

# Discriminative Subgraphs for Discovering Family Photos

Changmin Choi<sup>1</sup>, YoonSeok Lee<sup>1</sup>, and Sung-Eui Yoon<sup>1</sup> (✉)

© The Author(s) 2015. This article is published with open access at Springerlink.com

**Abstract** We propose to use discriminative subgraphs to discover family photos from group photos in an efficient and effective way. Group photos are represented as face graphs by identifying social contexts such as age, gender, and face position. The previous work utilized bag-of-word models and considered frequent subgraphs from all group photos as features for classification. This approach, however, produces numerous subgraphs, resulting in high dimensions. Furthermore, some of them are not discriminative. To solve these issues, we adopt a state-of-the-art, frequent subgraph mining method that removes non-discriminative subgraphs. We also use TF-IDF normalization, which is more suitable for the bag-of-word model. To validate our method, we experiment in two data sets. Our method shows consistently better performance, higher accuracy in lower feature dimensions, compared to the previous method. We also integrate our method with the recent MS face recognition API and release it in a public website.

**Keywords** image classification, subgraph mining, social context, group photographs.

## 1 Introduction

Recent studies on image classification focus on object and scene classification. They show remarkable performance thanks to the improvement of image features such as convolutional neural network (CNN) [11]. These image features are built from pixel-level descriptors, and may not be enough to describe group photos, since classifying group



**Fig. 1** We test our method against a new, extended dataset consisting of non-family (a) and different family types (b) - (e). Our method achieves the highest accuracy, 79.34%, with 90 dimensions, while the state-of-the-art method achieves 76.8% with 1000 dimensions.

photos requires to utilize more semantic information like relations, events, or activities. Interestingly, humans can classify types (e.g., friends and family) of group photos without much training, because we can estimate a variety of social contexts such as age, gender, proximity, and place, by observing face, position, clothing, and other objects.

Once we identify the social context on group photos, we can use this information for various applications. One application is to control privacy of shared images in various social websites (e.g. Facebook). People share

<sup>1</sup> Korea Advanced Institute of Science and Technology (KAIST), 291 Daehak-ro, Yuseong-gu, Daejeon, Korea. E-mail: sungui@gmail.com (✉).

Manuscript received: 2016-00-00; accepted: 2016-00-00.



(a) Sending a query



(b) The result of the query

**Fig. 2** Our demo site using the proposed classification method.

images sometimes without much consideration on what information shared images can deliver to other peoples. When we identify that a shared group photo is a family photo containing children, we may wish to share that image to a small circle of persons, e.g., relatives, instead of publicly.

For classifying group photos, Chen et al. [2] proposed a method to categorize group photos into family and non-family types. This method assumes that annotations about age, gender, and face position are well-estimated beforehand by using existing face detection and statistical estimation derived from the pixel context. On top of that, they proposed to use a social-level feature named as Bag-of-Face-subGraph

(BoFG) to represent group photos by graphs. For constructing BoFGs, a mining algorithm extracting frequent subgraphs is adopted. This is based on the assumption that prominent social subgroups captured in group photos can be identified by looking at frequently appearing subgraphs.

While the prior method enlightens an interesting research direction of classifying group photos, it has certain drawbacks. It first requires a user-specified threshold to determine the number of feature dimensions in a training phase. Furthermore, as we have more frequent subgraphs by having more feature dimensions, we also raise the probability as a side effect that more non-discriminative subgraphs are selected due to repetitive and redundant patterns. In other words, thresholding the number of subgraphs with the frequency criterion alone can cause a scalability problem.

**Main contributions.** To overcome these issues, we survey the state-of-the-art subgraph mining techniques, and propose to use a subgraph mining technique, CORK, that identifies discriminative subgraphs and culls out redundant subgraph generations. We also propose to use a TF-IDF, a widely-used feature normalization for the bag-of-models, to our BoFG feature.

To validate benefits of our method in terms of classifying family and non-family types of group photos, we have tested the prior and our methods in two different datasets (Fig. 1) including the public dataset [8]. Overall, our method shows higher accuracy with less dimensions over the prior method. Furthermore, our method does not require a manually tuned threshold for computing dimensions of our BoFG features.

We have also integrated our method with the face API <sup>1</sup> of Microsoft Project Oxford and released it at our demo site <sup>2</sup>. In this system (Fig. 2), users can test their own group images and see how well our method performs with them.

## 2 Related work and background

We review prior approaches that are related to our method.

