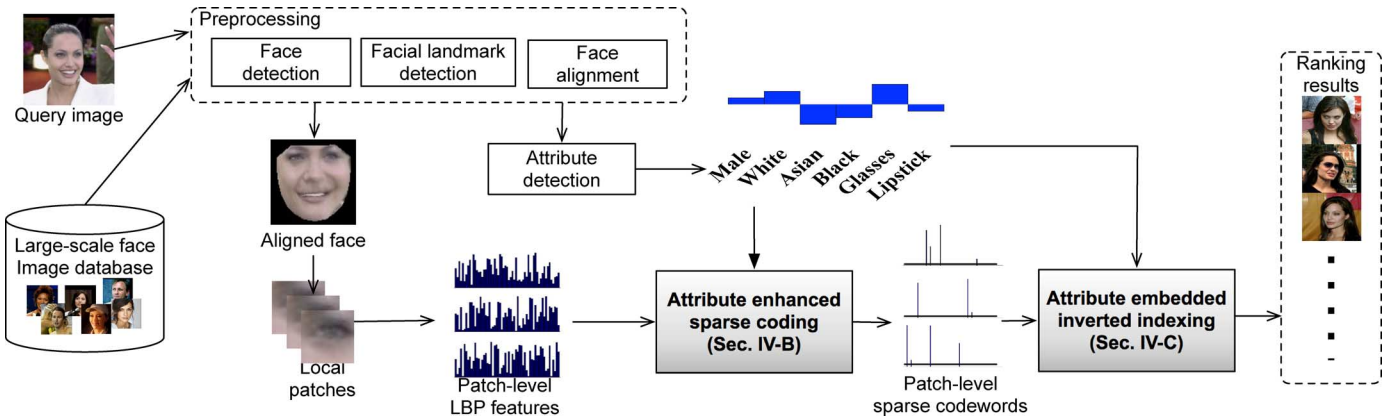

Fig. 3. The proposed system framework. Both query and database images will go through the same procedures including face detection, facial landmark detection, face alignment, attribute detection, and LBP feature extraction. Attribute-enhanced sparse coding is used to find sparse codewords of database images globally in the offline stage. Codewords of the query image are combined locally with binary attribute signature to traverse the attribute-embedded inverted index in the online stage and derive real-time ranking results over database images.

TABLE II
DATASET STATISTICS AND BASELINE PERFORMANCE. OVERALL PERFORMANCE IS WORSE ON LFW DATASET THAN ON PUBFIG, BECAUSE LFW CONTAINS MORE IMAGES AND MORE DIFFERENT PEOPLE. ATTR HAS THE HIGHEST MAP IN PUBFIG BUT LOWEST MAP IN LFW BECAUSE ATTR LOSES DISCRIMINABILITY WHEN THERE ARE TOO MANY PEOPLE IN THE DATASET. ATTR PERFORMS THE WORST IN TERMS OF P@10 BECAUSE IT ONLY HAS LIMITED DIMENSIONS. ALSO NOTE THAT LBP AND ATTR REQUIRE LINEAR SEARCH AND ARE NOT SCALABLE IN LARGE-SCALE DATASET. HOWEVER, WE SHOW HOW ATTRIBUTE-ENHANCED SPARSE CODEWORDS CAN HELP LARGE-SCALE FACE RETRIEVAL

	LFW			Pubfig		
# of people	5749			43		
Database size	13113			4300		
# of queries	120			430		
Performance	MAP	P@10	Time (s)	MAP	P@10	Time (s)
LBP	11.9%	**49.6%**	1.01	11.6%	47.4%	0.38
ATTR	11.6%	37.8%	0.04	**15.1%**	39.7%	0.01
SC	**13.0%**	46.8%	**0.03**	14.7%	**49.0%**	**0.01**
SC-original	10.0%	39.3%	1.73	10.9%	39.6%	0.59

plain the proposed methods: attribute-enhanced sparse coding and attribute-embedded inverted indexing in details. Note that the human attributes mentioned in the coming sections are automatically detected using the method described in [7].

A. System Overview

For every image in the database, we first apply Viola-Jones face detector [29] to find the locations of faces. We then use the framework proposed in [7] to find 73 different attribute scores. Active shape model [30] is applied to locate 68 different facial landmarks on the image. Using these facial landmarks, we apply barycentric coordinate based mapping process to align every face with the face mean shape [3]. For each detected facial component, we will extract 7×5 grids, where each grid is a square patch [4]. In total we have 175 grids from five components including two eyes, nose tip, and two mouth corners, on the aligned image using similar methods proposed in [4]. From each grid, we extract an image patch and compute a 59-dimensional uniform LBP feature descriptor as our local feature. After obtaining local feature descriptors, we quantize every descriptor into codewords using attribute-enhanced sparse coding described in Section IV-B. Attribute-embedded inverted index described in Section IV-C is then built for efficient retrieval. When a query image arrives, it will go through the same procedure to obtain sparse codewords and human attributes, and use these codewords with binary attribute signature to retrieve images in the index system. Fig. 3 illustrates the overview of our system.

B. Attribute-Enhanced Sparse Coding (ASC)

In this section, we first describe how to use sparse coding for face image retrieval. We then describe details of the proposed attribute-enhanced sparse coding. Note that in the following sections, we apply the same procedures to all patches in a single image to find different codewords and combine all these codewords together to represent the image.

