# A NOVEL FRAMEWORK FOR FAST SCENE MATCHING IN CONSUMER IMAGE COLLECTIONS

Xu Chen[1]
Dept. of Electrical and Computer Eng.,
University of Michigan, Ann Arbor, USA
Email: xhen@umich.edu

Madirakshi Das  Alexander Loui
Kodak Research Laboratories, Eastman Kodak Company,
Rochester, NY, USA
Email: madirakshi.das@kodak.com, alexander.loui@kodak.com

## ABSTRACT

*The widespread utilization of digital visual media has motivated many research efforts towards efficient search and retrieval from large photo collections. Traditionally, SIFT feature-based methods have been widely used for matching photos taken at particular locations or places of interest. These methods are very time-consuming due to the complexity of the features and the large number of images typically contained in the image database being searched. In this paper, we propose a fast approach to matching images captured at particular locations or places of interest by selecting representative images from an image collection that have the best chance of being successfully matched by using SIFT, and relying on only these representative images for efficient scene matching. We present a unified framework incorporating a set of discriminative features that can effectively select the images containing signature elements of particular locations from a large number of images. The proposed approach produces an order of magnitude improvement in computational time for matching similar scenes in an image collection using SIFT features. The experimental results demonstrate the efficiency of our approach compared to the traditional SIFT, PCA-SIFT, and SURF-based approaches.*

***Index Terms***— Clustering, Image Search and Retrieval, SIFT, Occlusion, Blur, Classification.

## 1. INTRODUCTION

Scene matching refers to the process of matching a region in one image with the corresponding region in another image where both image regions are part of the same scene. Scene matching plays an important role in determining location, since most digital media currently being captured and the billions of digital images taken before the availability of GPS lack detailed location information. In the absence of this information, the location at which a photograph was captured can be described by unique objects in the stationary background that can be matched across images [1].

Earlier work on scene matching involved computing correlation between images [2]. However, in addition to being very computationally intensive, these methods cannot handle the large variations in scale, lighting, and pose encountered in consumer images. There has been recent work on matching feature-rich complex scenes using scale-invariant features (SIFT) [3] and faster feature extractions such as PCA-SIFT and SURF [4][5]. However, these techniques have been mainly used to match and register the entire scene for every image, which is a time-consuming process when a large image database is involved. To avoid comparing each pair of images in the database to detect scene matches, we propose a method for selecting a few representative images that can be matched reliably based on their image characteristics.

The criteria used for choosing representative images from a group of images (or a keyframe from a segment of video) are often tied to image quality [6] and the best similarity with other images in the group. However, a good-quality photo may not be the best candidate for scene matching. For instance, photos that contain large homogenous regions such as water, grass, and sky could be of high quality but they are usually poor for scene matching since they lack specific features that could determine the locations. Also, photographs of people, which constitute a large portion of consumer image databases, are not good candidates for scene matching if the people in the picture are occluding the background, or if the background is plain, whereas these characteristics usually result in a good quality portrait. In this work, we explore image features that evaluate images in terms of their value in providing good matching opportunities for SIFT and SIFT-like features. Typically, good images for this application contain distinctive elements with complex edge structures. These image features are incorporated into a framework for selecting representative images appropriate for the scene matching task.

The rest of the paper is organized as follows: in Section 2, we present a framework for selection of representative images. In Section 3, we investigate the feature representation for our framework. We discuss the classification algorithms relying on the extracted features in Section 4. The comparison of the performance of our approach with traditional approaches is demonstrated in

[1] Work done as intern at Eastman Kodak Company

978-1-4244-7493-6/10/$26.00 ©2010 IEEE

Section 5. We finally give a brief summary and conclusion in Section 6.

