Effect of local binary pattern on performance of iterative quantization in large-scale image retrieval

Mona Shakerdonyavi
Department of Computer Engineering ,University of Kharazmi,
Tehran, Iran

Abstract:
Hashing is an efficient algorithm in order to approximate nearest neighbor in large-scale image datasets. Learning Binary Code is one of the key steps to improve the performance of these algorithms, however, it is still a challenge in this area. This article has improved hashing algorithm performance with the use of appropriate inputs. In the proposed method,image features extract by local binary pattern Then, thay use as input vector of iterativa quantization (ITQ) that leads to more compact codes, less memory and lower computational cost. The reasons behind these achivements are the binary nature and proper functioning of the local binary pattern. Finally in order to rerank results based on most similarity to query image, bit importance reranking method is used and then big weights are assigned to important bits and small weights are assigned to minor bits and weighted hamming distance is used to compute similarity between query image and top results.To evaluate performance of proposed method we use CIFAR-10 and MNIST as dataset and precision vs recall curve as evaluation criteria. The simulations compare the new algorithm with three state of the art and along the line algorithms from three points of view; the hashing code size, memory space and computational cost, and the results demonstrate the effectiveness of the new approach.
Key words:
content-based image retrieval, large scale, local binary pattern, iterative quantization, hashing algorithm

1. Introduction

Increasing growth of images in all fields has led to creation of large databases of images. Because of this, we need to find a solution to conduct image retrieval process with high speed and accuracy. Images retrieval techniques can be divided into two general categories:

• Text-based image retrieval

• Content-based image retrieval

In the first method, a keyword, caption or description be added to images in the database as meta-data. Search for similar results should be done based on these data. This method has problems that cause large number of undesirable retrieval results for the user such as language ambiguities including polysemy or sequenced, need to describe each image in the form of words, high costs and inefficiencies in large databases, human errors like spelling mistakes and lack of labeling in all concepts in an image.

Due to the problems raised, researchers have begun to focus to provide efficient ways to perform content-based image retrieval. Content-based image retrieval includes a set of methods for processing visual characteristics of an image and search for similar images with that image in a database.

The challenges of content-based image retrieval in large-scale are high dimension of image descriptors, complexity and speed of search algorithms, which leads to an increase in memory requirements and a reduction in the speed of retrieval.

Image descriptors should have high discriminative power and low computational cost. In addition, it should generate inputs for nearest neighbor search algorithms. Algorithms that have this goal include local features[28],bag of visual words , Fischer vector[19] and vector of locally aggregated descriptors.

Search algorithms in high-dimensional databases should have high speed and accuracy. Due to the high volume of images in large databases, using brute force matching techniques (matching the requested images with all images) are not practical. Given this challenge, the nearest neighbor estimation techniques were developed and we can point out to conquer and divide Approach, KDE Trees and hashing algorithm.

Gong and Lozabnik[9][10] proposed a successful hashing technique. This is another simple and efficient method for finding the best rotation for the zero centerd data in order to minimize quantization error, when the data is mapped on the binary hypercube vertices.

Appropriate feature extraction of the image plays an important role in the performance of this algorithm. Local features of an image are more appropriate candidates than the general features order to describe an image; because it has more discriminative power, but its computational cost is more than general features. Among the local feature extraction methods, local binary pattern is a good choice

Manuscript received March 5, 2017
Manuscript revised March 20, 2017

due to the speed of production, binary nature, low computational cost and high discriminative power.

This paper presents a method for large-scale content-based image retrieval using local binary pattern and iterative quantization hashing algorithm. In this method, the Local Binary Pattern is used to extract image features. Since this model uses statistical and structural features in a context, it is a powerful tool to describe an image. This descriptor has advantages such as tolerance to uniform changes and simplicity in computing. The extracted features in this stage should be given to the iterative quantization hashing algorithm as the input. After carrying out this algorithm, it produces a binary code with fewer dimensions. K binary code generated in this way are independent from each other. For the k bits cannot be determined which of them are more important. Bit importance reranking method is used to determine the importance of each bit and results based on the most similarity to the requested image will be displayed to the user.

The structure of this paper is organized as follows:

In the next section, the algorithms used in this paper will be discussed. The third part deals to provide observations and experiments and conclusions presented at the end.

2. Background of theory
Local Binary Pattern

Local Binary Pattern operator has been introduced as a powerful descriptor for image texture[29]. This operator for each pixel generates a binary number according to the label adjacent pixels of 3 × 3. Labels obtained with the threshold value of neighboring pixels with center pixel values. In this case, number 1 should be given to the pixels which are greater than or equal to the amount of central pixel values and number 0 should be given to the pixels which are smaller than the central pixel values. These labels then placed on a rotating basis together and an 8-bit number would be formed. The function of this operator is shown in Figure 1.

LBP operator restrictions based small neighborhood of 3 × 3 that causes that it cannot dominate the images on a large scale. For this purpose, this operator was raised by a developing neighborhood that appears as a circle with radius R pixels on the P pixels. The operator is displayed as $LBP_{P,R}$ and can be up to 2^P different values, according to 2^P binary pattern produced by P pixels on the neighborhood radius R. Figure 2 shows the neighboring pixels in the local binary pattern in exchange for three different radius.

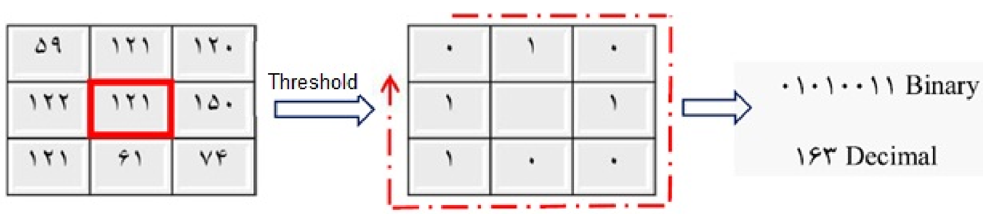


Fig. 1 local binary operator [29]

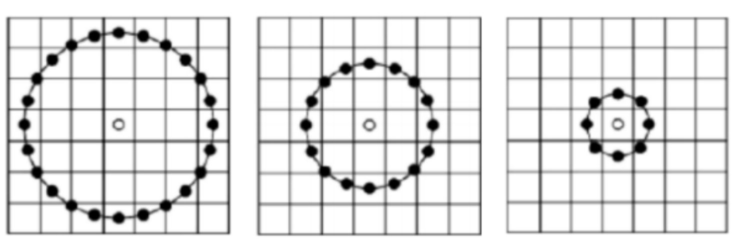


Fig. 2 LBP operator with different radius and neighbors [14].