1) Sparse Coding for Face Image Retrieval (SC): Using sparse coding for face image retrieval, we solve the following optimization problem:

$$\min_{D,V} \sum_{i=1}^{n} \|x^{(i)} - Dv^{(i)}\|_2^2 + \lambda \|v^{(i)}\|_1$$
$$\text{subject to } \|D_{*j}\|_2^2 = 1, \quad \forall j \quad (2)$$

where $x^{(i)}$ is the original features extracted from a patch of face image i, $D \in R^{d \times K}$ is a to-be-learned dictionary contains K centroids with d dimensions. $V = [v^{(1)}, v^{(2)}, \ldots, v^{(n)}]$ is the sparse representation of the image patches. The constraint on each column of D (D_{*j}) is to keep D from becoming arbitrarily large. Using sparse coding, a feature is a linear combination of the column vectors of the dictionary. [31] provides an efficient online algorithm for solving the above problem.

Note that the (2) actually contains two parts: dictionary learning (find D) and sparse feature encoding (find V). In [32], Coates *et al.* found that using randomly sampled image patches as dictionary can achieve similar performance as that by using learned dictionary ($< 2.7\%$ relative improvement in their experiments) if the sampled patches provide a set of overcomplete basis that can represent input data. Because learning dictionary with a large vocabulary is time-consuming (training 175 codebooks with 1600 dimension takes more than two weeks to finish), we can just use randomly sampled image patches as our dictionary and skip the time-consuming dictionary learning step by fixing D in the (2) and directly solve V. When D is fixed, the problem becomes a L1 regularized

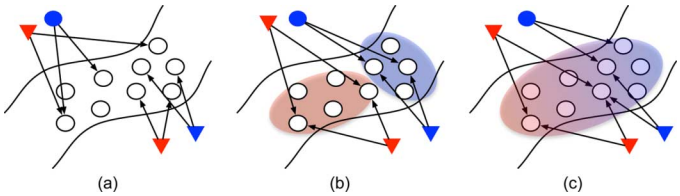

Fig. 4. Comparison between attribute enhancing coding methods: SC, ASC-D and ASC-W. Colors denote the attribute of the images (patches) and the shapes (e.g., triangle, circle) indicate the identity. (a) Because SC only considers low-level features, the images with the same attribute might be assigned to different dictionary centroids. (b) Using dictionary selection to force images with the same attribute being assigned to similar dictionary centroids. However, ASC-D uses attributes with hard assignment, when the detected attribute is wrong, the result will be undesired. In (b), blue triangle in the bottom-right corner is assigned to totally different codewords from the other two triangles. (c) Using attribute weights instead of inflexible dictionary selection, even when detected attributes are not perfect, it might still be able to retrieve correct images if appearance of the images is smiliar enough, because we use soft weights instead of hard assignment. (Best seen in color). (a) SC. (b) ASC-D. (c) ASC-W.

least square problem, and can be efficiently solved using LARS algorithm [33]. After finding $v^{(i)}$ for each image patch, we consider nonzero entries as codewords of image i and use them for inverted indexing. Note that we apply the above process to 175 different spatial grids separately, so codewords from different grids will never match. Accordingly, we can encode the important spatial information of faces into sparse coding. The choice of K is investigated in Section V-A. We use $K = 1600$ in the experiments, so the final vocabulary size of the index system will be $175 \times 1600 = 280,000$.

2) Attribute-Enhanced Sparse Coding (ASC): In order to consider human attributes in the sparse representation, we first propose to use dictionary selection (ASC-D) to force images with different attribute values to contain different codewords. For a single human attribute, as shown in Fig. 4(b), we divide dictionary centroids into two different subsets, images with positive attribute scores (blue ones in Fig. 4) will use one of the subset and images with negative attribute scores will use the other. For example, if an image has a positive male attribute score, it will use the first half of the dictionary centroids. If it has a negative male attribute score, it will use the second half of the dictionary centroids. By doing these, images with different attributes will surely have different codewords. For the cases of multiple attributes, we divide the sparse representation into multiple segments based on the number of attributes, and each segment of sparse representation is generated depending on single attribute. The above goal can be achieved by solving the following optimization problem modified from (2):

$$\min_{V} \sum_{i=1}^{n} \|x^{(i)} - Dv^{(i)}\|_2^2 + \lambda \|z^{(i)} \circ v^{(i)}\|_1$$

$$z_j^{(i)} = \begin{cases} \infty, & \text{if (1) } j \geq \left\lfloor \frac{K}{2} \right\rfloor \text{ and } f_a(i) \geq 0 \\ & \text{(2) } j < \left\lfloor \frac{K}{2} \right\rfloor \text{ and } f_a(i) < 0 \\ 1, & \text{otherwise,} \end{cases} \tag{3}$$

where "$\circ$" denotes the pairwise multiplication between two vectors, $f_a(i)$ is the attribute score for i_{th} image, and $z^{(i)}$ is a mask vector for deciding which codewords are allowed to be used by image i. By using the mask vector $z^{(i)}$, it forces the sparse representation $v_j^{(i)}$ to be zero if $z_j^{(i)}$ is ∞ because any other values in these dimensions will cause the objective function to become infinity. The final sparse representation $v^{(i)}$ can be found by solving a L1 regularized least square problem and only considering the dimensions where $z_j^{(i)} = 1$. Fig. 4(b) illustrates the above method. When an input image contains positive attribute value, the mask vector will be $[1, 1, \ldots, 1, \infty, \infty, \ldots, \infty]^T$, so only the first half of the sparse representation are likely to contain non-zero value for the image patches, and vice versa. To simplify the notation, the above equation only considers single attribute. For multiple attributes, we can simply define $z^{(i)}$ depending on multiple attributes. For example, if there are two attributes and image i contains positive scores for both attributes, $z^{(i)}$ will become $[1, \ldots, 1, \infty, \ldots, \infty, 1, \ldots, 1, \infty, \ldots, \infty]^T$. Although (3) can successfully encode the human attributes into the sparse representation, it has two main problems. First, it is not robust to possible attribute detection errors. Since our human attributes are automatically detected, they are not error-free. When the detected attribute is wrong, it will force images of the same person to be associated with totally different codewords (cf. Fig. 4(b)). Second, it only encodes the human attributes as binary indicators but we actually have relative confidence scores. Therefore, we further propose to integrate the relative scores of human attributes to relax (3) into a soft weighted version (ASC-W) by defining $z^{(i)}$ in (3) based on the attribute scores of images. Motivated by [34], we first assign a half of the dictionary centroids to have $+1$ attribute score and use them to represent images with the positive attribute; the other half of the dictionary centroids are assigned with -1 to represent images with the negative attribute. After the assignment, we can use the distance between attribute scores of the image and the attribute scores assigned to the dictionary centroids as the weights for selecting codewords. Because the weights are decided by attribute scores, two images with similar attribute scores will have similar weight vector, and therefore have a higher chance to be assigned with similar codewords and result in similar sparse representations. Fig. 4(c) illustrates the above method, images with similar attributes will be assigned with similar centroids, but images with erroneous attributes might still be able to retrieve correct images if their original features are similar (cf. Fig. 4(c) bottom-right blue triangle). In details, we first define an attribute vector $a \in \{1, -1\}^K$, where a_j contains the attribute scores of the j_{th} centroid as follows,