### 2.1 Social context in photographs

Social contexts contain various information such as clothing, age, gender, absolute or relative position, face

<sup>1</sup><https://www.projectoxford.ai/face/>

<sup>2</sup><http://is-fam.net/>

angle, gesture, body direction, and so on. They have been widely used to recognize people and groups [2, 14, 23]. Several works analyzed the contexts to study the structure of scenes in group photos [4, 7, 8]. Some researchers utilized them to classify group types [2, 5, 14, 17], retrieve similar group photos [3, 16, 26], discover social relations [18, 23], or predict occupations [19].

Pixel contexts in addition to the social contexts have been used together to recognize a type of group photos [14]. Some of well-known pixel-level features include SIFT [13], GIST [15], CNN [11], etc. Social-level features can be estimated by face detection, clothing segmentation, or partial body detection.

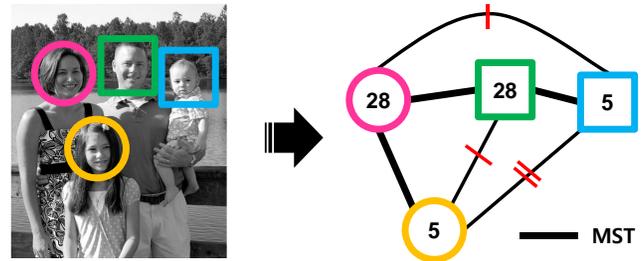
## 2.2 Frequent subgraph mining

Our work is based on identifying subgraphs from a graph representing the relationship between people shown in group photos. Frequently appearing subgraphs provide important cues on understanding graph structures and similarity between different graphs. As a result, mining frequent subgraphs has been widely studied [10]. For various classification, frequent subgraph mining has been used in training and test phases to build a social-level feature, as used in classifying family and non-family photo types [2].

We have found that extracted subgraphs significantly affect classification accuracy. There are two simple strategies to explore subgraphs in a database: (1) BFS-based and (2) DFS-based approach [10]. The BFS-based algorithm has been less used recently due to its technical challenges in generating candidates and pruning false positives. More advanced techniques focus on efficient candidate generation, since the subgraph isomorphism test is an NP-complete [6]. Recent successful algorithms proceed based on depth-first search and pattern growth [10], i.e., subgraph growing. Our method is also based on the DFS-based strategy, and uses canonical labels to avoid the scalability issue. We additionally measure the discriminative power of each subgraph during the pattern growth.

## 2.3 Graph-based image editing

In this work, we use graphs and histograms of their subgraphs for discovering family photos. Interestingly, there have been many graph based approaches for image extrapolation [24], interpolation [1], image segmentation [12], representations [9], etc. While these applications are not directly related to our classification problem, utilizing histograms of subgraphs could be useful in these applications, e.g., better graph matching



(a) Generating a group photo to a face graph

AGE RANGE	0-2	3-7	8-12	13-19	20-36	37-65	66+
FEMALE	1	5	10	16	28	51	75
MALE	1	5	10	16	28	51	75

(b) 14 types of vertices

	0	1	2	3
ORDER DISTANCE	—	—+	—++	—+++

(c) 4 types of edges

**Fig. 3** Representing an image as a face graph (a) using 14 vertex types (b) and 4 edge types (c).

for extrapolation.

## 3 Backgrounds on social subgraphs

In this section, we give the background of using BoFG features for group photo classification.

Chen et al. [2] proposed BoFG features for group photo classification. This method constructs face graphs (Fig. 3) and uses their subgraphs to describe various social relationships. BoFG is analogous to the bag-of-words model of text retrieval. For example, a text corpus corresponds to a group photo album, a document to an image, and a word to a subgraph in a face graph, respectively. The main difference between these models is that the bag-of-words model performs clustering over all vectors in order to obtain a codebook, whereas BoFG performs frequent subgraphs mining over all the face graphs.

Attributes of group members enable us to discriminate the type of groups, although we do not even know their names or relationships. In addition, understanding each one's position is informative to infer physical and relationship closeness among people. Chen et al. [2] showed that only knowing gender, age, and face positions as attributes of group members

works effectively for a binary classification of family and non-family photos. Our approach is also based on this approach, and represents a group photo into a face graph, elaborated below.

**Face graphs.** Fig. 3 illustrates an example of representing a group photo to a face graph. Each node of the graph corresponds to each person in the group photo, and is associated with a vertex label describing age and gender. Each edge between two nodes encodes relative positions between two people.

There are 14 different types describing age and gender for each vertex label. The age ranges from 0-year-old to 75-year-old, and is categorized into seven age types. There are two gender types, male and female, and they are visualized by square and circle, respectively in Fig. 3 (b). The combinations of age and gender result in 14 different types. Identifying faces and their attributes have been well studied [21, 27], and APIs of performing these operations are available, as mentioned in Sec. 1.

