

# Sparse Representation for Image Similarity Assessment using SIFT Algorithm

**Shanmugam C**

Assistant Professor(Senior)  
Dept of ECE  
Sri Shakthi Institute of Engineering and  
Technology, Coimbatore-62.

**Nandhini K**

Dept of ECE  
Sri Shakthi Institute of Engineering and  
Technology, Coimbatore-62.

**Poovizhi P**

Dept of ECE  
Sri Shakthi Institute of Engineering and  
Technology, Coimbatore-62.

**Priyadharshni C**

Dept of ECE  
Sri Shakthi Institute of Engineering and  
Technology, Coimbatore-62.

**Abstract** - Assessment of image similarity is fundamentally important to numerous multimedia applications. The goal of similarity assessment is to automatically assess the similarities among images in a perceptually consistent manner. In this project, we interpret the image similarity assessment problem as an information fidelity problem. More specifically, we propose a feature-based approach to quantify the information that is present in a reference image and how much of this information can be extracted from a test image to assess the similarity between the two images. Here, we extract the feature points and their descriptors from an image, followed by learning the dictionary/basis for the descriptors in order to interpret the information present in this image. Then, we formulate the problem of the image similarity assessment in terms of sparse representation. The prime importance of this project is even after the image is subjected to geometrical variations like rotating, shifting, change in orientation, change in background, still we can able to correctly identify the similarity between two image. To evaluate the applicability of the proposed feature-based sparse representation for image similarity assessment (FSRISA) technique, we apply FSRISA to three popular applications, namely, image copy detection, retrieval, and recognition by properly formulating them to sparse representation problems.

**Keywords:** *image, sparse, image similarity assessment.*

## I. INTRODUCTION

Image similarity assessment is fundamentally important to numerous multimedia information processing systems and applications, such as compression, restoration, enhancement, copy detection, retrieval, and recognition/classification. The major goal of image similarity assessment is to design algorithms for automatic and objective evaluation of similarity in a manner that is consistent with subjective human evaluation.

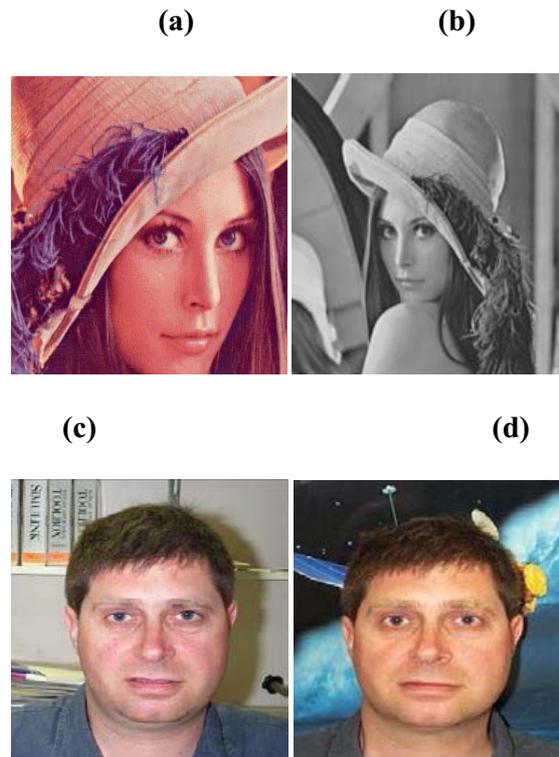


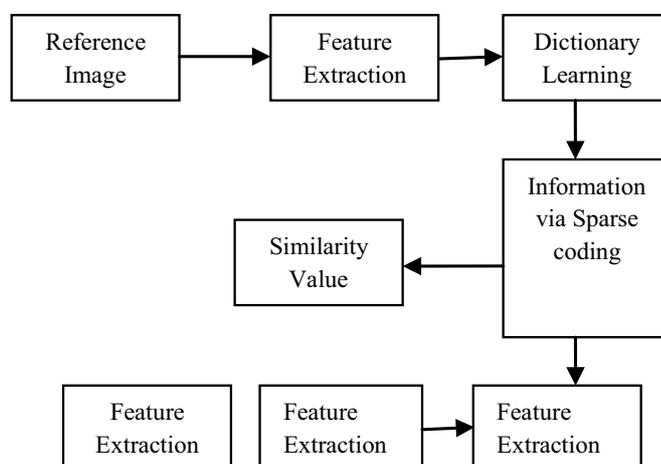
Figure 1: Some examples of image manipulations. (a) original image (b) zoomed and shifted version (c) original image (d) background modified version.