$$a_j = \begin{cases} +1, & \text{if } j \geq \left\lfloor \frac{K}{2} \right\rfloor \\ -1, & \text{otherwise,} \end{cases} \tag{4}$$

then we change the $z^{(i)}$ in (3) to become

$$z_j^{(i)} = \exp\left(\frac{d(f_a(i), a_j)}{\sigma}\right), \tag{5}$$

where $d(f_a(i), a_j)$ is the distance between the attribute score of the i_{th} image patch and that of the j_{th} dictionary centroid, and σ is used to adjust the decaying weights. To solve the above problem, we use the modified version of the LARS algorithm [33] by adjusting the weights according to $z_j^{(i)}$.

C. Attribute Embedded Inverted Indexing (AEI)

The methods described in Section IV-B aim to construct codewords enhanced by human attributes. In this section we describe the second method that can utilize human attributes by adjusting the inverted index structure.

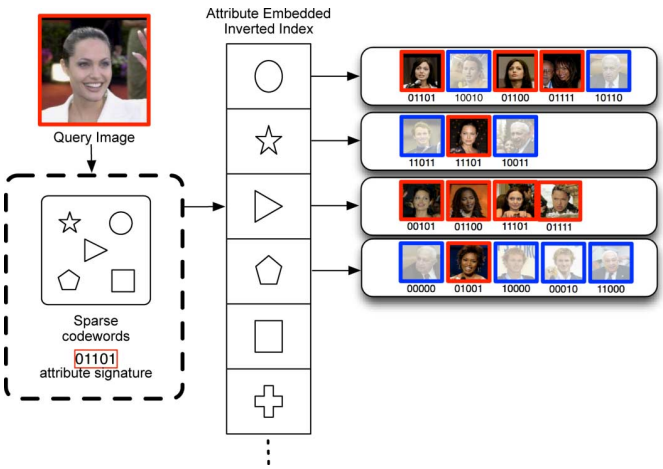

Fig. 5. Illustration of attribute-embedded inverted indexing. By considering binary attribute signature in the index system, we can skip images with large hamming distance in attribute hamming space and improve retrieval performance when traversing the codeword index in the online stage.

1) Image Ranking and Inverted Indexing: For each image, after computing the sparse representation using the method described in Section IV-B, we can use codeword set $c^{(i)}$ to represent it by taking non-zero entries in the sparse representation as codewords. The similarity between two images are then computed as follows,

$$S(i,j) = \|c^{(i)} \cap c^{(j)}\|. \tag{6}$$

The image ranking according to this similarity score can be efficiently found using inverted index structure.

2) Attribute-Embedded Inverted Indexing: To embed attribute information into index structure, for each image, in addition to sparse codewords $c^{(i)}$ computed from the facial appearance, we use a d_b dimension binary signature to represent its human attribute, $b^{(i)}$:

$$b_j^{(i)} = \begin{cases} 1 & \text{if } f_a^{(i)}(j) > 0 \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

The similarity score is then modified into,

$$S(i,j) = \begin{cases} \|c^{(i)} \cap c^{(j)}\| & \text{if } h(b^{(i)}, b^{(j)}) \leq T \\ 0 & \text{otherwise,} \end{cases} \tag{8}$$

where $h(i,j)$ denotes hamming distance between i and j, and T is a fixed threshold such that $0 \leq T \leq d_b$. As shown in Fig. 5, attribute-embedded inverted index is built using the original codewords and the binary attribute signatures associated with all database images. The image ranking according to (8) can still be efficiently computed using inverted index by simply doing a XOR operation to check the hamming distance before updating the similarity scores. As mentioned in [35], since XOR operation is faster than updating scores, by skipping images with high hamming distance in attribute hamming space, the overall retrieval time significantly decreases. Note that storage of the inverted index can be further compressed using many different techniques in information retrieval [14].

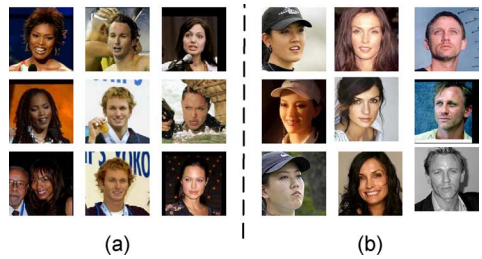

Fig. 6. Sample images from LFW (a) and Pubfig (b). Images in the same column are of the same person. These images contain large variances in expression, pose and illuminance which are very challenging for face retrieval.