By LBP operator, a histogram of labels be defined as follows:

$$H_i = \sum_{xy} I(f_i(x,y) = i), i = 0, \dots, n-1 \quad (1)$$

Where n is the number of labels generated by the operator LBP and the function of I be defined as the following equation:

$$I(A) = \begin{cases} 1 & A \text{ is true} \\ 0 & A \text{ is false} \end{cases} \quad (2)$$

Iterative quantization hashing algorithm

This is another simple and efficient method for finding the best rotation for the zero centered data in order to minimize quantization error, when the data is mapped on the binary hypercube vertices.

In order to find codes with the greatest variance and Incoherence between them, input vectors are written with an unsupervised method such as principal component analysis technique or CCA supervised techniques.

If $W \in R^{d\times q}$ is matrix coefficients which is obtained by PCA method, then every bit k = 1,..., q can be calculated with the following equation:

$$h_k(x) = sgn(xw_k) = sgn(v) \quad (3)$$

All of encryption process is as follows:

$$Y = sgn(XW) = sgn(V) \quad (4)$$

If W is an optimal solution, then WR in which R is an orthogonal matrix q × q is also optimized. Therefore, the mapping data V = XR become orthogonal. ITQ transforms the orthogonal form of data to minimize the quantization error.

$$Y = sgn(XWR) = sgn(VR) \quad (5)$$

Suppose $v \in R^d$ is a vector in the mapping space, sgn(v) is one of the vertices of the hypercube $\{-1,1\}^q$ which is close to v in Euclidean space.

Lack of quantization is the difference between vector v and its mapping in relation to the binary code in hypercube$\{-1,1\}^q$:

$$\|sgn(v) - v\|_2 \quad (6)$$

If the value is less than this lack, some better binary codes would be produced. This method is seeking orthogonal rotation mapped in such a way that the points be close to their peer binary code.

$$\min\big(Q(Y,R)\big) \quad (7)$$

$$Q(Y,R) = \|Y - VR\|_F \quad (8)$$

Figure 3 shows the ITQ algorithm performance[9][10]. The main idea of this method is to quantize each of the data points to the nearest vertices of the hypercube $(\pm 1, \pm 1)$. In the first image, vectors x and y are in accordance with the PCA direction. Quantization attributes the data points in a cluster to different vertices. In the second image (random rotation data) the data variance has been balanced and quantization errors is lower. In the third image, the optimal rotation obtained by using ITQ and quantization error is lower.

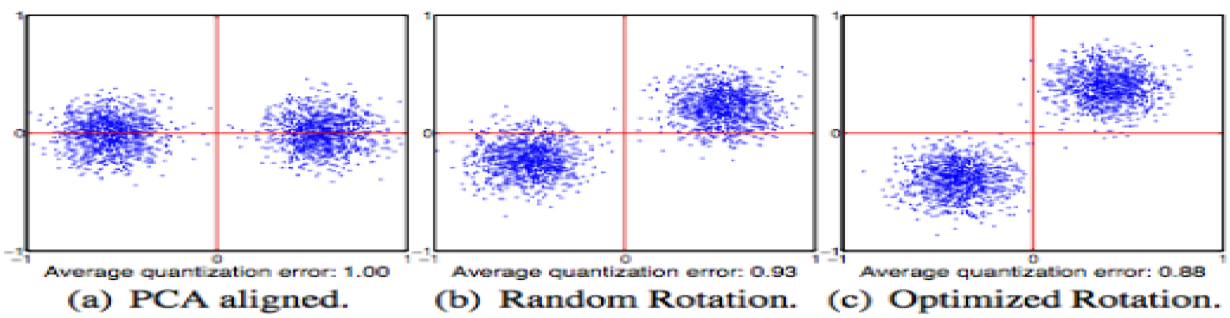


Fig. 3 performance of iterative quantization algorithm [10]

Maximizing the variance:

This method aims to produce code in which the variance of every bit is maximized and the bits are pairwise uncorrelated. This goal is achieved by maximizing the following objective function:

$$\tau(w) = \sum_k var\big(h_k(x)\big) = \sum_k var\big(sgn(xw_k)\big) \quad (9)$$

The variance would be maximized, when this encoding function produces exactly balanced bits; which means that $h_k(x) = 1$ for half of the data and $h_k(x) = -1$ for other half of the data.

Condition of maximizing the variance:

A hash function with maximum entropy $H\big(h_k(x)\big)$ need to maximize the variance of hash values, and vice versa:

$$H\big(h_k(x)\big) \leftrightarrow maxvar\big(h_k(x)\big) \quad (10)$$

Suppose likely attributable $h_{\,k}(x) = 1$ to data point is equal to p and likely attributable $h_{\,k}(x) = -1$ is equal to p-1. Then the entropy $H\big(h_k(x)\big)$ is obtained from the following equation:

$$H\big(h_k(x)\big) = -plog(p) - (1-p)\log(1-p) \quad (11)$$

The entropy value will be maximized, when the segmenting be done in a balanced way.

$$E(x) = M = 2P - 1 \quad (12)$$

$$var\big(h_{\,k}(x)\big) = (h_{\,k}(x) - M)^2 = 4P(1-P) \quad (13)$$

The maximum amount of variance occurs in P= 1/2, when this value is balanced in relation to the segmentation.