## 2. OUR APPROACH

A large number of images in consumer image collections are not suitable for scene matching. Earlier work using consumer collections [1] observed that only about 10% of the images taken at co-located events could be actually matched using scene-matching techniques. The main reasons for this are: (1) The background elements that can be used for matching images captured at the same location are mostly occluded by the people in the images. (2) The images are blurry due to focusing problems or camera or object motion, resulting in failure of SIFT feature point detection. (3) The images contain few meaningful edges and specific objects, e.g., images with natural scenes, or generic objects such as cabinets and furniture common to many locations. Our goal is to select the best images for successful SIFT-based scene matching, while eliminating the images with the above-mentioned problems.

An event-clustering algorithm described in [7] divides a user's collection into events and sub-events using temporal and color histogram information, where events are very likely to have been captured at the same location because of their temporal proximity. Our approach to selecting the best images for SIFT-based scene matching can be used to select a few representative images from each event, thus greatly reducing the number of images that need to be matched in the collection.

We formulate the problem of selection of representative images as a fast binary classification problem, namely separation of good representative images from unsuitable images in consumer image collections, using features that can discriminate between the two classes. Fig. 1 shows a block diagram of our approach. The output of our method is a shortlist of images that are suitable for SIFT-based scene matching.

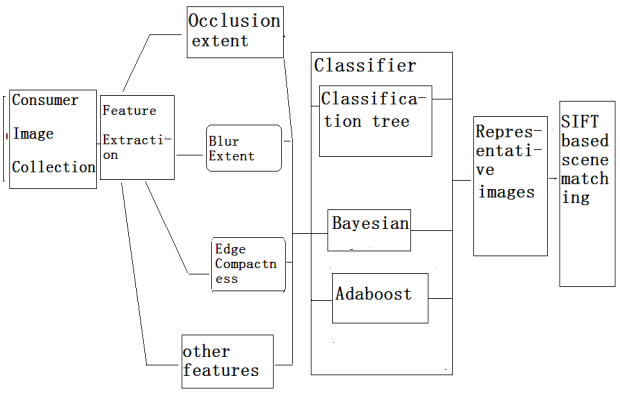


Fig. 1. Block diagram of our framework.

## 3. RELEVANT FEATURE EXTRACTION

This section describes the features we have developed to distinguish between good images for scene matching, and other images in the collection.

### 3.1. Occlusion extent

Based on evidence gathered from a large number of consumer images, it can be seen that the occlusion of objects of interest because of the presence of people is an important factor in determining whether the image is a good candidate image for scene matching. Obviously, the higher the extent of the occlusion, the smaller the probability that the image can be matched with other images from the same scene by using unique objects present in the scene. Typically, a large number of images in consumer image collections contain people. To measure the occlusion extent, we determine the approximate position of people in images using face detection. Specifically, we estimate the position of people from the size and the position of the facial circle (the radius, x and y coordinates of the center of the circle) output by a face detector.

We utilize the face detection algorithm by Jones et al. [8] due to its faster implementation. It has to be noted that there are other face detection approaches that could yield higher accuracy, while they may not achieve the speed of [8]. We rely on bagging [9] to reduce the false detection and improve the accuracy for face detection. Particularly, in three runs of the detector using different parameters, if the detection results satisfy $|x(i)-x(j)|<5$, $|y(i)-y(j)|<5$, $|r(i)-r(j)|<5$, where $i, j = 1,2,3\ i \neq j$, we consider the detection to be correct. We choose the regions with the highest vote as the region for face detection. Subsequently, we estimate the area of the regions of human body by making reasonable assumptions that (1) the width of the shoulder is approximately twice as long as the width of the head, (2) the length of the body is approximately four times as long as the length of the head. We obtain these priors by averaging the results over 3000 images with different postures of people, including standing and sitting. We compute the occlusion extent as:

Occlusion extent $k_1 = \dfrac{P}{Q}$

where P is the total number of occluded pixels and Q is the total number of pixels in the image. Fig. 2 illustrates the occlusion extent for different consumer images. It must be noted that there could be some situations that people overlap with each other, which could make our results slightly larger than the true value. However, from the experimental results, it has little influence on our binary classification results. Furthermore, this problem can be solved by first sorting the human face by x and y-coordinates and then determining the overlapping regions of adjacent human faces.