Most previous works used the Euclidean distance in image space, i.e., pixel distance, to measure the closeness between persons in group photos [8, 18, 23, 26]. Unfortunately, it has been known not to be invariant to scales of images, faces, distance to camera, or the orientation angle of a face. Instead, we use an order distance that indicates how close people stand each other. The order distance has been demonstrated to be more stable over the pixel distance in terms of various factors [2]. The order distance is computed as the path length among vertices on a minimum spanning tree (MST) generated from a face graph. Such order distance is used for each edge label such as Fig. 3 (c).

**Bag-of-Face-subGraph (BoFG).** Once we represent group photos into face graphs, we extract frequent subgraphs and regard them as BoFG features for classification. BoFG has been proposed to be a useful feature to compare structures of group photos. It helps to infer a type of a group by using substructures of groups. For example, in Fig. 3, edges between two vertices of  $28_f$  and  $28_m$  (i.e., mother-father relationship), and between  $28_f$  and  $5_m$  (i.e., mother and son relationship) provide additional information on social relationship over each node of those edges;  $f$  and  $m$  represent female and male gender types, respectively.

**Subgraph enumeration via gSpan.** The prior work regarded frequent subgraphs as BoFG features, and generated such subgraphs by frequent subgraph mining,

specifically, the gSpan method [25]. Most prior approaches of frequent subgraph mining [10] initially generate candidates of frequent subgraphs and adopt a pruning process to remove false positives. The pruning process, unfortunately, has a heavy computational cost, because it requires subgraph isomorphism testing.

gSpan adopted in the prior classification system [2] ameliorated this computational overhead issue by utilizing two techniques, DFS lexicographic order and minimal DFS code. Specifically, we first traverse an input graph,  $G$ , in a Depth-First Search (DFS) and assign an incrementally increasing visiting order to a newly visited vertex. Whenever we traverse an edge from  $v_m$  and to  $v_n$  of the graph  $G$ , we represent the traversed edge into a 5-tuple DFS code:

$$G = (m, n, L_m, L_{(m,n)}, L_n), \quad (1)$$

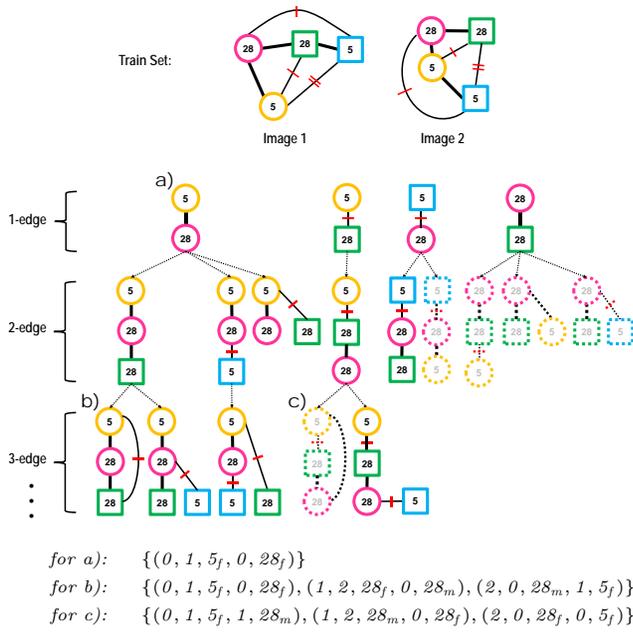
where  $m$  and  $n$  are vertex indices computed by the visiting ordering during the DFS traversal,  $L_m$  and  $L_n$  are vertex labels of  $v_m$  and  $v_n$ ,  $L_{(m,n)}$  is a edge label associated with the edge.

A graph, however, can have multiple DFS codes depending on traversal orders of vertices and edges. gSpan particularly allows the DFS lexicographic order computed from labels,  $L_m, L_{(m,n)}, L_n$ , of vertices and edges, and uses the DFS code corresponding to the minimal lexicographic order from the graph  $G$ . In this way, we can remove redundant subgraphs and maintain a subgraph among its isomorphic subgraphs.

To check the subgraph isomorphism, we simply look at the DFS code of a subgraph,  $G_S$ , to see whether the code is equal or bigger than ones generated by prior subgraphs. If so, this indicates that  $G_S$  is a redundant subgraph, which is isomorphic to a prior subgraph. An illustration of generating DFS codes and pruning process is shown in Fig. 4.

To define frequently appearing subgraphs, gSpan requires a user defined parameter, known as minimum frequency. We consider all different subgraphs whose frequency counts are bigger than the minimum frequency to be features of the BoFG.

The aforementioned method focuses on extracting frequency-based subgraphs and has some limitations for graph classification. Extracted frequent subgraphs in this approach may not show structural differences between classes. This is a similar problem even in the text classification. For instance, ‘a’ and ‘the’ are most commonly appearing words, but are not discriminative words for document classification. Moreover, the minimum frequency of subgraphs for defining BoFGs should be picked through a tedious trial-and-error approach for achieving high accuracy.