Sort Results based on importance of each bit

Binary code generated in the iterative qunatization hashing method are independent from each other and we cannot determine importance of each bit, so when calculating the Hamming distance equal weight given to each of the bits[8]. This method acts like a two-class classifier and in the classification process, some of the images which belongs to the category +1, would display with binary code 1 and other category which belongs to the category -1, would display with binary code 0. If a bit in a binary code is identical with its peer bits in a lot of the same images, it shows importance of that bit and more weight should be assigned to that.

Weight calculation:

Suppose that q is user's requested image and m is the number of images that have most similarity to user's requested image and calculated using the hashing technique. This m images can be displayed based on the Hamming distance as a concentric form and the center of the requested image and other images that their Hamming distance is equal to 0. Binary code of m images is as $H = \{H_1, \dots, H_m\}\ where\ \ H_i = \{H_i^{(1)}, H_i^{(2)}, \dots,\}$ and the binary code of the requested image is as $H_q^{(k)}\ where\ H_q = \{H_q^{(1)}, H_q^{(2)}, \dots\}$ and the weight of k-bit hash code in the requested image is as $w = \{w_1, \dots, w_k\}$ and its initial value is equal to $w_k = 1$.

In this section, H_i وH_q values are compared bit by bit. For k-th binary code, if $H_i^{(k)} = H_q^{(k)}$, then weight of w_k increases otherwise the weight would be reduced. Recursively, the weight of w whose iterations are equal to m will be updated. J-th iterations of w_k is calculated with the following equation:

$$w_k = \begin{cases} \varepsilon w_k, & \text{if } H_q^k \neq H_i^k \\ (1+\varepsilon) w_k, & \text{otherwise} \end{cases} \quad 0 < \varepsilon < 1 \quad (14)$$

ε is parameter in this equation. After m, the iteration amount of w obtained and can be used to calculate the weighted Hamming distance. The following algorithm describes how to calculate weights:

Algorithm 1-3: Weight calculation algorithm for each bit in the reordering based on the importance of each bit
Input: the binary code of user's requested image $H_q = \left\{h_q^{(1)}, h_q^{(2)}, \dots, h_q^{(k)}\right\}$; m images that have most similarity to user's requested image based on Hamming distance. The binary code of i-th retrieved image $H_i = \left\{H_i^{(1)}, H_i^{(2)}, \dots, H_i^{(k)}\right\}$; ε parameter Output: Weight $w = \{w_1, \dots, w_k\}$ for H_q 1: The weight of initial value $w_k = 1$ 2: For m: i=1 3: For K: k=1 4: Is k-th bit of H_q and H_i are equal? If yes, $w_k = (1+\varepsilon)w_k$ Otherwise $w_k = \varepsilon w_k$ 5: For finish 6: For finish

3. Proposed method

Figure 4 shows a diagram of the proposed method. In general, the proposed algorithm is divided into four stages including image feature extraction, creating compact binary code, obtain the nearest neighbor and reordering results.

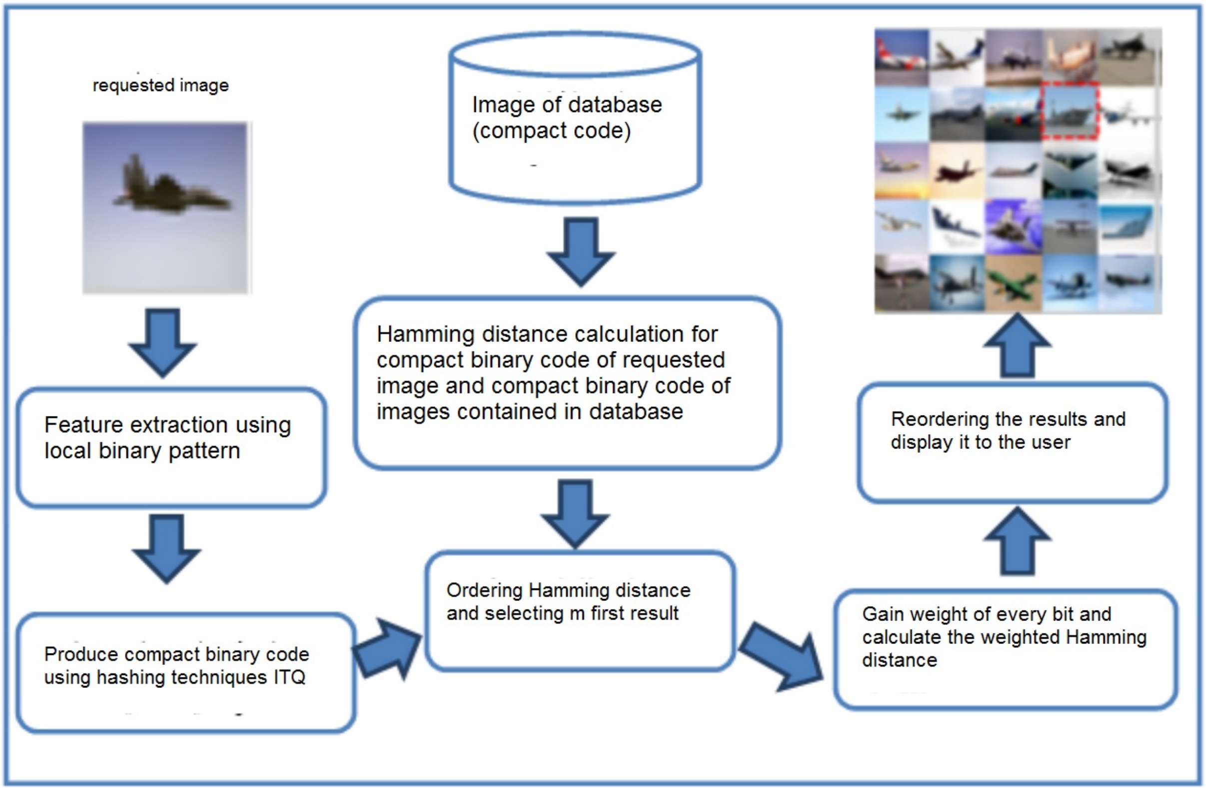


Fig. 4 Diagram of the proposed method

First stage, extraction image features

In order to describe an image, its features should be extracted. Local features of an image are more appropriate candidates than the global features order to describe an image; because it has more discriminative power, but its computational cost is more than global features. Due to the high volume of images in the database, the method used to extract local features should have the following characteristics:

- High discriminative power
- Low computational cost

Among the local feature extraction methods, local binary pattern is a good choice due to the speed of production, binary nature, low computational cost and high discriminative power. Local binary pattern as a descriptor is non-parametric. Since this model uses statistical and structural features in a context, it is a powerful tool to describe an image. This descriptor has advantages such as tolerance to uniform changes and simplicity in computing.