V. Experiments

A. Experimental Setting

1) Datasets: We use two different public datasets (LFW [11] and Pubfig [12]) for the following experiments. LFW dataset contains 13,233 face images among 5,749 people, and 12 people have more than 50 images. We take 10 images from each of these 12 people as our query set (120 images) and all other images as our database (13,113 images). In Pubfig [12], we take 100 images from 43 people as our database images (4,300 images) and 10 images each from those 43 people as our query set (430 images). Example images from these two datasets can be found in Fig. 6. The facial attribute scores of Pubfig and LFW are provided by [12], which use pre-trained facial attribute detectors to measure 73 attribute scores. Note that the 73 attribute scores for these two datasets are also publicly available [12].

2) Compared Algorithms: We use several different baselines to compare with the proposed methods including two state-of-the-art face recognition features. The methods are described as follows: (1) *LBP*: concatenated 59-dimension uniform LBP [13] features computed from 175 local patches described in Section IV-A, i.e., totally 10325 dimensions; (2) *ATTR*: 73 dimensional human attributes computed by the method described in [7]; (3) *SC*: the sparse representation computed from LBP features using 1600 random samples as dictionary centroids combined with inverted indexing. Similar methods are used in [5]; (4) SC-original: similar to the feature extracted by (3), but we directly use the weight of sparse representation with linear search instead of using inverted index. (5) *ASC-D*: attribute-enhanced sparse representation with dictionary selection using the method described in Section IV-B2. (6) *ASC-W*: attribute-enhanced sparse representation with attribute weights using the method described in Section IV-B2. (7) *AEI*: attribute-embedded inverted indexing described in Section IV-C.

Note that although these datasets are widely used by many researches, most literatures use these datasets for research on face verification. In [4], they also use LFW dataset for face image retrieval. However, in their experiments, they need a (name) labeled training set besides the original dataset, so we are unable to reproduce their results. Also we notice that the results highly depend on the selection of query images—some of the query images can achieve more than 90% average precision (AP), some other query images can only achieve less than 5% AP. Since they do not release their experimental settings, we cannot directly compare with their work. Therefore we set some

state-of-the-art (conventional sparse coding and LBP) baselines in our own work.

Throughout the experiments we use mean average precision (MAP) and precision at K (P@K) as our performance metric. Note that for SC, ASC-D and ASC-W, we use random samples as dictionary centroids. We aim at a scalable framework for exponentially growing large-scale (and unseen) face photos rather than optimizing a specific benchmark. Therefore, in order to make sure that the performance does not overfit the specific dataset and our system can be generalized to other datasets, for experiments on LFW dataset, we use samples from Pubfig dataset and vice versa. The two datasets are designed with no overlapping in people, so there will be no images of the same person in dictionary centroids and database images.

3) Parameter Settings: In SC and ASC we need to decide parameter λ and dictionary size K. In order to decide the parameters, we run sensitive test and find out that the behavior is similar on both datasets. In the experiments, we run different dictionary sizes from 100 to 3200 and λ from 10^{-6} to 10^{-1}. When λ is too large ($\lambda = 10^{-1}$), almost all entries in sparse representation become zero, so the performance will drop. When λ and K are set properly ($\lambda = [10^{-6}, 10^{-2}], K = [100, 3200]$), the performance of SC is stable regardless the parameters on both datasets. For fair comparisons, we set $K = 1600$ on SC and ASC for both datasets. For SC, we use the best λ according to LFW for experiments on Pubfig and vice versa. Because the parameters are not sensitive to datasets, we only use one of the setting (the parameters decided by sensitivity test on Pubfig) for the other experiments. In ASC-W, we also need to decide the weight decay parameter σ. We fix $K = 1600$ and $\lambda = 10^{-2}$, run experiments using ASC-W with Male attribute and σ from 10 to 200 and set $\sigma = 120$ for all following experiments.

B. Baseline Performance

Table II shows the statistics of two datasets and the performance of four different baseline methods. In this work, we would like to highlight what improvements we can bring in as exploiting face attributes for semantic-rich sparse codeword representations. That is why we need to compare with LBP (one of the state-of-the-art low-level features), human attributes alone (ATTR), and the conventional sparse coding method. The setting in LBP (175 grids $\times 59$ dimensions) reaches the best performance in our experiments and it performs better than the original setting (49 grids $\times$ 59 dimensions) in [13] because it is more robust to pose variations. In addition, the 73 attributes have covered most of the general attributes discussed in the related work (e.g., [7]), which is the state-of-the-art of multiple facial attribute detection. ATTR obtains low P@10 in both datasets because it only has 73 dimensions and has very limited discriminability. In addition, ATTR obtains slightly higher MAP (15.1%) compared to SC-based approach (14.7%). However, SC has superior P@10 (49.0%) compared to ATTR (39.7%) because sparse codewords preserve the distinctive face traits of a specific person and benefit retrieving the faces with similar visual appearances. Therefore, we suggest that attribute-enhanced sparse codewords would further improve the accuracy of content-based face image retrieval. The experiments will be discussed in Section V-C.[1] The results of SC is consistent with the findings in [5]. Note that the performance in LFW dataset is slightly different from the number reported in [5]. There are two reasons for the performance difference. First, we randomly select different queries for evaluation, therefore, the overall MAP is slightly different. Second, the result reported in [5] is based on the parameters discovered within the single dataset; while the result in this paper is using the parameters determined by cross-validation manner over the two datasets.

We also compare with the SC-original to illustrate the superiority of inverted index. We find that SC-original for linear search takes average 1.73 seconds to retrieval LFW dataset with 13 K images while using SC for inverted index only takes about 0.03 seconds; the speedup is 57.7. Interestingly, we also find out that binarizing[2] SC actually improves the performance because it increases the discriminability between non-zero entries and zero entries of the feature vector. If we do not binarize the sparse representation, the number in each non-zero dimension is too small as comparing with the high dimension of the feature vector. Also, it is worth noting that the performance on Pubfig dataset is better than LFW dataset, it is also consistent with the entropy measure in Table I, because higher entropy indicates the task is harder and requires additional information.