**Fig. 4** This figure shows a process of generating all the subgraphs having one edge or more in gSpan. During the enumeration of subgraphs, gSpan prunes subgraphs once their DFS codes are equal to or bigger than to prior ones. We highlight three subgraphs labelled a), b), and c), and their DFS codes in below.  $5_f$  is a 5-year-old female, while  $5_m$  is a 5-year-old male. Let the lexicographic orders of vertex and edge be  $5_f < 5_m < 28_f < 28_m$  and  $0 < 1 < 2$ . Note that the subgraphs b) and c) are isomorphic to each other. However, the subgraph of c) is not a minimal DFS code because it is bigger than that of b). In this manner, the search space can be pruned; the dotted subgraphs are pruned during the DFS-based expansion.

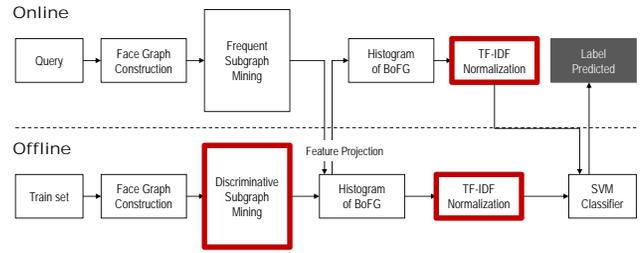
To address these drawbacks of using frequently appearing subgraphs, we propose to use discriminative subgraphs, adopt a recent subgraph mining method, CORK [22], extracting such discriminative subgraphs, and apply it to our classification problem of group photos. Additionally, we further improve the classification accuracy by adopting and tailoring the TF-IDF normalization scheme to our problem.

## 4 Our approach

In this section, we explain our approach for classifying group photos into family and non-family types.

### 4.1 Overview

Fig. 5 shows the overview of our method. As offline process, we first generate face graphs from group photos in a training set and extract discriminative subgraphs as Bag-of-Face-subGraph (BoFG) features from face graphs. We utilize family and non-family labels associated with training images. We then extract



**Fig. 5** The overview of our approach. The red-box indicates the main contributions of our method.

a BoFG feature for each photo and normalize the feature using by the TF-IDF weighting. Through discriminative learning, we finally construct a SVM classifier.

When a query image is provided, we represent it to a face graph, and extract and normalize a BoFG feature from the graph. We then estimate a query’s label by utilizing the pre-trained classifier.

Our work adopts face graphs and their subgraphs as the BoFG features for the classification problem (Sec. 3). For achieving higher accuracy in an efficient manner, we additionally propose using discriminative subgraphs (Sec. 4.2) inspired by a recent near-optimal selection method [22]. We also normalize BoFG features using the term frequency and inverse document frequency, i.e., the TF-IDF weighting scheme (Sec. 4.3).

### 4.2 Discriminative Subgraphs Mining

We would like to identify discriminative subgraphs that are characteristic features in each category. We have identified similar issues from data mining, and found that CORK [22] works well for our problem.

CORK considers statistical significance to select discriminative subgraphs. It defines a new measurement counting the number of features that are not helpful for classification among candidate features. This measurement can be integrated into gSpan as a culling method. It can reduce the number of features, while preserving performance in classification and can prune search space without relying upon a manually-tuned frequency threshold.

A near-optimality of CORK is obtained from a submodular quality function,  $q(\cdot)$ , using a greedy forward feature selection. The function  $q(\cdot)$  considers presence or absence of each subgraph in each class.  $q(\cdot)$  for the set containing subgraph,  $S$ , is defined as the following:

$$q(\{S\}) = -(A_{S_0} \cdot B_{S_0} + A_{S_1} \cdot B_{S_1}), \quad (2)$$

where  $A$  and  $B$  are two classes in a data set.  $A_{S_0}$  is

		data set					
		class A			class B		
		$a_1$	$a_2$	$a_3$	$b_1$	$b_2$	$b_3$
subgraphs	$S$	1	1	1	1	1	0
	$T$	1	1	1	0	0	0

**Fig. 6** A and B are two different classes in a given dataset.  $a_{1\sim 3}$  and  $b_{1\sim 3}$  are images in class A and B, respectively. Each indicator is 1, if its corresponding subgraph appears in each image, otherwise 0. Referred to Eq. 2,  $q(\{S\}) = -(0 \cdot 1 + 3 \cdot 2) = -6$  and  $q(\{T\}) = -(0 \cdot 3 + 3 \cdot 0) = 0$ . As a result, the subgraph  $T$  has a higher discriminative power than  $S$ .

the number of images of the class  $A$  that do not have the subgraph set  $\{S\}$ .  $A_{S_1}$  is the number of images including the subgraph set  $\{S\}$  in the class  $A$ . The subscript  $S_0$  and  $S_1$  are used in the same manner for another class  $B$ .