Due to the reasons mentioned, at this stage, local binary pattern used and image feature vectors generated in order to extract features of an images contained in database.

Second stage, production of compact binary code

Due to the high volume of images in large databases, using brute force matchin techniques, in which an image must compare with all images contained in a databases, are not practical, because this method requires a lot of memory and its computational cost is high. Hashing is an appropriate method to estimate the nearest neighbor where for each image, a compact binary code will be generated and finally, the images are compared on the basis of Hamming distance that this would enhance the computational speed and reduce the memory requirements.

At this stage, we use hashing algorithms ITQ to generate compact binary code. This algorithm is one of the most efficient algorithm to reduce quantization error. Feature vector generated in the previous stage is considered as input to the algorithm and dimentional of data reduced by PCA in unlabled data and canonical correlation in labeled data, the best rotation for the data obtained using singular value decomposition. Then each of the data has been projected to the nearest vertices of the hypercube and compact binary code is obtained for them.

Third stage, obtain nearest neighbor

Due to the nature of binary code generated in the previous step, the Hamming distance could be used as a benchmark to measure the distance between the requested image and the images in the database. Since this measure is a bitwise operator, it has a high speed. This is very important in large scale databases. At this stage, the distance between the request image's compact binary code and the compact binary code of image contained in the database obtained by using Hamming distance and the results would be ordered.

Fourth stage, reordering based on the importance of each bit

As mentioned in the previous section, the Hamming distance was used to measure the similarity between two binary codes. For k-bit binary code, hamming distance is an integer value and the maximum amount is equal to k. So a lot of images and the requested images have same Hamming distance. In the hamming distance, all the bits have the same value and the same is done with them, but in fact some bits are more important. In this regard, the importance of every bit at this stage is calculated using bit importance reranking method. For this purpose, the m first image obtained from the previous step selected and bits of compact binary code compared with the bits of requested image's binary code and in case of being equal, the weight of that bit would be increased and otherwise, it would be reduced. With the implementation of these algorithms for selected images, the weight of bits be obtained and using weighted hamming distance, the calculations be carried out and results be sorted and n first image displayed to the user.

Observations and experiments Dataset collection

In order to observe and evaluate proposed method MNIST and CIFAR 10 dataset have been used . CIFAR-10 contains 60,000 color images in the dimensions of 32 × 32. The images are categorized in 10 different classes. There are 6000 images in each class and this collection includes 50,000 training data and 10,000 test data. Figure 5 shows an example of the images in this database. MNIST contains the handwritten digits 0 to 9 with dimensions of 20 × 20. This collection contains 60,000 training data and 10,000 test data.

Figure 6 shows an example of the images contained in MNIST database.

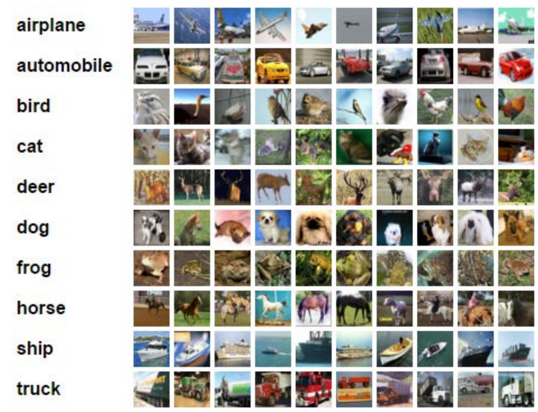


Fig. 5 Example of images in CIFAR-10 database

Fig. 6 Example of images in MNIST database

Evaluation criteria

precision: the proportion of retrieved images to the total number of retrieved images.

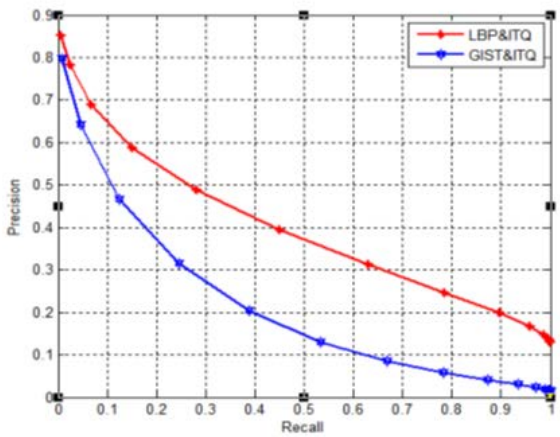
recall: the number of retrieved images to the total number of images in the database.

Precision-recall curves: precision is an important matter in the retrieval process and recall is an important matter inrecognition. This curve is used to display and compare the performance of algorithms. This curve shows how to reduce the precision, when a greater fraction of images contained in the database be restored. The ideal precision-recall curve has a 100% accuracy for all recall values which means retrieve all the related images.