#### A. Steps in FSRISA

- Feature extraction-SIFT
- Dictionary learning-K-SVD
- Information Extraction

Feature Extraction have been discussed in the section II. Dictionary Learning using K-SVD algorithm have been discussed in section III. Finally Information extraction to calculate similarity value have been discussed in detail in section IV. Applications are described in section V. Results have been discussed in section VI.

#### B. Block diagram



### C. Algorithm

**Input:** A reference image  $I_1$  and a test image  $I_2$ .

**Output:** The similarity value between  $I_1$  and  $I_2$ , i.e.,  $\text{Sim}(I_1, I_2)$ .

1. Extract the SIFT feature vectors  $y_{1i}, i = 1, 2, \dots, K_1$ , from  $I_1$ , followed by learning the dictionary feature  $D_1$  sparsely representing  $y_{1i}$ .
2. Extract the SIFT feature vectors  $y_{2j}, j = 1, 2, \dots, K_2$ , from  $I_2$ , followed by learning the dictionary feature  $D_2$  sparsely representing  $y_{2j}$ .
3. Perform  $l_1$ -minimization by solving equation for  $y_{2j}, j = 1, 2, \dots, K_2$ , with respect to  $D_{12} = [D_1|D_2]$ .
4. Calculate the reconstruction errors,  $E_{1j}$  and  $E_{2j}$ , for  $y_{2j}, j = 1, 2, \dots, K_2$ , with respect to  $D_1$  and  $D_2$ , respectively.
5. Perform voting by comparing  $E_{1j}$  and  $E_{2j}$ , for  $y_{2j}, j = 1, 2, \dots, K_2$ , and get the percentages of votes,  $V_1$  and  $V_2$ , with respect to  $D_1$  and  $D_2$ , respectively.
6. Calculate  $\text{Sim}(I_1, I_2) = (V_1 - V_2 + 1) / 2$ .

## II. EXTRACTION OF FEATURES (SIFT)

SIFT, Scale Invariant feature transform is a method for extracting distinctive invariant features from images that can be used to perform reliable matching between different views of an object or scene. The features are invariant to image scale and rotation, and are shown to provide robust matching across a substantial range of affine distortion, change in 3D viewpoint, addition of noise, and change in illumination. The features are highly distinctive, in the sense that a single feature can be correctly matched with high probability against a large database of features from many images.

## III. DICTIONARY LEARNING (K-SVD)

Given a set of  $K$  SIFT feature vectors,  $y_i \in \mathbb{R}^{M \times 1}, i=1, 2, \dots, K$ , we apply K-SVD to find the dictionary  $D$  of size  $M \times N, M < N \ll K$ , by formulating the problem as:

$$\text{Min} ( \|x_i\|_0^2 ) \text{ subject to } \|y_i - Dx_i\|_2 \leq \epsilon, \dots (1)$$