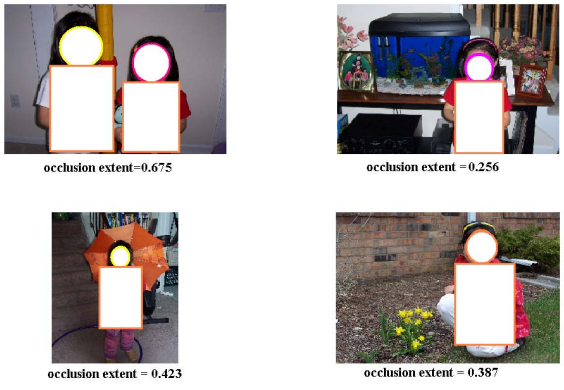


Fig. 2. Visual illustration of occlusion extent for some example images from consumer collections.

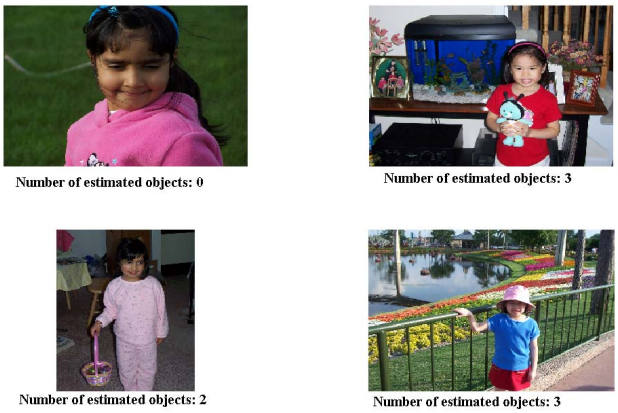


Fig. 3. Visual illustration of the number of estimated objects for some example images from consumer collections.

### 3.2. Compactness of edges

Since the success of scene matching is highly dependant upon the number of unique objects contained in the images, it is desirable to roughly determine the number of unique objects in any candidate image. The larger the number of unique objects contained in the image, the higher the probability that the image could be a good candidate image for scene matching. The edge information is the most salient information for determination of specific objects [10][11]. The edge-clustering method has shown to be an efficient method for fast object detection [12]. In this work, we evaluate the number of possible unique objects by first spatially clustering the edges and then computing the variance of each cluster. If the variance of each cluster is smaller than some threshold, we consider the cluster to be a specific object.

To remove the small variations in pixel values and false edges due to the illuminations and lightness, we preprocess the image with Gaussian filters to smooth the images. Subsequently, we apply the canny detectors to the smoothed images. We subtract the edges of people by combining the results from face detection and people detection discussed in subsection 3.1 from the total edge map. By doing this, it is guaranteed that the remaining edges are mostly from non-people objects in the image.

Once we obtain the edge map for objects, k-means clustering [13] is utilized to partition the $n$ edges into $k$ clusters in which the mean distance of edges from the partition center is minimized. Based on our experiment, a value of $k$=5 is found to be discriminative for classification. We compute the feature compactness of edges, $k_2$, as the number of estimated specific objects. Fig. 4 shows the steps in determining this feature.

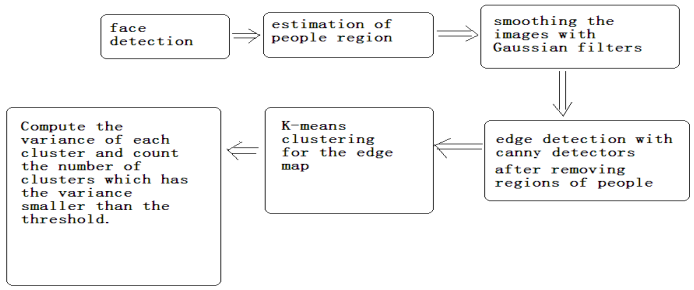


Fig. 4. Steps in the extraction of compactness of edges feature.