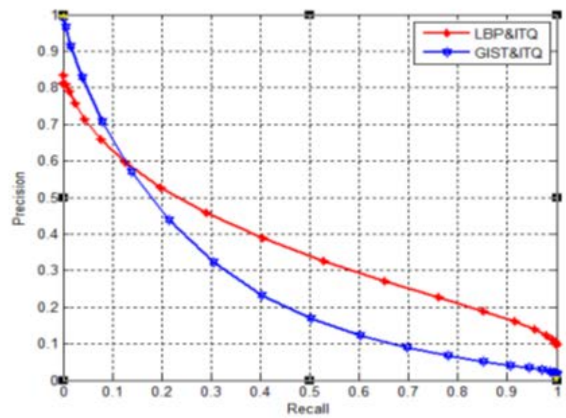
4. Results

The effect of Local Binary Pattern features to improve the performance of algorithm

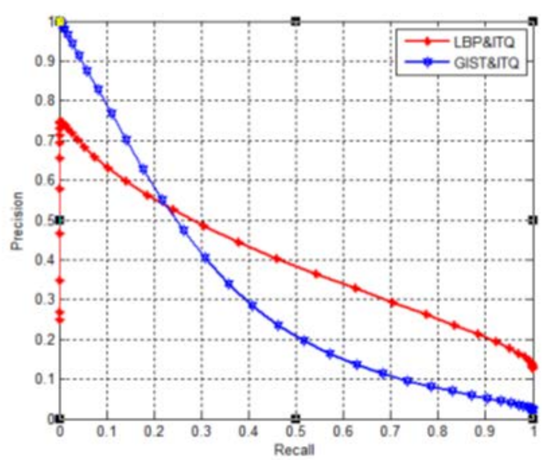
Vector image features as an input to hashing algorithm ITQ play a key role in performance of this algorithm.diagrams are presented in Figure 7 and 8, which are comparison between the performance of local binary pattern and the global descriptor GIST as an input to hashing algorithm ITQ. As can be seen in these figures, the performance of this algorithm has improved using local binary pattern as input. Due to the binary nature, this model has reduced computation cost and on the other hand, the retrieval accuracy has been increased, because this method is efficient to extract image features. By utilizing this method, the system obtained more accuracy with fewer hash bits, which is very important in large-scale content-based image retrieval systems.



(A) Precision vs. Recall @ 16 bit

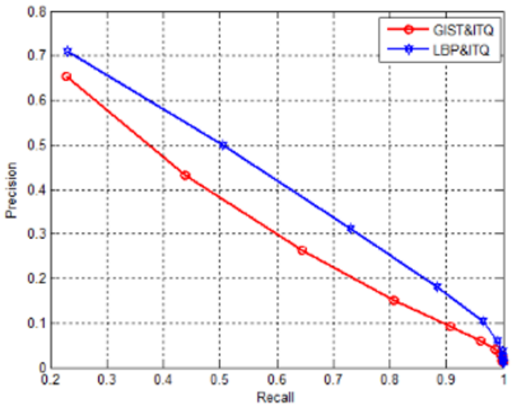


(B) Precision vs. Recall @ 32 bit

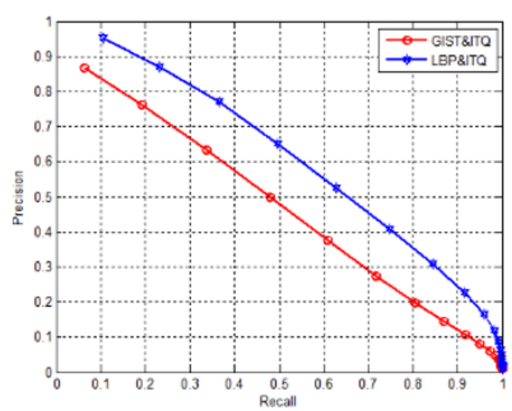


C) Precision vs. Recall @ 64-bit

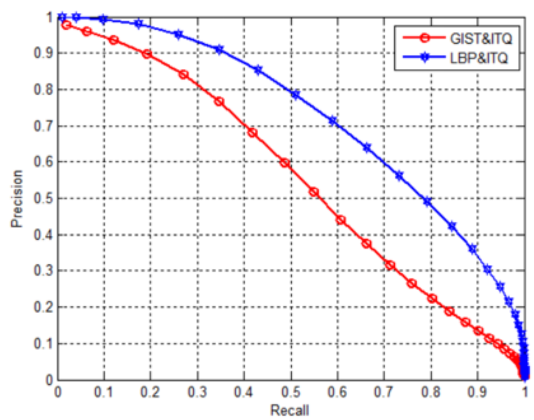
Fig. 7 The effect of local binary as input to the hashing algorithm ITQ in the database CIFAR-10



A) Precision vs. Recall @ 16 bit



B) Precision vs. Recall @ 32 bit



C) Precision vs. Recall @ 64-bit

Fig. 8 The effect of local binary as input to the hashing algorithm ITQ in the database MNIST

Compare the proposed method with other methods:

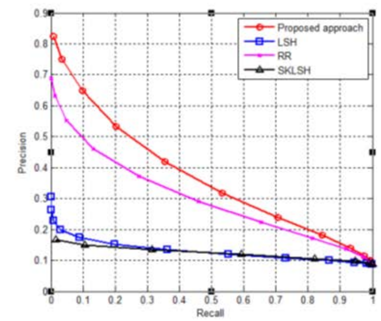
In this section, the proposed algorithm has been compared with three basic hashing algorithm that is follow the hashing process $B = sgn(X\overline{W})$ in which the matrix of image $\overline{W}$ obtained with different methods:

Local Sensitive Hashing(LSH): $\overline{W}$ is a Gaussian random matrix.

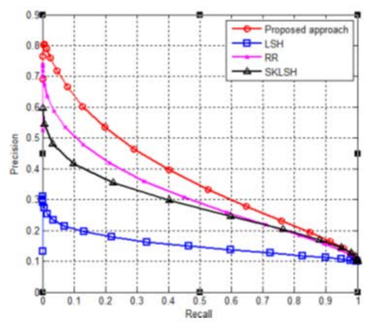
Random Rotation(RR): There is $\overline{W} = WR$ in this method where $\overline{W}$ is the matrix of PCA direction and R is a random orthogonal matrix .the idea of ITQ derived from it.

shift-invariant kernels: This method is based on random instant feature to estimate the Gaussian kernels.

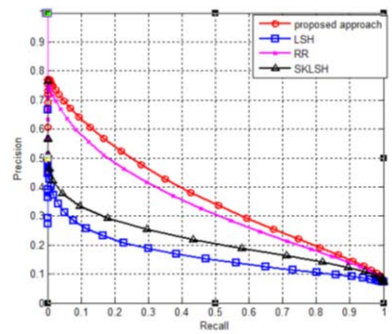
Figure 9 compare the performance of the proposed algorithm in a database CIFAR-10 and Figure 10 compares the performance of the proposed algorithm in a database MNIST and a diagram is presented to show precision-recall curves for the methods listed and proposed method. As can be seen in this diagram, the accuracy of the proposed algorithm is higher than the presented methods. In this way, with a lower number of hash bit, a higher accuracy is achieved.