Where  $x_i \in \mathbb{R}^{N \times 1}$  is the sparse representation coefficients of  $y_i$ ,  $\|x_i\|_0$ ,  $l_0$ -norm of  $x_i$ , counts the number of nonzero coefficients of  $x_i$ , and  $L$  is the most desired number of nonzero coefficients of  $x_i$ . We apply K-SVD to solve Eq. (1) via an iterative manner with two stages: (i) sparse coding stage: apply OMP (orthogonal matching pursuit) to solve  $x_i$  for each  $y_i$  while fixing  $D$ ; and (ii) dictionary update stage: update  $D$  together with the nonzero coefficients of  $x_i$ . The two stages are iteratively performed until convergence. It should be noted that the  $l_0$ -minimization formulation in Eq. (1) can be converted into an  $l_1$ -minimization problem and other dictionary learning algorithm, (e.g., the online dictionary learning algorithm) can be also applied in the dictionary feature extraction stage. The obtained dictionary feature  $D$  is an over complete dictionary, where  $D = \sum_{n=1, 2, \dots, N} \mathbb{R}^{M \times N}$ , contains  $N$  prototype feature vector atoms as the column vectors in  $D$ . Each original feature vector  $y_i \in \mathbb{R}^{M \times 1}, i=1, 2, \dots, K$ , can be sparsely represented as a linear combination of the atoms defined in  $D$ , satisfying  $\|y_i - Dx_i\|_2 \leq \epsilon$ , where  $\epsilon \geq 0$  is an error tolerance.



#### IV. INFORMATION EXTRACTION

After obtaining the dictionary feature for each image, we formulate the image similarity assessment based on dictionary feature matching as a sparse representation problem, described as follows. First, consider the two SIFT feature (column) vectors with First, consider the two SIFT feature (column) vectors with length  $M$ ,  $y_{1i}$ ,  $i = 1, 2, \dots, K_1$ , and  $y_{2j}$ ,  $j = 1, 2, \dots, K_2$ , extracted, respectively, from the two images,  $I_1$  and  $I_2$ , where  $K_1$  and  $K_2$  are the numbers of feature vectors of  $I_1$  and  $I_2$ , respectively. The dictionary features of  $I_1$  and  $I_2$  are  $D_1$  (of size  $M \times N_1$ ) and  $D_2$  (of size  $M \times N_2$ ), respectively, where  $M < N_1$  and  $M < N_2$ . Hence,  $y_{1i} = D_1 x_{1i}$  and  $y_{2j} = D_2 x_{2j}$ , where  $x_{1i}$  and  $x_{2j}$  are the two sparse coefficient (column) vectors with length  $N_1$  and  $N_2$ , of  $y_{1i}$  and  $y_{2j}$ , respectively.

Obviously, if  $y_{1i}$  and  $y_{2j}$  can be matched,  $y_{1i}$  can be represented sparsely and linearly with respect to  $D_2$ . On the other hand,  $y_{2j}$  can be represented sparsely and linearly with respect to  $D_1$ . To assess the similarity between a reference image  $I_1$  and a test image  $I_2$ , exploiting the discriminative characteristic of sparse representation, we want to quantify how much information present in  $I_1$  can be extracted from  $I_2$ . A sparse representation problem for representing each SIFT feature vector  $y_{2j}$  of  $I_2$  with respect to the joint dictionary  $D_{12} = [D_1 | D_2]$  can be defined as

$$\hat{x}_j = \min_{x_j} \quad \text{subject to} \quad \|x_j\|_2 \leq \epsilon \dots \dots \dots (2)$$

where  $x_j$  with length  $(N_1 + N_2)$  is the sparse coefficient vector of  $y_{2j}$  with length  $M$  of  $I_2$ .  $D_{12} = [D_1 | D_2]$  of size  $M \times (N_1 + N_2)$  is the joint dictionary concatenating  $D_1$  and  $D_2$ , and  $\epsilon \geq 0$  is an error tolerance. To solve the sparsest solution for  $x_j$ , Eq. (2) can be cast to an  $l_1$ -minimization problem as

$$\hat{x}_j = \arg \min_{x_j} (\|x_j\|_1 + \tau \|x_j\|_2) \dots \dots \dots (3)$$