### 3.3 Blur extent

Some of the consumer images are of poor quality due to lack of focus and motion blur, and not suitable as candidate images for scene matching. As we know, edges can be generally classified into four types: Dirac-Structure, Astep-Structure, Gstep-Structure and Roof-Structure. We rely upon the approach described in Tong et al. [14] to determine the blur extent of images. Typically, if blur occurs, both of the Gstep-structure and Roof-structure tend to lose their sharpness as described in [14]. Fig. 5 shows the steps to determine the types of edges. The blur extent is computed as:

Blur extent $k_3 = \frac{N_1}{N_2}$

where $N_1$ denotes the sum of the number of Gstep-structure and Roof-structure edges, $N_2$ denotes the total number of edges. There are other features that could further improve the accuracy of selections, such as: contrast, brightness, color histogram, and the camera focal length. However, in the experimental results, the overall improvement is not very

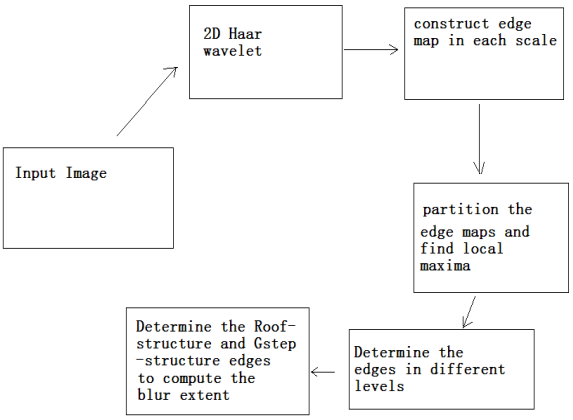


Fig. 5. Steps in the extraction of the blur extent feature.

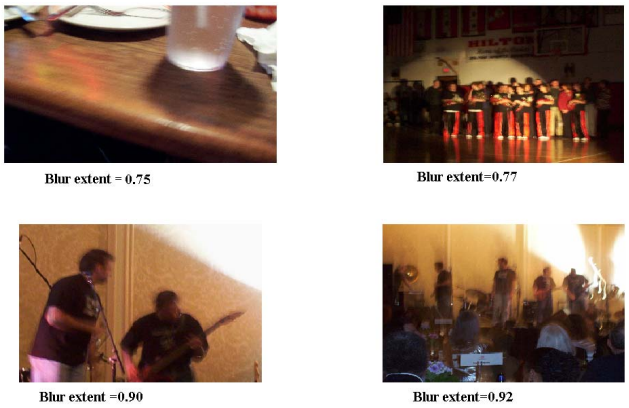


Fig. 6. Visual illustration of the blur extent for some example images from consumer collections.

significant since these affect only certain types of events. Therefore, in the section on experimental results, we mainly present the results using the first three most discriminative features.

## 4. CLASSIFICATION ALGORITHMS

We explore a number of classification strategies for grouping images into two categories – good candidates for scene matching and poor candidates for scene matching – based on the features we have described in the previous section.

### 4.1. Naive Bayesian framework

Given the list of features $k_i$, one way to integrate them in one unified framework is with the Naïve Bayesian framework [6]. Let us assume we consider the first three most discriminative features (it can be easily extended to utilize more features). The overall quality metric according to Bayes rule is defined as:

$$k_{all} = \frac{P(good \mid k_1, k_2, k_3)}{P(bad \mid k_1, k_2, k_3)} = \frac{P(k_1, k_2, k_3 \mid good)P(good)}{P(k_1, k_2, k_3 \mid bad)P(bad)}$$

Assuming independence of the features given the class,

$$k_{all} = \frac{P(k_1 \mid good)P(k_2 \mid good)P(k_3 \mid good)P(good)}{P(k_1 \mid bad)P(k_2 \mid bad)P(k_3 \mid bad)P(bad)}$$

We can choose equal numbers of good and bad candidate images, so that P(good) and P(bad) can be dropped from the equations.