When a subgraph appears or does not appear simultaneously in both classes, it can be considered as a non-discriminative feature between two classes. To consider this observation,  $A_{S_0}$  and  $B_{S_0}$  are multiplied together; the same reasoning applies for the product of  $A_{S_1}$  and  $B_{S_1}$ . In this context, a feature becomes more discriminative, as the quality function  $q(\cdot)$  becomes higher. Fig. 6 shows examples of the quality function for two subgraphs in class  $A$  and  $B$ .

While generating subgraphs, we commonly expand a subgraph  $S$  into another one,  $T$ , by adding a neighboring edge or so. During this incremental process, suppose that we already decided to include  $S$  into our feature set. We then need to check the quality of having a newly expanded feature  $T$  on top of  $S$ . As a result, we need to reevaluate  $q(\{T\})$ .

Unfortunately, this process can require an excessive amount of running time, since as the number of features increases to  $N$ , the number of possible feature combinations can increase exponentially to  $2^N$ .

To accelerate this process, CORK relies on a pruning criterion. Especially, the upper bound of the quality function is derived based on three possible cases, when we consider a supergraph  $T$  from its subgraph  $S$ . One of such cases is that images from class  $A$  do not have the supergraph  $T$ , while images in the other class have the supergraph and thus their indicator values are affected. The second case is that the scenario of the first case is applied in the reverse way to class  $A$  and  $B$ . The third case is where we do not have any changes. By considering these three different cases, the upper bound of the quality function is derived as the following [22, Theorem 2.2, 2.3]:

$$q(\{T\}) \leq q(\{S\}) + \max \left\{ \begin{array}{l} A_{S_1} \cdot (B_{S_1} - B_{S_0}) \\ (A_{S_1} - A_{S_0}) \cdot B_{S_1} \\ 0 \end{array} \right\}. \quad (3)$$

While expanding subgraphs, we prune the children of supergraphs  $T$  expanded from the subgraph  $S$ , when the quality function of  $T$  is equal to the one of those upper bounds. This culling criterion is adopted, since it is guaranteed that we cannot find any better supergraphs than  $T$  whose quality function is higher than the the upper bound shown in the aforementioned inequality. This approach has been proven to identify discriminative subgraphs whose quality function values are bigger than a certain lower bound [22, Theorem 2.1]. Furthermore, unlike gSpan, users do not need to provide manually-tuned parameters for identifying discriminative subgraphs.

### 4.3 TF-IDF normalization

Once we extract features, we normalize those features. TF-IDF [20] is one of commonly adopted normalization schemes, mainly for document classification. We apply this normalization to our feature, which resembles the bag-of-word model. Inspired by the TF-IDF normalization scheme, we give higher weights to more frequent features in each image and deemphasize features that appear in more images.

In particular, our TF-IDF weighting scheme of a subgraph  $s$  occurring in an image  $i$  given an image database  $D$  is defined as the following:

$$\begin{aligned} TF\text{-}IDF(s, i, D) &= TF(s, i) \times IDF(s, D), \\ TF(s, i) &= \log(1 + f_{s,i}), \\ IDF(s, D) &= \log\left(\frac{N}{1 + n_s}\right), \end{aligned} \quad (4)$$

where  $f_{s,i}$  is the number of the subgraph  $s$  occurring in the image  $i$ ,  $N$  is the number of all images in the database  $D$ , and  $n_s$  is the number of images with the subgraph  $s$ . If  $f_{s,i}$  is zero, TF term would be undefined. To prevent this case, a small constant, 1, is added. Similarly, to avoid divide-by-zero, we also add the small constant 1 to the denominator of the IDF term.

## 5 Results

We implemented prior and our methods for discovering family photos in a machine that has Xeon 3.47GHz with 192GB main memory. We evaluate the effectiveness of computing and using discriminative feature selection along with TF-IDF normalization. For classification, we use the support vector machine

	(a) images only in Chen's	(b) images both in Chen's and Ours	(c) images only in Ours
family	66	1,111	502
non-family	136	1,131	759

**Tab. 1** The composition of Chen's ( $a + b$ ) and our data set ( $b + c$ ). (b) indicates the number of co-occurring images in both Chen's and ours. Many images of family and non-family co-occur in ours and Chen's, although we prepare the extended data set without looking at their original labels.

(SVM). The classification is conducted with linear kernel and 5-fold cross validation.