C. Experiments on Attribute-Enhanced Sparse Coding

1) Performance of Single Attribute: Table III shows the MAP of attribute-enhanced sparse coding based on single attribute. We have computed the performance of ASC based on 73 different attributes on both datasets, due to the space limitation, we only show the results of top 10 and bottom 10 attributes ranked by ASC-W. We find that based on only single attribute, we can achieve up to 12.2% relative improvement in Pubfig dataset and 16.2% in LFW dataset using ASC-W. For some attributes, ASC-D performs worse than SC, because ASC-D is highly correlated with attribute detection accuracy. ASC-W performs better than ASC-D in most attributes, because ASC-W can take advantage of the relative attribute scores and is thus more robust to attribute detection error.

We also notice that using certain attributes (smiling, frowning, harsh lighting, etc.) will decrease the performance in both datasets. It is probably because these attributes are not correlated with the identity of the person. Informative human attributes across both datasets are also similar. Six out of the top ten attributes are overlapped in two datasets, while 15 out of the top 20 attributes are overlapped. To further demonstrate our findings, we compute the Spearman's rank correlation coefficient (ρ)[3] of two ranking lists and the coefficient is 0.89. It suggests the rankings from the two datasets are highly correlated, therefore the informative attributes are consistent across human photos.

2) Informative Attributes Discovery and Performance of Multiple Attributes: Using attributes ranked by ASC-W across two datasets, we are able to identify informative attributes. Here we categorize top 20 overlapped attributes into five different categories. Including gender related attributes (**G**):

[1]We also combine SC and ATTR using late fusion technique. The results of late fusion can achieve up to 17.3% relative improvement over SC.

[2]For sparse coding, only the important codewords will be assigned with non-zero value. Therefore, we assign these non-zero elements with 1 and the others with 0.

[3]Spearman's rank correlation coefficient is a measure used in statistics to compute the correlation between the ranked variables, it range from -1 (perfect negatively correlated) to 1 (perfect positively correlated).

TABLE III
MAP OF ATTRIBUTE-ENHANCED SPARSE CODING USING A SINGLE ATTRIBUTE. SOME ATTRIBUTES HAVE A NEGATIVE EFFECT ON PERFORMANCE BECAUSE THOSE ATTRIBUTES ARE NOT CORRELATED WITH THE IDENTITY. ASC-W PERFORMS BETTER IN MOST ATTRIBUTES BECAUSE IT CONSIDERS THE RELATIVE SCORES OF ATTRIBUTES AND IS ROBUST TO POSSIBLE ATTRIBUTE DETECTION ERROR. INTERESTINGLY, TOP RANKED AND BOTTOM RANKED ATTRIBUTES ARE SIMILAR ACROSS THE TWO DATASETS. SPEARMAN'S RANK CORRELATION COEFFICIENT BETWEEN TWO RANKING LIST USING ASC-W IS 0.89; IT INDICATES THAT ATTRIBUTE EFFECTIVENESS RANKED BY ASC-W IS VERY SIMILAR ACROSS TWO DATASETS

LFW			Pubfig		
Top 10	ASC-D	ASC-W	Top 10	ASC-D	ASC-W
Senior	12.5%	**15.1%**	Male	16.1%	**16.5%**
White	13.0%	14.9%	**Bushy Eyebrows**	14.5%	16.2%
Bald	11.4%	14.8%	**Receding Hairline**	14.3%	16.2%
Black Hair	13.2%	14.6%	**No Eyewear**	15.7%	16.1%
Gray Hair	10.1%	14.6%	**Black Hair**	13.4%	16.1%
Receding Hairline	12.0%	14.6%	Eyeglasses	15.9%	16.0%
Bushy Eyebrows	10.5%	14.5%	Wearing Lipstick	15.8%	16.0%
Youth	12.8%	14.4%	**Youth**	13.2%	15.9%
No Eyewear	12.7%	14.4%	Blond Hair	14.4%	15.9%
Wearing Earrings	13.1%	14.4%	**Bald**	13.6%	15.9%
Bottom 10	ASC-D	ASC-W	Bottom 10	ASC-D	ASC-W
Teeth Not Visible	10.1%	13.0%	**Harsh Lighting**	10.8%	13.7%
Flash	11.1%	12.9%	**Smiling**	11.8%	13.7%
Harsh Lighting	9.9%	12.8%	**Frowning**	11.8%	13.7%
Mouth Closed	9.0%	12.5%	Soft Lighting	8.9%	13.6%
Frowning	10.7%	12.4%	**Mouth Closed**	10.9%	13.6%
Smiling	10.5%	12.4%	Fully Visible Forehead	11.3%	13.5%
Rosy Cheeks	10.3%	12.2%	**Rosy Cheeks**	10.7%	12.9%
Flushed Face	10.9%	12.1%	**Flushed Face**	11.0%	12.4%
Pale Skin	8.6%	11.8%	**Pale Skin**	9.1%	12.0%
Color Photo	13.2%	11.0%	**Color Photo**	14.2%	10.9%

TABLE IV
PERFORMANCE OF ASC-W USING MULTIPLE ATTRIBUTES. WE CAN ACHIEVE UP TO 27.9% RELATIVE IMPROVEMENT IN LFW AND 26.9% IN PUBFIG USING TWO DIFFERENT ATTRIBUTES AGAINST THE BASELINES. USING THREE DIFFERENT ATTRIBUTES, WE CAN ACHIEVE UP TO 41.0% RELATIVE IMPROVEMENT IN LFW AND 36.9% IN PUBFIG