(A) Precision vs. Recall @ 16 bit

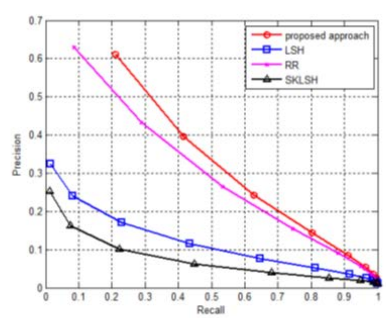


(B) Precision vs. Recall @ 32 bit

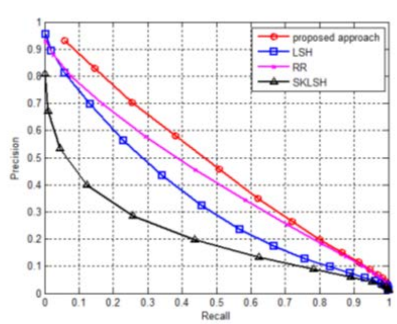


C) Precision vs. Recall @ 64 bit

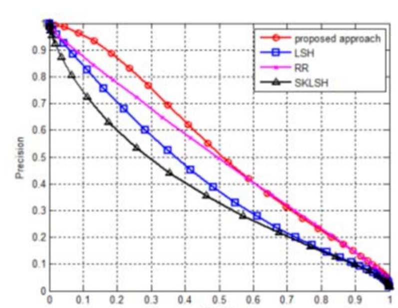
Fig. 9 Compare the performance of the proposed algorithm with other methods of data collection CIFAR-10



A) Precision vs. Recall @ 16 bit



B) Precision vs. Recall @ 32 bit

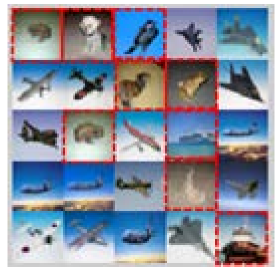
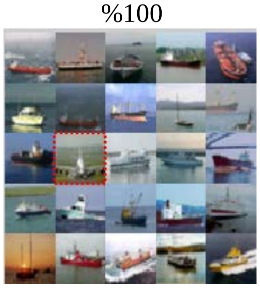
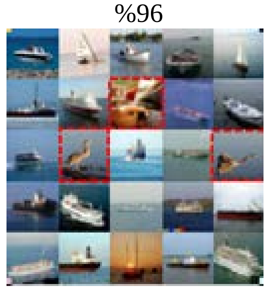
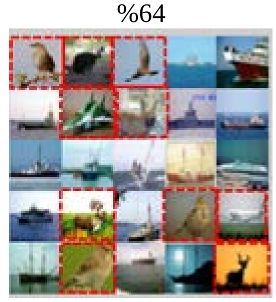
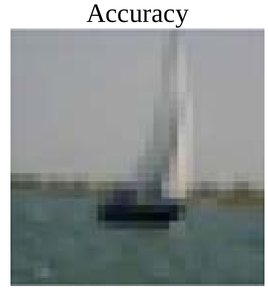
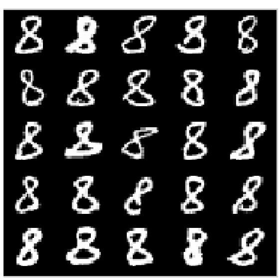
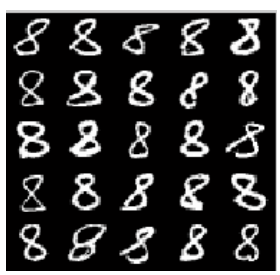
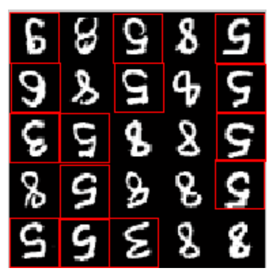
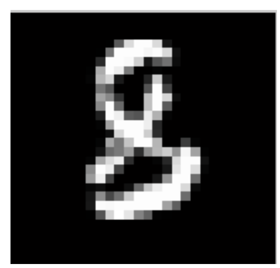
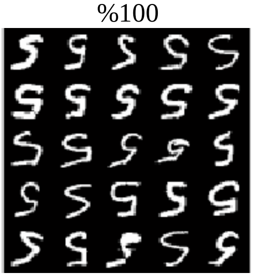
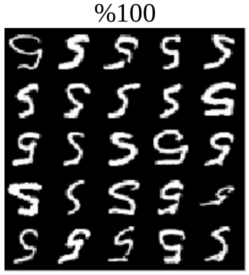
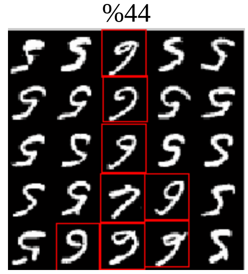
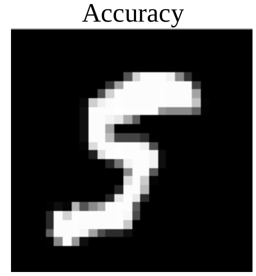


C) Precision vs. Recall @ 64 bit

Fig. 10 Comparison of the proposed algorithm with other methods of data collection MNIST

Display output to the user

In this section, by receiving an image as user request and with the implementation of the proposed algorithm, 25 first retrieved images displayed to the user. The following results were obtained using the technique of principal component analysis and canonical correlation analysis and computational accuracy of each is displayed. As can be seen, the algorithm has been enhanced after applying reordering based on the importance of every bit. The results are displayed according to the most similarity to the requested image.