## 5.1 Datasets

To validate our approach, we use the existing dataset provided by Chen et al. [2]. We additionally test different methods against a new, larger and diverse dataset, which is rearranged from the public data set [8], as adopted also in the previous work. Based on the protocol laid out in the prior work, we obtain a 'soft' ground truth containing 1,613 family photos and 1,890 non-family photos for our new, extended dataset. The 'soft' ground truth for the new dataset is generated without any prior knowledges such as looking at labels of those images.

The new extended dataset also shares the same images to the Chen's dataset, since these two datasets are arranged from the public dataset. We also measure the common images in both or either one of two datasets (Tab. 1). The difference from the previous one is that the new dataset has 1,073 more photos and includes wider sets of family types such as siblings, single parent, nuclear family, and extended family, as shown in Fig. 1.

Note that these images from the public data set have labels, which are groups, wedding, and family types. Our methods independently predict family types of these images and measure accuracy by comparing their predicted labels with the original ones associated with the public data set.

We have also considered other datasets related to group photos [5, 14]. Unfortunately, these datasets do not contain labels directly for family and non-family types. As a result, we were unable to use them for our problem.

## 5.2 Effects of discriminative subgraphs

We test accuracy of different methods including ours and the gSpan method [2]. We have implemented the prior method by following the guideline of the original paper [2]. For gSpan, we generate frequent subgraphs

up to 10 K subgraphs and sort them in the order of document frequency and select them as BoFG. To achieve the best accuracy for the gSpan method, it is required for users to specify the number of subgraphs. In this approach, we need to rely on many trial-and-error procedures, while our method automatically constructs a set of discriminative subgraphs.

We were unclear how the prior method uses the DF term, because there is an ambiguity in which the DF term is evaluated either after or during the running process of gSpan<sup>3</sup>. We thus experiment both cases.  $gSpan + DF(1)$  and  $gSpan + DF(2)$  correspond respectively to the adaption of DF *posterior to* and *during* gSpan in Tab. 2. In Tab. 2, our method finds the maximal number of subgraphs without using the minimum frequency.

Our methods w/ and w/o the TF-IDF scheme in the Chen's dataset identify a small set of discriminative subgraphs (i.e., 76 subgraphs), and achieves 80.61% and 78.65% accuracy respectively. Our method in the extended dataset achieves 77.26%, and shows 79.34% with the TF-IDF scheme.  $gSpan + DF(1)$  and  $gSpan + DF(1)$  methods show inferior results over our method in most cases. Interestingly, the prior methods show even lower accuracy as they use higher dimensions. This is mainly because frequent subgraphs may not be discriminative.

## 5.3 Effects of TF-IDF normalization

We measure accuracy of different methods with and without TF-IDF normalization. Since  $gSpan + DF(2)$  achieves higher accuracy than  $gSpan + DF(1)$ , we show the results of  $gSpan + DF(2)$  and ours for the test.

In both  $gSpan + DF(2)$  and ours, using TF-IDF over DF improves the classification accuracy in most cases. Especially, our method using TF-IDF achieves the highest accuracy, 79.34%, for the extended dataset.

## 5.4 Comparison of subgraphs

We check the number of subgraphs co-occurring in the BoFG features generated by both gSpan and our method. This investigation can help us to understand how many dimensions prior methods require in order to obtain discriminative features extracted by our method. Even in hundreds of thousands of dimensions extracted by gSpan, some of discriminative subgraphs extracted by our method are not identified (Tab. 4).

We also measure how well query images used in the test phase are represented by extracted features. For this, we measure how many query images are

<sup>3</sup>We have consulted authors of the gSpan technique for faithful re-implementation of the gSpan method.

Chen's data set									
<i>dimension</i>	76	100	500	1,000	2,000	3,000	4,000	5,000	10,000
$gSpan + DF(1)$	50.00%	51.51%	52.98%	54.65%	61.92%	68.33%	69.52%	68.78%	77.76%
$gSpan + DF(2)$	78.61%	77.92%	80.12%	78.16%	77.51%	77.31%	76.49%	77.14%	77.76%
<b><i>Ours + TF-IDF</i></b>	<b>80.61%</b>	N/A							
Our data set									
<i>dimension</i>	90	100	500	1,000	2,000	3,000	4,000	5,000	10,000
$gSpan + DF(1)$	56.00%	58.37%	62.25%	61.26%	64.51%	67.48%	69.64%	71.84%	75.61%
$gSpan + DF(2)$	74.78%	74.84%	77.43%	76.80%	76.63%	76.83%	76.49%	76.09%	75.61%
<b><i>Ours + TF-IDF</i></b>	<b>79.34%</b>	N/A							