Attributes	LFW	Pubfig
A (Senior)/**H** (Gray Hair)	16.6%	17.3%
G (Male)/**P** (No Eyewear)	15.8%	18.7%
R (White)/**A** (Senior)/**P** (Bald)	**18.3%**	18.3%
G (Male)/**H** (Black Hair)/ **P** (No Eyewear)	17.3%	**20.2%**

Male, Wearing Earrings, Wearing Lipstick, Attractive Woman; hair colors (**H**): Blond Hair, Black Hair, Gray Hair; races (**R**): White, Asian, Indian; ages (**A**): Youth, Senior; and personal features (**P**): Receding Hairline, Bald, No Beard, Mustache, No Eyewear, Eyeglasses, Bushy Eyebrows, Double Chin. Combining these attributes using ASC-W, we can further achieve better performance. Table IV shows some example results of ASC-W which combines several attributes. We can achieve higher performance by using more than one attribute. By choosing two attributes from the above categories, the relative improvement can be up to 27.9% in LFW dataset and 26.9% in Pubfig dataset. When using three different attributes, we can achieve up to 41.0% and 36.9% relative improvement respectively.

Note that when more attributes are used, each attribute-enhanced patch is represented by less codewords if the number of feature dimensions is fixed. However, in our experiments, if we use up to 16 attributes, each attribute will be represented by a dictionary with size of 100 (for generating a 1,600 dimensional feature), which is still an over-complete basis compared to the dimension of the LBP feature (which is 59 dimensions) and therefore still has enough discriminative power for sparse feature representation. As suggested in [32], when using sparse coding as an encoder, the main contribution of the dictionary is to provide a over-complete basis. For more attributes, we need a larger dictionary with more codewords to represent the images; this is a trade-off between performance and online retrieval speed.

In our experiments, we show that the proposed methods can achieve salient performance (cf. Table IV) using only several informative human attributes while still maintaining the scalability of the system. We also find that using more attributes does not guarantee better performance since some attributes will have negative effects on the performance (e.g., Smiling, Mouth Closed). Therefore, we need to carefully select informative attributes identifying informative attributes complementary to the low-level face feature which is another contribution for this work.

D. Experiments on Attribute-Embedded Inverted Indexing

Fig. 7 shows the performance of attribute-embedded inverted indexing in Pubfig using different threshold values T in (8). When T is large, the performance will converge to SC because it ignores the attribute signature. When T is small, the performance will be improved, but when T is too small, the performance will drop dramatically. There are two possible reasons for this phenomenon. First, attribute detection error; when T is too small, the algorithm can not tolerate attribute detection error, so the performance will drop. Second, some attributes are not effective for identifying a person, and these attributes will cause the performance drop when T is too small. To manifest the second point, we run the same experiments on top 40 attributes ranked by the results of ASC-W. By removing some non-informative attributes, the performance can be further improved from

TABLE V
THE RESULTS COMBINED WITH ASC-W AND AEI. BY COMBINING TWO METHODS TOGETHER, WE CAN ACHIEVE UP TO 43.55% RELATIVE IMPROVEMENT IN LFW DATASET AND 42.39% IN PUBFIG DATASET

LFW	MAP	Relative improv.	Absolute improv.	P@10	P@20
SC	13.0%	-	-	46.8%	36.2%
ASC-W	18.3%	41.0%	5.3%	57.2%	45.3%
AEI	14.3%	10.5%	1.3%	49.0%	37.5%
ASC-W+AEI	**18.6%**	43.6%	5.6%	**57.3%**	**45.5%**
Pubfig	MAP	Relative improv.	Absolute improv.	P@10	P@20
SC	14.8%	-	-	49.0%	40.4%
ASC-W	20.2%	36.9%	5.4%	56.4%	47.7%
AEI	17.6%	19.3%	2.8%	51.4%	42.8%
ASC-W+AEI	**21.0%**	42.4%	6.2%	**56.9%**	**48.3%**

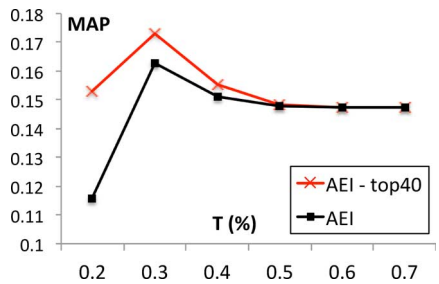

Fig. 7. The results of attribute-embedded inverted indexing in Pubfig using different threshold T in (8). When T is too large, we ignore the attribute signature so the performance will converge to SC. When T is too small, the performance will drop as it is sensitive to few noisy detectors. When using top 40 attributes ranked by ASC-W, the performance of attribute-embedded inverted indexing can be further improved.

16.3% to 17.6% in MAP (cf. Fig. 7). This result also indicates the effectiveness of the attributes ranked by ASC-W.

E. Combining ASC-W and AEI

ASC-W and AEI can be combined to further improve the retrieval performance. Table V shows the performance of combination results. By combining two methods, we can achieve up to 43.55% and 42.39% relative improvement over the original sparse coding. This is because ASC-W exploits the global structure of the feature space using only several informative attributes in the offline stage while AEI further considers locally to the designated query and just uses binary signature of all informative attributes in the online stage. Note that the performance gain is mainly contributed by ASC-W, but AEI is perfectly embedded into inverted index and is able to reduce the retrieval time. According to evidence reported in [35], such signatures can reduce the online retrieval time for 1 million images from 0.62 seconds to 0.20 seconds. Besides the overall MAP, precision on the top rank is also improved. Note that our MAP is much lower than P@10 and P@20, this is because the content-based face retrieval generally suffers from low recall rate problem due to the semantic gap. Although MAP is relatively low, the precision in top ranks is high enough for many applications; for instance, the system can be combined with method proposed in [2] for automatic face annotation.