### 4.2. Adaptive boosting classifier

AdaBoost [15] can also be effectively used in our problem since AdaBoost is adaptive in the sense that subsequent classifiers built are tweaked in favor of those instances misclassified by previous classifiers. AdaBoost provides a good way for us to assign proper weights to these different features. In our case, it is entirely possible that the image that has less occlusion is heavily blurred. The weak classifiers used are constructed from single features using a Bayesian classifier based on a unitary Gaussian model. Fig. 7 shows the performance when using AdaBoost with two and three features.

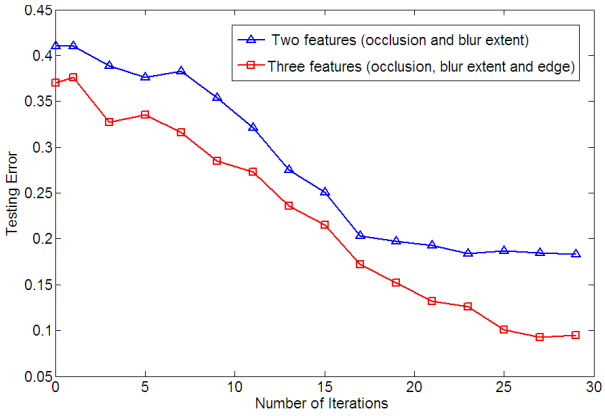


Fig. 7. The classification error with AdaBoost relying on two features (occlusion and blur extent) and three features (occlusion, blur extent, and compactness of edges).

### 4.3. Classification tree-based approach

Another classifier that can be used in this problem is a rule-based classification tree. Fig. 8 shows the structure of the classification tree that is constructed based on experimental results. We put the strongest feature at the root node of the tree and the second strongest feature at the second level of

the tree, and so on. For example, if the occlusion extent of an image is very large, the image has already been determined as a bad candidate for scene matching and there is no need to consider other features for this image. Therefore, this method can save significant computational time while it maintains high classification accuracy.

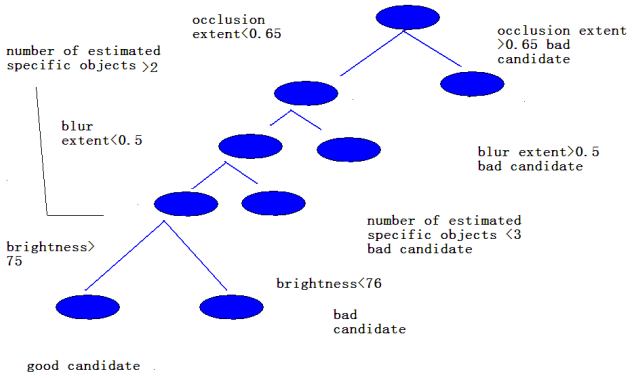


Fig. 8. Classification tree used for identifying good candidate images for scene matching.

## 5. EXPERIMENTAL RESULTS

### 5.1. Selection of ground truth

We collected the training and test data by using the software described in [1], which performs scene matching based on SIFT features. We selected 3000 positive example images (images producing matches) and 3000 negative example images (images that could not be matched with other images). For the testing stage, five-fold cross validation was used. For each test, 1 in 5 images is used as a test image and the other 80% serve as training images. In Naïve Bayesian framework, we assume each feature has Gaussian distribution and the mean and variance can be computed in the training stage.

### 5.2. Simulation results and comparison

Fig.9 shows the classification performance with the Naive Bayesian classifier, adaptive boosting classifier and classification tree for different combination of features. We achieve an average accuracy of 88% when using Adaboost with the three features described in section 3. The accuracy refers to the percentage of images automatically selected by our method that would be considered good candidates for matching according to the ground truth.

Fig. 10 provides a visual illustration of the SIFT-based scene-matching results using selected good representative images with our system, which demonstrates the effectiveness of our approach. Fig. 11 shows more examples of good representative images with the proposed approach. Among the 25 representative images chosen by our system, only 2 are false positives. The main reason for the failures in this case is because the faces of people shown in the image are not frontal faces, therefore the face detector cannot detect them correctly. There are also differences in characteristics between images captured indoor and outdoor images causing some failures. For the indoor events, the occlusion extent usually plays the most important role and serves as the strongest feature since in consumer image collections, there are often many people in the indoor events. For the outdoor events, the brightness and compactness of edges are more important.