**Tab. 2** The accuracy of different methods in Chen's and our datasets.

Chen's dataset						Our data set				
<i>dimension</i>	76	100	1,000	5,000	10,000	90	100	1,000	5,000	10,000
$gSpan + DF(2)$	78.61%	77.92%	78.16%	77.14%	77.76%	74.78%	74.84%	76.8%	76.09%	75.61%
$gSpan + DF(2) + TF-IDF$	77.67%	77.63%	81.31%	81.18%	82.04%	75.40%	75.09%	78.09%	77.55%	77.2%
<i>Ours</i>	78.65%	N/A				77.26%	N/A			
<b><i>Ours + TF-IDF</i></b>	<b>80.61%</b>	N/A				<b>79.34%</b>	N/A			

**Tab. 3** The accuracy of  $DF$  vs.  $TF-IDF$  in Chen's and our extended datasets.

represented by null vector, indicating that query images are not represented by any features extracted by  $gSpan$  or our method (Tab 5). As a result, we can conclude that the feature extraction of our method performs better than other tested methods ( $gSpan + DF(1)$  or  $gSpan + DF(2)$ ).

## 6 Conclusions and future works

We have proposed a novel classification system utilizing discriminative subgraph mining for achieving high accuracy. We represent group photos as graphs with age, gender, and face position, and then extract discriminative subgraphs and construct BoFG features. For extracting discriminative subgraphs, we proposed to use a recent discriminative subgraph mining method, CORK, that adopts a quality function with near-optimal guarantees. We additionally proposed to use the TF-IDF normalization to better support the characteristic of BoFG features. To validate benefits of our approach, we have tested different methods including ours against two different datasets including our new, extended dataset. Our method achieves higher accuracy in the same dimensionality over the prior methods. Furthermore, our method achieves higher or similar accuracy over the prior work that relies on manual turning and requires a higher dimensionality.

There are many interesting future directions. Since our work is based on the concept of social relationships, we consider subgraphs consisting of at least two nodes. However, only a single node can provide

useful social cues. Incorporating single nodes in BoFGs and investigating its effects should be interesting. We would like to also investigate recent deep learning techniques that learns low-level features and classification functions. Due to the lack of sufficient training datasets, we did not consider recent deep learning techniques, but this approach should be worthwhile for achieving higher accuracy.

## Acknowledgements

We are thankful to our lab members for valuable feedbacks, and to Ph.D. Yan-Ying Chen for sharing her dataset. This work was supported in part by MSIP/IITP [R0126-16-1108, R0101-16-0176] and MSIP/NRF (No. 2013-067321).

## References

- [1] X. Chen, B. Zhou, Y. Guo, F. Xu, and Q. Zhao. Structure guided texture inpainting through multi-scale patches and global optimization for image completion. *Science China Information Sciences*, 57(1):1–16, 2013.
- [2] Y.-Y. Chen, W. H. Hsu, and H.-Y. M. Liao. Discovering informative social subgraphs and predicting pairwise relationships from group photos. In *Proceedings of the 20th ACM international conference on Multimedia*, pages 669–678. ACM, 2012.
- [3] Y.-I. Chiu, R.-Y. Hsu, and C.-R. Huang. Spatial face context with gender information for group photo similarity assessment. In *Pattern Recognition (ICPR)*,

Chen's data set: <b>77 subgraphs</b> by our method									
<i>gSpan subgraphs</i>	78	100	500	1,000	10,000	...	111,764	...	560,177
# of common subgraphs	16	17	21	23	36	...	54	...	<b>59</b>
Our data set: <b>85 subgraphs</b> by our method									
<i>gSpan subgraphs</i>	85	100	500	1,000	10,000	...	84,713	...	326,034
# of common subgraphs	20	21	25	26	29	...	50	...	<b>59</b>

**Tab. 4** The number of common subgraphs between gSpan and ours in Chen's and our data set.

<i>dimension</i>	76	100	1,000	5,000	10,000
<i>gSpan + DF(1)</i>	283	283	189	82	30
<i>gSpan + DF(2)</i>	53	52	39	37	30
<b>Ours</b>	<b>26</b>	N/A			
<i>dimension</i>	90	100	1,000	5,000	10,000
<i>gSpan + DF(1)</i>	351	351	236	163	43
<i>gSpan + DF(2)</i>	62	62	50	48	43
<b>Ours</b>	<b>28</b>	N/A			

**Tab. 5** This table shows the number of query images that are represented by the null vector in Chen's and our extended data.