F. Example Results

Fig. 8(a) and (b) shows two query examples using SC and ASC-W respectively. The red boxes indicate the false positives, and the number below each image is its rank in the retrieval results and the number in parentheses represents the rank predicted by SC. The improvement of ASC-W compared with SC is probably contributed by the attributes such as eye wears and hair colors. Note that even though some attributes are not always consistent with the same person (cf. Fig. 8(b) top 1 in SC), we can still retrieve the images using ASC-W. Fig. 8(c) and (d) show two examples using the proposed methods. Fig. 8(c) shows an example of occlusions. Using human attributes like hair colors we can gather information from not only face regions, therefore we can still achieve good performance under the occlusion. Fig. 8(d) shows a failure case, because the quality of the query image is poor, we cannot correctly predict the human attributes and sparse codewords, therefore the performance is not improved using ASC-W.

G. Scalability

In this section, we discuss the scalability of our system in two aspects, memory usage and online retrieval time.

1) Memory Usage: In our current implementation, each codeword only needs 16 bits to store its image ID in the index, and each image contains about 3,800 codewords on average. Total memory usage for inverted index with 13 K images is about 94.2 MB. We also need to store 40 bits attribute signature for each image (totally 0.1 MB for 13 K images). Therefore, total memory usage is about 94.3 MB for LFW dataset. For dataset with one million images, each codeword needs 20 bits to store its image ID; therefore, total memory usage for inverted index is about 9,060 MB. For attribute signatures, they take about 4.8 MB. Total memory usage is about 9,064.8 MB, which is a reasonable amount for a general computer server. Note that memory usage can be further reduced by many compression techniques in information retrieval [14] (e.g., reducing around half of memory by adopting d-gap technique).

2) Online Retrieval Time: Our system is implemented using C++ and operates on a 2.4 GHz Intel Xeon server. For a single query, face detection and alignment take about 0.7 seconds, LBP feature extraction takes about 0.06 seconds, computing sparse representation takes about 0.35 seconds, and retrieving index with 13 K images takes about 0.03 seconds. For attribute detection, as shown in [36], detecting a single attribute can be done within few milliseconds once the classifier is learned. For dataset with one million images, we refer to the results in [35]. In their reports, retrieving index with one million images takes 0.2 seconds. Since we have similar index structure with [35] (cf. Table VI), we expect that retrieving index with one million face photos can be done in less than one second.

Fig. 8. Example retrieval results using the proposed methods. The red boxes indicate the false positives, and the number below each image is its rank in the retrieval results and the number in a parenthesis represents the rank by SC only. Attributes used in ASC-W including G: Male, H: Blond Hair, and P: No Eyewear (a) The improvement is probably due to that P: No Eyewear is one of the attributes included in ASC-W. (b) Similarly as probably including H: Blond Hair. Note that even if some attributes are not consistent within the same person, we might still be able to retrieve the correct images (e.g., Top 1 in SC and Top 4 in ASC-W) as using appearance codewords and other attributes. (c) Even the query image has some occlusions, we can still find certain results because the codewords and attributes can capture additional information from the face region. (d) The failure case is due to the poor quality for the query image such that we are unable to correctly detect human attributes and reliable codewords. (Best seen in color).

TABLE VI
COMPARISON BETWEEN OUR INDEX STRUCTURE WITH THE ONE IN [35]. FOR A DATABASE CONTAINING ONE MILLION IMAGES, THE AVERAGE LENGTH OF POSTING LIST IN [35] IS ABOUT 15,000 WHILE OUR SYSTEM IS ABOUT 13,571. BASED ON THE RESULTS REPORTED IN [35] AND THE MEASURES BELOW, WE ARE ABLE TO RETRIEVE ONE MILLION FACE PHOTOS WITHIN ONE SECOND

	Baluja [35]	Our system
(a) Average # of codewords per image	3,000	3,800
(b) Vocabulary size	200,000	280,000
(c) Average length of posting list ((a) × 1,000,000 images ÷ (b))	15,000	13,571

VI. CONCLUSIONS

We propose and combine two orthogonal methods to utilize automatically detected human attributes to significantly improve content-based face image retrieval (up to 43% relatively in MAP). To the best of our knowledge, this is the first proposal of combining low-level features and automatically detected human attributes for content-based face image retrieval. Attribute-enhanced sparse coding exploits the global structure and uses several human attributes to construct semantic-aware codewords in the offline stage. Attribute-embedded inverted indexing further considers the local attribute signature of the query image and still ensures efficient retrieval in the online stage. The experimental results show that using the codewords generated by the proposed coding scheme, we can reduce the quantization error and achieve salient gains in face retrieval on two public datasets; the proposed indexing scheme can be easily integrated into inverted index, thus maintaining a scalable framework. During the experiments, we also discover certain informative attributes for face retrieval across different datasets and these attributes are also promising for other applications (e.g., face verification). Current methods treat all attributes as equal. We will investigate methods to dynamically decide the importance of the attributes and further exploit the contextual relationships between them.

REFERENCES

[1] Y.-H. Lei, Y.-Y. Chen, L. Iida, B.-C. Chen, H.-H. Su, and W. H. Hsu, "Photo search by face positions and facial attributes on touch devices," in *Proc. ACM Multimedia*, 2011.
[2] D. Wang, S. C. Hoi, Y. He, and J. Zhu, "Retrieval-based face annotation by weak label regularized local coordinate coding," in *Proc. ACM Multimedia*, 2011.
[3] U. Park and A. K. Jain, "Face matching and retrieval using soft biometrics," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 3, pp. 406–415, Sep 2010.
[4] Z. Wu, Q. Ke, J. Sun, and H.-Y. Shum, "Scalable face image retrieval with identity-based quantization and multi-reference re-ranking," in *Proc. IEEE Conf. Computer Vision and Pattern Recognit.*, 2010.
[5] B.-C. Chen, Y.-H. Kuo, Y.-Y. Chen, K.-Y. Chu, and W. Hsu, "Semi-supervised face image retrieval using sparse coding with identity constraint," in *Proc. ACM Multimedia*, 2011.