We compared the performance of our framework with three different classifiers with scene matching using SIFT [3], PCA-SIFT [4] and SURF [5] in terms of computation time. The baseline computation time for the scene matching algorithms is the time taken to match every pair of images in the collection. This is compared to the approach of selecting representative images by our method, followed by matching performed on the representative images. The experimental results are shown in Table 1 for finding scene matches for 3000 images with 46 events. It can be seen that our approach provides an order of magnitude savings in the computation time for scene matching.

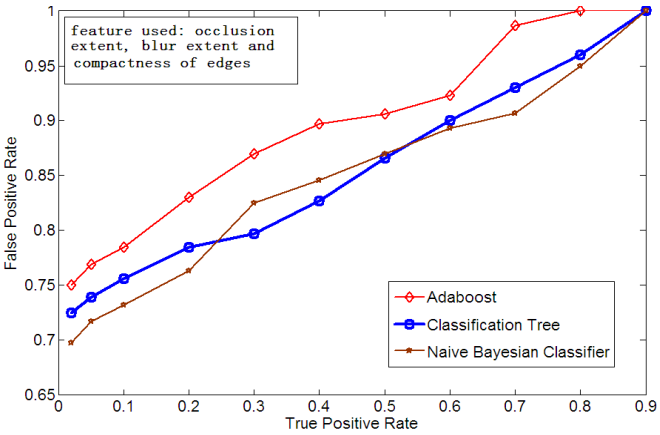


Fig. 9. Comparison of classification accuracy with different classifiers.

## 6. CONCLUSION

In this paper, we present a framework for selecting good candidate images for scene matching using SIFT-like features. The proposed approach is an order of magnitude faster than scene matching using the traditional approach of comparing images pair-wise. We extract features relevant to successful matching including occlusion extent, compactness of edges, and blur extent. We investigate a number of classifiers for binary classification of images into good candidates and poor candidates for scene matching. We apply our framework on large consumer image databases and measure the savings in computational time.

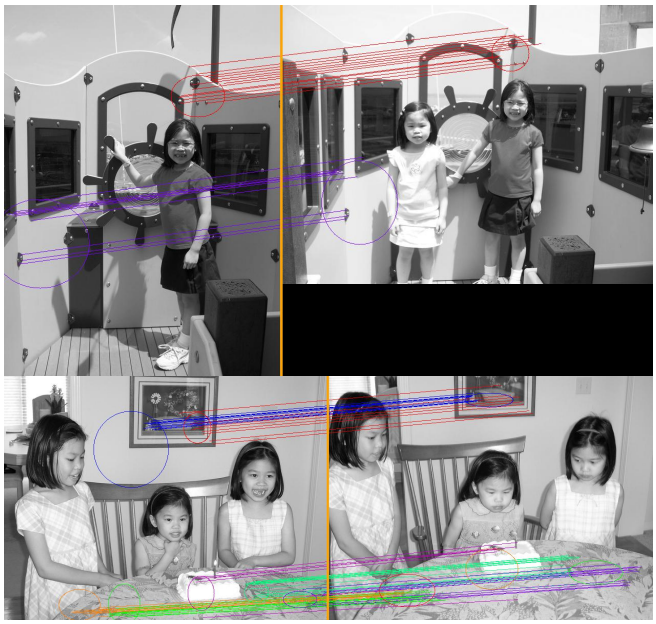

Fig. 10. Examples of matching results with our selected candidate images.