- 2014 22nd International Conference on, pages 2673–2678. IEEE, 2014.
- [4] Y.-I. Chiu, C. Li, C.-R. Huang, P.-C. Chung, and T. Chen. Efficient graph based spatial face context representation and matching. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 2001–2005. IEEE, 2013.
- [5] W. Choi, Y.-W. Chao, C. Pantofaru, and S. Savarese. Discovering groups of people in images. In *Computer Vision–ECCV 2014*, pages 417–433. Springer, 2014.
- [6] S. A. Cook. The complexity of theorem-proving procedures. In *ACM symposium on Theory of computing*, pages 151–158. ACM, 1971.
- [7] A. C. Gallagher and T. Chen. Finding rows of people in group images. In *ICME*, pages 602–605. IEEE, 2009.
- [8] A. C. Gallagher and T. Chen. Understanding images of groups of people. In *CVPR*, pages 256–263. IEEE, 2009.
- [9] S.-M. Hu, F.-L. Zhang, M. Wang, R. R. Martin, and J. Wang. Patchnet: A patch-based image representation for interactive library-driven image editing. *ACM Trans. Graph.*, 32(6), Nov. 2013.
- [10] C. Jiang, F. Coenen, and M. Zito. A survey of frequent subgraph mining algorithms. *The Knowledge Engineering Review*, 28(01):75–105, 2013.
- [11] A. Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.
- [12] H. Li, W. Wu, and E. Wu. Robust interactive image segmentation via graph-based manifold ranking. *Computational Visual Media*, 1(3):183–195, 2015.
- [13] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
- [14] A. C. Murillo, I. S. Kwak, L. Bourdev, D. Kriegman, and S. Belongie. Urban tribes: Analyzing group photos from a social perspective. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 28–35. IEEE, 2012.
- [15] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *IJCV*, 2001.
- [16] K. Shimizu, N. Nitta, Y. Nakai, and N. Babaguchi. Classification based group photo retrieval with bag of people features. In *ACM International Conference on Multimedia Retrieval*, page 6. ACM, 2012.
- [17] H. Shu, A. Gallagher, H. Chen, and T. Chen. Face-graph matching for classifying groups of people. In *Image Processing (ICIP), 2013 20th IEEE International Conference on*, pages 2425–2429. IEEE, 2013.
- [18] P. Singla, H. Kautz, J. Luo, and A. Gallagher. Discovery of social relationships in consumer photo collections using markov logic. In *Computer Vision and Pattern Recognition Workshops, 2008. IEEE Computer Society Conference on*, pages 1–7. IEEE, 2008.
- [19] Z. Song, M. Wang, X.-s. Hua, and S. Yan. Predicting occupation via human clothing and contexts. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1084–1091. IEEE, 2011.
- [20] K. Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21, 1972.
- [21] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.
- [22] M. Thoma, H. Cheng, A. Gretton, J. Han, H.-P. Kriegel, A. Smola, L. Song, P. S. Yu, X. Yan, and K. M. Borgwardt. Discriminative frequent subgraph mining with optimality guarantees. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 3(5):302–318, 2010.
- [23] G. Wang, A. Gallagher, J. Luo, and D. Forsyth. Seeing people in social context: Recognizing people and social relationships. In *Computer Vision–ECCV 2010*, pages 169–182. Springer, 2010.
- [24] M. Wang, Y.-K. Lai, Y. Liang, R. R. Martin, and S.-

- M. Hu. Biggerpicture: data-driven image extrapolation using graph matching. *ACM Transactions on Graphics (TOG)*, 33(6):173, 2014.
- [25] X. Yan and J. Han. gspan: Graph-based substructure pattern mining. In *Data Mining, IEEE International Conference on*, pages 721–724. IEEE, 2002.
- [26] T. Zhang, H. Chao, C. Willis, and D. Tretter. Consumer image retrieval by estimating relation tree from family photo collections. In *ACM International Conference on Image and Video Retrieval*, pages 143–150. ACM, 2010.
- [27] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Comput. Surv.*, 35(4), 2003.



**Changmin Choi** is in preparation for a start-up company. He received a M.S. degree from School of Computing at KAIST, and a B.A. degree from Business School at Hanyang Univ. His research interest is understanding group photos in social media.



**YoonSeok Lee** is a M.S. student in

School of Computing at KAIST, South Korea and he received a B.S. degree in Computer Science from KAIST in 2014. His research interest lies in image classification, image representation and hashing techniques.



**Sung-Eui Yoon** is currently an associate professor at KAIST. He received the B.S. and M.S. degrees in computer science from Seoul National University in 1999 and 2001, respectively. His main research interest is on designing scalable graphics, image search, and geometric algorithms.

He gave numerous tutorials on proximity queries and large-scale rendering at various conferences including ACM SIGGRAPH and IEEE Visualization. Some of his work received a distinguished paper award at Pacific Graphics, invitations to IEEE TVCG, an ACM student research competition award, and other domestic research-related awards. He is a senior member of IEEE, and a member of ACM and KIISE.