where  $\tau$  is a positive real number parameter. In this project, we apply an efficient sparse coding algorithm, called the SPARSA (sparse reconstruction by separable approximation) algorithm to solve Eq. (3) in order to find the sparse representation ( $\hat{x}_j$ ) of  $y_{2j}$  with respect to the dictionary  $D_{12}$ . SPARSA is a very efficient iterative algorithm, where each step is obtained by solving an optimization sub problem involving a quadratic term with diagonal Hessian plus the original sparsity-inducing regularizer. Of course, Eq. (3) can be directly solved via a greedy algorithm, such as OMP and other  $l_1$ -minimization algorithms. It is expected that the positions of nonzero coefficients in  $\hat{x}_j$  (or the selected atoms from  $D_{12}$ ) should be highly concentrated on only one sub-dictionary (e.g.,  $D_1$  or  $D_2$ ), and the remaining coefficients in  $\hat{x}_j$  should be zeros or small enough. Also, it is intuitive to expect that the atoms for sparsely representing  $y_{2j}$  should be mostly selected from the sub-dictionary  $D_2$  learned from the feature vectors extracted from the image  $I_2$  itself, instead of  $D_1$ . If the parameters for learning the two dictionaries ( $D_1$  and  $D_2$ ) can be adequately tuned, the manner of atom selection in the sparse coding process may be changed accordingly. That is, we intend to make the sparse coefficients  $\hat{x}_j$  (or the used atoms) solved by performing sparse coding for  $y_{2j}$  more consistent with our expectation to help for similarity assessment. More specifically, we expect  $y_{2j}$  will use more atoms from  $D_2$  to represent it when  $I_2$  and  $I_1$  are visually different. On the other hand, we expect  $y_{2j}$  will use more atoms from  $D_1$  to represent it when  $I_2$  and  $I_1$  are visually similar. The details are described in the seventh paragraph of this subsection. Based on the obtained solution  $\hat{x}_j$  of Eq. (3), we can calculate

the reconstruction error as  $\|y_{2j} - D_{12} \hat{x}_j\|_2$ . By letting the elements in  $\hat{x}_j$ , corresponding to the atoms from  $D_2$ , be zeros, we can get the reconstruction error  $E_{1j}$ , using only the atoms from  $D_1$  for reconstructing  $y_{2j}$ . On the other hand, by letting the elements in  $\hat{x}_j$ , corresponding to the atoms from  $D_1$ , be zeros, we can get the reconstruction error  $E_{2j}$ , using only the atoms from  $D_2$  for reconstructing  $y_{2j}$ . If  $E_{1j} < E_{2j}$ , it is claimed that the atoms from  $D_1$  are more suitable for representing  $y_{2j}$  than those from  $D_2$ , and  $D_1$  will get a vote. Otherwise, if  $E_{2j} < E_{1j}$ ,  $y_{2j}$  is more suitable to be represented by the atoms from  $D_2$  (the dictionary learned from the feature vectors  $y_{2j}$  itself) than  $D_1$ , and  $D_2$  will get a vote. Considering all the SIFT feature vectors of  $I_2$ ,  $y_{2j}$ ,  $j = 1, 2, \dots, K_2$ , the obtained percentage of votes from  $D_1$  and  $D_2$  are denoted by  $V_1$  and  $V_2$ ,  $0 \leq V_1, V_2 \leq 1$ , respectively. Based on the voting strategy, we define the similarity between the two images,  $I_1$  and  $I_2$ , as:

$$\text{Sim}(I_1, I_2) = (V_1 - V_2 + 1)/2, \dots, \dots, \dots (4)$$

where the range of  $(V_1 - V_2)$  is  $[-1, 1]$ , which can be shifted to  $[0,1]$ , resulting in the  $\text{Sim}(I_1, I_2)$  defined in Eq. (4). Larger  $\text{Sim}(I_1, I_2)$  indicates that more atoms from  $D_1$  learned from  $I_1$  can well represent the feature vectors  $y_{2j}$  extracted from  $I_2$ . This implies that a considerable amount of information (denoted by  $D_1$ ) presented in  $I_1$  can be extracted (via sparse coding by solving Eq.(3)) from  $I_2$ . On the other hand, smaller  $\text{Sim}(I_1, I_2)$  indicates most suitable atoms for representing  $y_{2j}$  extracted from  $I_2$  are from  $D_2$  learned from  $I_2$  itself. This implies that less/no information presented in  $I_1$  can be extracted from  $I_2$ . Hence, the larger the  $\text{Sim}(I_1, I_2)$  is, the more similar the images  $I_1$  and  $I_2$  are. Obviously, if  $I_1$  is visually very different from  $I_2$ ,  $V_2$  is larger than  $V_1$ . Nevertheless, if  $I_1$  is visually similar to  $I_2$ ,  $V_2$  will not be larger than  $V_1$  in all instances. That is, better (or similar) reconstruction performance for  $y_{2j}$  may be achieved using  $D_1$  as the dictionary than using  $D_2$  due to some feature vectors extracted from  $I_2$  being able to be matched by the feature vectors extracted from  $I_1$ . To achieve this goal, we propose three rules for tuning the parameters used by K-SVD for learning the two dictionaries,  $D_1$  and  $D_2$ : (i) the number of the atoms in  $D_1$  should be larger than that in  $D_2$  ( $N_1 > N_2$ ); (ii) the number ( $J_1$ ) of iterations K-SVD performs for learning  $D_1$  should be larger than that ( $J_2$ ) for learning  $D_2$  ( $J_1 > J_2$ ); and (iii) the number of the target sparsity ( $L_1$ ), i.e., the number of nonzero coefficients for representing each feature vector for learning  $D_1$  should be larger than that ( $L_2$ ) for learning  $D_2$  ( $L_1 > L_2$ ). According to the rules designed above, when  $I_1$  is visually similar to  $I_2$  and  $D_1$  is finer than  $D_2$ , the  $l_1$ -minimizer for solving Eq. (3) may prefer more promising atoms from  $D_1$  than  $D_2$  to reconstruct  $y_{2j}$ , resulting in  $V_1 > V_2$  and larger  $\text{Sim}(I_1, I_2)$ .

Otherwise, when  $I_1$  is visually different from  $I_2$ , most atoms for reconstructing  $y_{2j}$  will still be selected from  $D_2$ , resulting in  $V_1 < V_2$  and smaller  $\text{Sim}(I_1, I_2)$ . The proposed FSRISA technique is summarized in Algorithm.

The major goal of performing sparse coding with respect to the dictionary consisting of  $D_1$  and  $D_2$ , instead of only  $D_1$  can be addressed as follows. When  $I_1$  (reference image) is visually different from  $I_2$  (test image),  $D_1$  and  $D_2$  are significantly different. In this scenario, the idea behind our FSRISA is somewhat related to that of sparse coding-based image classification approach, or sparse coding-based image decomposition approach. We use the similar concept to quantify the similarity between  $I_2$  and  $I_1$ , which may be interpreted as either (i) classifying  $I_2$  into  $I_1$  or  $I_2$  itself; or (ii) decomposing  $I_2$  into the components of  $I_1$  and/or those of  $I_2$  itself. When  $I_1$  is visually similar to  $I_2$ ,  $D_1$  and  $D_2$  are similar, which is enforced to that  $D_1$  is finer than  $D_2$  in FSRISA. Hence, the above discussions are also valid in this scenario. Moreover, why we don't perform sparse coding with respect to only one



dictionary  $D_1$  can be explained as follows. When performing sparse coding for the feature vectors of  $I_2$  with respect to a dictionary  $D_1$  consisting of atoms which may be not suitable for sparsely representing them, the sparse coding procedure still attempts to minimize the reconstruction errors. Based on our experience, it is usually not well distinguishable from reconstruction errors obtained with respect to either related or unrelated dictionaries. On the other hand, it is not easy to define a bounded score based on reconstruction error obtained by only one dictionary

## V. APPLICATIONS

In this section, we introduce three multimedia applications, including image copy detection, retrieval, and recognition, of the proposed FSRISA.

### A. Image Copy Detection via FSRISA

Digital images distributed through the Internet may suffer from several possible manipulations, such as (re)compression, noising, contrast/brightness adjusting, and geometrical operations. To ensure trustworthiness, image copy detection techniques have emerged to search for duplicates and forgeries.