[6] M. Douze, A. Ramisa, and C. Schmid, "Combining attributes and fisher vectors for efficient image retrieval," in *Proc. IEEE Conf. Computer Vision and Pattern Recognit.*, 2011.
[7] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Describable visual attributes for face verification and image search," *IEEE Trans. Pattern Anal. Mach. Intell., Special Issue on Real-World Face Recognition*, vol. 33, no. 10, pp. 1962–1977, Oct. 2011.
[8] W. Scheirer, N. Kumar, K. Ricanek, T. E. Boult, and P. N. Belhumeur, "Fusing with context: A Bayesian approach to combining descriptive attributes," in *Proc. Int. Joint Conf. Biometrics*, 2011.
[9] B. Siddiquie, R. S. Feris, and L. S. Davis, "Image ranking and retrieval based on multi-attribute queries," in *Proc. IEEE Conf. Computer Vision and Pattern Recognit.*, 2011.
[10] W. Scheirer, N. Kumar, P. Belhumeur, and T. Boult, "Multi-attribute spaces: Calibration for attribute fusion and similarity search," in *Proc. IEEE Conf. Computer Vision and Pattern Recognit.*, 2012.
[11] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments, Univ. Massachusetts, Amherst, MA, USA, 2007, Tech. Rep. 07-49.
[12] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and simile classifiers for face verification," in *Proc. Int. Conf. Computer Vision*, 2009.
[13] T. Ahonen, A. Hadid, and M. Pietikainen, "Face recognition with local binary patterns," in *Proc. Eur. Conf. Computer Vision*, 2004.
[14] J. Zobel and A. Moffat, "Inverted files for text search engines," *ACM Comput. Surveys*, 2006.
[15] A. Gionis, P. Indyk, and R. Motwani, "Similarity search in high dimensions via hashing," in *Proc. VLDB*, 1999.
[16] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. Int. Conf. Computer Vision*, 2003.
[17] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, 2003.
[18] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman, "Total recall: Automatic query expansion with a generative feature model for object retrieval," in *Proc. IEEE Int. Conf. Computer Vision*, 2007.
[19] L. Wu, S. C. H. Hoi, and N. Yu, "Semantics-preserving bag-of-words models and applications," *IEEE Trans. Image Process.*, vol. 19, no. 7, pp. 1908–1920, Jul. 2010.
[20] Y.-H. Kuo, H.-T. Lin, W.-H. Cheng, Y.-H. Yang, and W. H. Hsu, "Unsupervised auxiliary visual words discovery for large-scale image object retrieval," in *Proc. IEEE Conf. Computer Vision and Pattern Recognit.*, 2011.
[21] J. Wang, S. Kumar, and S.-F. Chang, "Semi-supervised hashing for scalable image retrieval," in *Proc. IEEE Conf. Computer Vision and Pattern Recognit.*, 2010.
[22] D. Parikh and K. Grauman, "Relative attributes," in *Proc. IEEE Int. Conf. Computer Vision*, 2011.
[23] Z. Cao, Q. Yin, J. Sun, and X. Tang, "Face recognition with learning-based descriptor," in *Proc. IEEE Conf. Computer Vision and Pattern Recognit.*, 2010.
[24] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin, "Context-based vision system for place and object recognition," in *Proc. Int. Conf. Computer Vision*, 2003.
[25] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng, "Self-taught learning: Transfer learning from unlabeled data," in *Proc. ICML*, 2007.
[26] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Proc. IEEE Conf. Computer Vision and Pattern Recognit.*, 2009.
[27] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Conf. Computer Vision and Pattern Recognit.*, 2006.
[28] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.
[29] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Conf. Computer Vision and Pattern Recognit.*, 2001.
[30] S. Milborrow and F. Nicolls, "Locating facial features with an extended active shape model," in *Proc. Eur. Conf. Computer Vision*, 2008.
[31] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proc. ICML*, 2009.
[32] A. Coates and A. Y. Ng, "The importance of encoding versus training with sparse coding and vector quantization," in *Proc. ICML*, 2011.
[33] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Ann. Statist.*, 2004.
[34] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proc. IEEE Conf. Computer Vision and Pattern Recognit.*, 2010.
[35] H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *Proc. Eur. Conf. Computer Vision*, 2008.
[36] Shumeet Baluja and H. Rowley, "Boosting sex identification performance," *Int. J. Comput. Vision*, 2007.

Bor-Chun Chen received the M.S. degree from the Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, in 2012. His research interests include multimedia analysis, large scale image retrieval, and face image retrieval.

Yan-Ying Chen is currently working toward the Ph.D. degree from the Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan. Her research interests include face image analysis and multimedia data mining.

Yin-Hsi Kuo received the M.S. degree from the Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, in 2010. She is currently working toward the Ph.D. degree from the Graduate Institute of Networking and Multimedia, National Taiwan University. Her research interests include multimedia content analysis and image retrieval.

Winston H. Hsu (M'07–SM'12) received the Ph.D. degree in electrical engineering from Columbia University, New York, NY, USA.

He has been an Associate Professor in the Graduate Institute of Networking and Multimedia, National Taiwan University, Taipei, Taiwan, since February 2007. Prior to this, he was in the multimedia software industry for years. He is also with the Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan. His research interests include multimedia content analysis, image/video indexing and retrieval, machine learning, and mining over largescale databases.

Dr. Hsu serves in the Editorial Board for the IEEE Multimedia Magazine.