The results after reordering	The results using PCA	The results using PCA	The requested image
			
%100	%96	%64	Accuracy
			
%96	%88	%60	Accuracy
			
%100	%100	%44	Accuracy
			
%100	%100	%72	Accuracy

5. Conclusion

This paper provided an efficient algorithm to carry out large scale content-based image retrieval process using local binary pattern and hashing algorithm ITQ and showed that image feature vector plays a key role as an input to hashing algorithm. In this method, the Local Binary Pattern is used to extract image features. Since this model uses statistical and structural features in a context, it is a powerful tool to describe an image. This descriptor has advantages such as tolerance to uniform changes and simplicity in computing and has been successfully applied in many areas of image analysis. Achieving small set of features is the most distinctive features of the local binary pattern that results in better performance and reduction in its dimensions. Choose an easy way to extract features of an image is very important due to the high volume of images contained in database. Because the method of choice must have high efficiency and low computational cost. Local Binary Pattern is simply calculated, because it is a non-parametric method for feature extraction. Thus,by using this method, we can achieve appropriate features with lower cost. ITQ technique have been used to estimate the nearest neighbor. It is also a simple and effective methods to estimate the nearest neighbor. Experimental results showed that the proposed method is more efficient than other methods. This means that we have been able to achieve higher accuracy using lower number of hash bit. Hash out an image with less code lead to use less memory to display an image and on the other hand, the computational cost to compare the two images will be reduce. This will be especially important in high-dimensional database, because it requires techniques that reduce memory consumption and computational cost.

Because the importance of each bit is not identical, using reordering results based on the importance of each bit, the weights of each bits obtained and utilizes the Hamming distance weighted criteria, and the results would reorder. This step in the proposed method is intended, because the results display based on the most similarity to the requested image.

According to the results obtained in testing, the accuracy of the proposed method is higher than existing methods. By applying lower number of hash code, this method could achieve a higher accuracy and this is a desirable result to carry large scale content-based image retrieval process.

Reference

[1] Ahonen, T., Hadid, A., and Pietikainen, M. Face description with local binary patterns: Application to face recognition. Pattern Analysis and Machine Intelligence, IEEE Transactions on 28, 12 (2006), 2037–2041.

[2] Ahonen, T., Matas, J., He, C., and Pietikäinen, M. Rotation invariant image description with local binary pattern histogram fourier features. in Image Analysis. Springer, 2009, pp. 61–70.

[3] Ahonen, T., and Pietikäinen, M. Soft histograms for local binary patterns. in Proceedings of the Finnish signal processing symposium, FINSIG (2007), volume 5, p. 1.

[4] Andoni, A., and Indyk, P. Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. in Foundations of Computer Science, 2006. FOCS'06. 47th Annual IEEE Symposium on (2006), IEEE, pp. 459–468.

[5] Bai, G., Zhu, Y., and Ding, Z. A hierarchical face recognition method based on local binary pattern. in 2008 congress on image and signal processing (2008), IEEE, pp. 610–614.

[6] Fehr, J. Rotational invariant uniform local binary patterns for full 3d volume texture analysis. in Finnish signal processing symposium (FINSIG) (2007).

[7] Fergus, R., Weiss, Y., and Torralba, A. Semi-supervised learning in gigantic image collections. in Advances in neural information processing systems (2009), pp. 522–530.

[8] Fu, H., Kong, X., and Wang, Z. Binary code reranking method with weighted hamming distance. Multimedia Tools and Applications (2014), 1–18.

[9] Gong, Y., and Lazebnik, S. Iterative quantization: A procrustean approach to learning binary codes. in Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on (2011), IEEE, pp. 817–824.

[10] Gong, Y., Lazebnik, S., Gordo, A., and Perronnin, F. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. Pattern Analysis and Machine Intelligence, IEEE Transactions on 35, 12 (2013), 2916–2929.

[11] Guo, Z., Zhang, L., and Zhang, D. A completed modeling of local binary pattern operator for texture classification. Image Processing, IEEE Transactions on 19, 6 (2010), 1657–1663.

[12] He, L., Zou, C., Zhao, L., and Hu, D. An enhanced lbp feature based on facial expression recognition. in Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the (2006), IEEE, pp. 3300–3303.

[13] Heikkilä, M., Pietikäinen, M., and Schmid, C. Description of interest regions with local binary patterns. Pattern recognition 42, 3 (2009), 425–436.

[14] Huang, D., Shan, C., Ardabilian, M., Wang, Y., and Chen, L. Local binary patterns and its application to facial image analysis: a survey. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on 41, 6 (2011), 765–781.

[15] Huang, D., Wang, Y., and Wang, Y. A robust method for near infrared face recognition based on extended local binary pattern. in Advances in Visual Computing. Springer, 2007, pp.437–446.

[16] Huang, D., Zhang, G., Ardabilian, M., Wang, Y., and Chen, L. 3d face recognition using distinctiveness enhanced facial representations and local feature hybrid matching. In Biometrics: Theory Applications and Systems (BTAS), 2010 Fourth IEEE International Conference on (2010), IEEE, pp. 1–7.

[17] Huang, Y., Wang, Y., and Tan, T. Combining statistics of geometrical and correlative features for 3d face recognition. in BMVC (2006), Citeseer, pp. 879–888.

[18] Indyk, P., and Motwani, R. Approximate nearest neighbors: towards removing the curse of dimensionality. in Proceedings of the thirtieth annual ACM symposium on Theory of computing (1998), ACM, pp. 604–613.

[19] Jaakkola, T., Haussler, D., et al. Exploiting generative models in discriminative classifiers. Advances in neural information processing systems (1999), 487–493.

[20] Jégou, H., Douze, M., Schmid, C., and Pérez, P. Aggregating local descriptors into a compact image representation. in Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on (2010), IEEE, pp. 3304–3311.

[21] Jégou, H., Perronnin, F., Douze, M., Sanchez, J., Perez, P., and Schmid, C. Aggregating local image descriptors into

compact codes. Pattern Analysis and Machine Intelligence, IEEE Transactions on 34, 9 (2012), 1704–1716.

[22] Jin, H., Liu, Q., Lu, H., and Tong, X. Face detection using improved lbp under Bayesian framework. in Multi-Agent Security and Survivability, 2004 IEEE First Symposium on (2004), IEEE, pp. 306–309.