Image copy detection can be achieved via content-based copy detection approach, which measures the similarity/distance between an original image and its possible copy version through comparing their extracted image features, SIFT-based features have been recently investigated. In this section, we study content-based image copy detection by applying the proposed FSRISA approach.

A user can perform image copy detection to detect possible copies of her/his original image from the Internet or an image database. To detect whether a test image  $I_2$  is actually a copy of a query image  $I_1$  with the dictionary feature  $D_1$  of size  $M \times N_1$ , we first extract the SIFT feature vectors  $y_{2j}, j = 1, 2, \dots, K_2$ , and learn the dictionary feature  $D_2$  of size  $M \times N_2$  of  $I_2$ , such that  $D_1$  is finer than  $D_2$ .

Then, we perform  $l_1$ -minimization by solving Eq.(3) for each  $y_{2j}$  with respect to  $D_{12} = [D_1|D_2]$ , and voting to get the percentages of votes,  $V_1$  and  $V_2$ , with respect to  $D_1$  and  $D_2$ , respectively. Finally, based on Eq. (4), the similarity between  $I_1$  and  $I_2$  can be calculated as  $\text{Sim}(I_1, I_2)$ . Given an empirically determined threshold  $\lambda$ , if  $\text{Sim}(I_1, I_2) \geq \lambda$ , then  $I_2$  can be determined as a copy version of  $I_1$ . Otherwise,  $I_1$  and  $I_2$  can be determined to be unrelated. The computational complexity for performing the FSRISA-based image copy detection can be also similarly analyzed.

### B. Image Retrieval via FSRISA

The most popular image retrieval approach is content-based image retrieval (CBIR), where the most common technique is to measure the similarity between two images by comparing their extracted image features.

In the proposed scheme, for a query image, we extract its dictionary feature (with  $N$  atoms) and transmit it to an image database, where each image is stored together with its dictionary feature and original SIFT feature vectors. For comparing the query image  $I_Q$  and each



database image  $ID_i$ ,  $i = 1, 2, \dots, size\_of\_database$ , where  $size\_of\_database$  denotes the total number of database images, we apply the proposed FSRISA scheme to perform the  $l_1$ -minimization (Eq. (3)) for each SIFT feature vector of  $ID_i$  with respect to the dictionary consisting of the dictionary features of the two images. Then, we calculate the reconstruction errors of all the stored feature vectors of  $ID_i$  and perform voting to get the similarity value between the two images (Eq. (4)) to be the score of  $ID_i$ .

Finally, we retrieve the top  $Q$  database images with the largest scores. Similarly, the computational complexity of comparing the two images can be analyzed except that the complexity for extracting the dictionary feature of each database image can be excluded due to the fact that the process can be performed in advance during database construction.

### C. Image Recognition via FSRISA

Consider a well-classified image database, where each class includes several images with the same object, but with different variations. Given a query image, a user may enquire which class the image belongs to. For image recognition, sparse representation techniques have been extensively used.

The major idea is to exploit the fact that the sparsest representation is naturally discriminative. Among all of the subsets of atoms in a dictionary, it selects the subset that most compactly expresses the input signal and rejects all of the others with less compact representations. More specifically, image recognition/classification can be achieved by representing the feature of the query image as a linear combination of those training samples from the same class in a dictionary.

Moreover, the conclusions in [1] claimed that their sparse representation-based face recognition algorithm should be extended to less constrained conditions (*e.g.*, variations in object pose or misalignment). In order not to incur such a constraint, both variability-invariant features and sparse representation should be properly integrated. In addition, in [1] a dictionary consists of several subsets of image features (down-sampled image pixels were used), where each subset contains the features of several training images belonging to the same class.