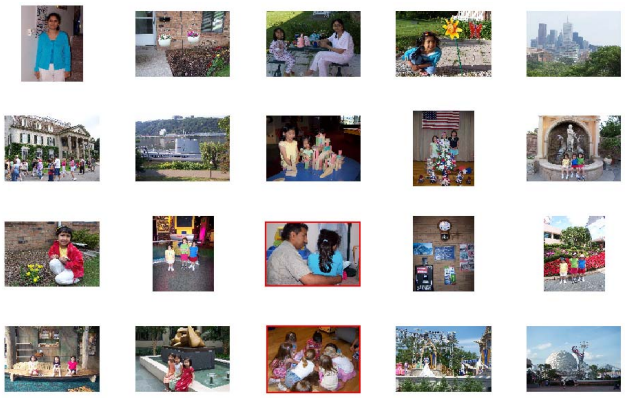

Fig. 11. Good representative images for fast scene matching with SIFT features. Red boxes indicate false positives.

The experimental results demonstrate the advantages of our approach in terms of saving significant computational time while achieving high accuracy. Future work will focus on utilizing sub-images to save additional computational time. Detection of non-frontal faces can further improve the accuracy of selection of candidate images, as well as training separate classifiers for indoor and outdoor images and applying indoor/outdoor detection to determine the image type.

Table 1: Computation time for our approach compared with traditional approaches.

| **Traditionl approachs** | Time taken (seconds) | **Our approaches** | Time taken (seconds) |
|---|---|---|---|
| SIFT | $7.43 * 10^4$ | Classification tree | 1288.6 |
| PCA-SIFT | $4.35 * 10^4$ | Bayesian Classifier | 2536.5 |
| SURF | $2.37 * 10^4$ | AdaBoost | 2876.2 |

## 7. REFERENCES


[1] M. Das, J. Farmer, A. Gallagher, and A. Loui, " Event-based Location Matching for Consumer Image Collections," International Conference on Image and Video Retrieval, 2008.

[2] E. Hall, D. Davies, and M. Casey, "The Selection of Critical Subsets for Signal, Image and Scene Matching," IEEE Transactions on PAMI, vol. 2, no. 4, pp. 313-322, 1980.

[3] D. Lowe, "Distinctive Image Features from Scale Invariant Features," Intl. J. Comput. Vision (IJCV), 60 (2), 2004.

[4] Y. Ke and R. Sukthankar, "PCA-SIFT: A More Distinctive Representation for Local Image Descriptors," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2004.

[5] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "SURF: Speeded Up Robust Features," Computer Vision and Image Understanding (CVIU), vol. 110, no. 3, pp. 346-359, 2008.

[6] Y. Ke, X. Tang, and F. Jing, "The Design of High-level Features for Photo Quality Assessment," IEEE CVPR, 2006.

[7] A. Loui and A. Savakis, "Automated Event Clustering and Quality Screening of Consumer Pictures for Digital Albuming," IEEE Trans. Multimed., 390-402, Sept 2003.

[8] M.J. Jones and P. Viola, "Face Recognition Using Boosted Local Features," IEEE Conference on Computer Vision, 2003.

[9] L. Breiman, "Bagging Predictors," Machine Learning, 1996.

[10] R.Y. Wong, "Sequential Scene Matching Using Edge Features," IEEE Transactions on Aerospace and Electronic Systems, vol. AES-14, Jan. 1978, p. 128-140.

[11] J. Gao and J. Yang, "An Adaptive Algorithm for Text Detection from Natural Scenes," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2001.

[12] W. Cui, H. Zhou, H. Qu, P. Wong, and X. Li, "Geometry-Based Edge Clustering for Graph Visualization," IEEE Transactions on Visualization and Computer Graphics, 2008.

[13] R. Duda, P. Hart, and D. Stork, 2001. Pattern Classification. Second Edition, Wiley, New York, 526-528.

[14] H. Tong, M. Li, H. Zhang and C. Zhang, "Blur Detection for Digital Images Using Wavelet Transform," IEEE International Conference on Multimedia and Expo (ICME), 2004

[15] R. Schapire and Y. Singer, "Improving Boosting Algorithms Using Confidence-rated Predictions," Machine Learning, vol. 37, no. 3, pp. 297-336, Dec 1999.