[23] Kamel, A., Mahdi, Y. B., and Hussain, K. F. Multi-bin search: improved large-scale contentbased image retrieval. in Image Processing (ICIP), 2013 20th IEEE International Conference on (2013), IEEE, pp. 2597–2601.

[24] Lei, Z., Liao, S., He, R., Pietikainen, M., and Li, S. Z. Gabor volume based local binary pattern for face representation and recognition. in Automatic Face & Gesture Recognition, 2008. FG'08. 8th IEEE International Conference on (2008), IEEE, pp. 1–6.

[25] Liao, S., Zhu, X., Lei, Z., Zhang, L., and Li, S. Z. Learning multi-scale block local binary patterns for face recognition. in Advances in Biometrics. Springer, 2007, pp. 828–837.

[26] Liao, S., and Chung, A. C. Face recognition by using elongated local binary patterns with average maximum distance gradient magnitude. in Computer Vision–ACCV 2007. Springer, 2007, pp. 672–679.

[27] Lowe, D. G. Distinctive image features from scale-invariant keypoints. International journal of computer vision 60, 2 (2004), 91–110.

[28] Mikolajczyk, K., and Schmid, C. A performance evaluation of local descriptors. Pattern Analysis and Machine Intelligence, IEEE Transactions on 27, 10 (2005), 1615–1630.

[29] Ojala, T., Pietikäinen, M., and Harwood, D. A comparative study of texture measures with classification based on featured distributions. Pattern recognition 29, 1 (1996), 51–59.

[30] Ojala, T., Pietikäinen, M., and Mäenpää, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. Pattern Analysis and Machine Intelligence, IEEE Transactions on 24, 7 (2002), 971–987.

[31] Oliva, A., and Torralba, A. Modeling the shape of the scene: A holistic representation of the spatial envelope. International journal of computer vision 42, 3 (2001), 145–175.

[32] Paulhac, L., Makris, P., and Ramel, J.-Y. Comparison between 2d and 3d local binary pattern methods for characterisation of three-dimensional textures. in Image Analysis and Recognition. Springer, 2008, pp. 670–679.

[33] Perronnin, F., and Dance, C. Fisher kernels on visual vocabularies for image categorization. in Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on (2007), IEEE, pp. 1–8.

[34] Raginsky, M., and Lazebnik, S. Locality-sensitive binary codes from shift-invariant kernels. in Advances in neural information processing systems (2009), pp. 1509–1517.

[35] Ruiz-del Solar, J., and Quinteros, J. Illumination compensation and normalization in eigenspace-based face recognition: A comparative study of different pre-processing approaches. Pattern Recognition Letters 29, 14 (2008), 1966–1979.

[36] Shan, S., Zhang, W., Su, Y., Chen, X., and Gao, W. Ensemble of piecewise fda based on spatial histograms of local (gabor) binary patterns for face recognition. in Pattern Recognition, 2006. ICPR 2006. 18th International Conference on (2006), volume 4, IEEE, pp. 606–609.

[37] Singh, R., Vatsa, M., and Noore, A. Integrated multilevel image fusion and match score fusion of visible and infrared face images for robust face recognition. Pattern Recognition 41, 3 (2008), 880–893.

[38] Sivic, J., and Zisserman, A. Video google: A text retrieval approach to object matching in videos. in Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on (2003), IEEE, pp. 1470–1477.

[39] Tan, X., and Triggs, B. Enhanced local texture feature sets for face recognition under difficult lighting conditions. in Analysis and Modeling of Faces and Gestures. Springer, 2007, pp. 168–182.

[40] Tan, X., and Triggs, B. Fusing gabor and lbp feature sets for kernel-based face recognition. in Analysis and Modeling of Faces and Gestures. Springer, 2007, pp. 235–249.

[41] Tsaneva, M., Krezhova, D., and Yanev, T. Development and testing of a statistical texture model for land cover classification of the black sea region with modis imagery. Advances in Space Research 46, 7 (2010), 872–878.

[42] Vasconcelos, N. From pixels to semantic spaces: Advances in content-based image retrieval. Computer, 7 (2007), 20–26.

[43] Wang, J., Shen, H. T., Song, J., and Ji, J. Hashing for similarity search: A survey. arXiv preprint arXiv:1408.2927 (2014).

[44] Weiss, Y., Torralba, A., and Fergus, R. Spectral hashing. in Advances in neural information processing systems (2009), pp. 1753–1760.

[45] Wolf, L., Hassner, T., and Taigman, Y. Descriptor based methods in the wild. in Workshop on Faces in'Real-Life'Images: Detection, Alignment, and Recognition (2008).

[46] Yan, S., Wang, H., Tang, X., and Huang, T. Exploring feature descritors for face recognition. in Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on (2007), volume 1, IEEE, pp. I–629.

[47] Yang, H., and Wang, Y. A lbp-based face recognition method with hamming distance constraint. in Image and Graphics, 2007. ICIG 2007. Fourth International Conference on (2007), IEEE, pp. 645–649.

[48] Zhang, L., Chu, R., Xiang, S., Liao, S., and Li, S. Z. Face detection based on multi-block lbp representation. in Advances in biometrics. Springer, 2007, pp. 11–18.

[49] Zhang, W., Shan, S., Gao, W., Chen, X., and Zhang, H. Local gabor binary pattern histogram sequence (lgbphs): A novel non-statistical model for face representation and recognition. In Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on (2005), volume 1, IEEE, pp. 786–791.

[50] Zhao, G., and Pietikainen, M. Dynamic texture recognition using local binary patterns with an application to facial expressions. Pattern Analysis and Machine Intelligence, IEEE Transactions on 29, 6 (2007), 915–928.

[51] Zhao, G., and Pietikäinen, M. Experiments with facial expression recognition using spatiotemporal local binary patterns. in Multimedia and Expo, 2007 IEEE International Conference on (2007), IEEE, pp. 1091–1094.

[52] Zhao, H., Wang, Z., and Liu, P. The ordinal relation preserving binary codes. Pattern Recognition (2015).