Nevertheless, if the number of classes, the number of training images in each class, or the feature dimension of a training image, is too large, the dictionary size will be very large. It will induce very high computational complexity in performing sparse coding for the feature vector(s) of a query image. We propose an image recognition approach, where we assess the similarity between a query image and each class of training images based on the proposed FSRISA. In the training stage, for the  $i$ -th image class,  $i = 1, 2, \dots, C$ , where  $C$  denotes the number of classes in an image database, we extract the SIFT feature vectors for each image as the training samples. Then, we apply K-SVD to learn the dictionary  $DC_i$  of size  $M \times NC_i$  to be the “dictionary feature” of the  $i$ -th image class, where  $M (= 128)$  denotes the length of a SIFT feature vector and  $NC_i$  denotes the number of atoms of  $DC_i$ . In the recognition stage, for a query image  $I_Q$ , we extract the SIFT feature vectors  $y_{Qj}$ ,  $j = 1, 2, \dots, K_Q$ , where  $K_Q$  denotes the number of SIFT feature vectors, and the dictionary feature  $D_Q$ .

Then, we apply FSRISA to assess the similarity between  $I_Q$  and the  $i$ -th image class,  $i = 1, 2, \dots, C$ , by performing the  $l_1$ -minimization (similar to Eq. (3)) to obtain the sparse representation coefficients  $x_{Qj}$  for  $y_{Qj}$  of  $I_Q$ , with respect to the dictionary  $DC_i$  consisting of  $DC_i$  and  $D_Q$ . Then, we calculate the reconstruction errors for  $y_{Qj}$  with respect



to  $DC_i$  and  $DQ$ , respectively, and perform voting for each  $yQ_j$ . Based on Eq. (4), we can calculate the similarity between  $IQ$  and the  $i$ -th image class, denoted by  $\text{Sim}(IQ, \text{Class-}i)$ .

Finally, the query image  $IQ$  can be determined to belong to the  $i$ -th class with the largest  $\text{Sim}(IQ, \text{Class-}i)$ . Moreover, the computational complexity for assessing the similarity between the query image  $IQ$  and the  $i$ -th class of images can be also approximately analyzed based, where the dictionary feature extraction for the  $i$ -th class of images can be performed in advance during database construction. In our image recognition approach, the sparse coding procedure should be performed for a query image and each image class, which is indeed computationally expensive. The complexity of our approach can be improved by applying more efficient sparse coding techniques, such as multi-core OMP.

## VI. CONCLUSION

Thus we have found similarity between two images. The core is to propose a feature-based image similarity assessment technique by exploring the two aspects of a feature detector in terms of representation and matching in our FSRISA framework. Then, we properly formulate the image copy detection, retrieval, and recognition problems as sparse representation problems and solve them based on our FSRISA.

## REFERENCES

1. Li-Wei Kang, Hsu, Chen, Chun-Shien Lu, Chih-Yang Lin and Soo-Chang Pei ,Feature-based Sparse Representation for Image Similarity Assessment, *Multimedia IEEE Transactions*, volume 13, Issue5, October 2011
2. L. W. Kang, C. Y. Hsu, H. W. Chen, and C. S. Lu, Secure SIFT-based sparse representation for image copy detection and recognition ,*Proc. of IEEE Int. Conf. on Multimedia and Expo*, Singapore, July 2010.
3. D. Nistér and H. Stewénus ,Scalable recognition with a vocabulary tree , *IEEE Conf. on Computer Vision and Pattern Recognition*,2006, pp. 2161–2168
4. D. G. Lowe , Distinctive image features from scale-invariant key points , *Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
5. Michal Aharon, Michael Elad, and Alfred Bruckstein,K-SVD: An Algorithm for Designing Over complete Dictionaries for Sparse Representation , *IEEE Transactions On Signal Processing*, VOL. 54, NO. 11, November2006.
6. Z. Wang and A. C. Bovik, “Mean squared error: love it or leave it? – A new look at signal fidelity measures,” *IEEE Signal Processing Magazine*,vol. 26, no. 1, pp. 98-117, Jan. 2009.
7. H. R. Sheikh and A. C. Bovik, “Image information and visual quality,”*IEEE Trans. on Image Processing*, vol.15, no.2, pp. 430-444, Feb. 2006.
8. Zhong Wu,Qifa Ke, JianSun,Heung-Yeung Shum,A Multi-Sample, Multi-Tree Approach to Bag-of-Words Image Representationfor Image Retrieval.

