Navigation

- Home
- Photos
- Committee
- Call for papers ^r
- Paper Submission
- Important Dates
- Keynote Speakers
- Workshop
- Registration
- Program
- Accommodation
- Prizes
- Sponsors
- Presentation Guidelines
- Previous Conferences
- Contact

DICTA 2015 Conference

Latest News

11 Feb: DICTA 2015 Conference [photos added!](#)

05 Dec: Goodbye all DICTA2015 delegates and see you at DICTA 2016: Gold Coast, Australia, 30th Nov to 2nd Dec. More details coming soon!

02 Nov: Only 3 weeks until DICTA 2015 the [preliminary program](#) is now available!

Conference Dates: 23rd - 25th November, 2015

Conference Venue: Adelaide Town Hall, Adelaide, South Australia, Australia

The International Conference on Digital Image Computing: Techniques and Applications (DICTA) is the main Australian Conference on computer vision, image processing, pattern recognition, and related areas. DICTA was established in 1991 as the premier conference of the Australian Pattern Recognition Society (APRS). DICTA 2015 is endorsed by the IAPR and technically co-sponsored by the IEEE. The proceedings of the conference will be submitted for inclusion to [IEEE Xplore](#).

The DICTA 2015 committee welcomes international keynote speakers [Prof. Bjarne K. Ersbøll](#) (DTU), [Dr. Francois Chaumette](#) (INRIA), [Prof. Manik Varma](#) (Microsoft Research India) and [Prof. Vincent Lepetit](#) (TU Graz).

Topics include, but are not limited to, the following:

- Image coding and processing
- Shape and texture analysis
- Surveillance, Defence and industrial applications
- Remote sensing and spectral imaging
- Astronomical image analysis
- Statistical and structural pattern recognition
- Machine learning
- Computer vision
- Human-computer interaction
- Biomedical and e-health applications
- Content-based image retrieval and image databases

Work-in-progress Workshop

An interactive workshop will be held at the event for early career researchers and PhD students to present work in progress.

Visiting Adelaide

Large-scale image retrieval using local binary patterns and iterative quantization

Mona Shakerdonyavi
School of Computer Engineering
University of Kharazmi
Tehran, Iran
Email: std_mona.shaker@khu.ac.ir

Jamshid Shanbehzadeh
School of Computer Engineering
University of Kharazmi
Tehran, Iran
Email: jamshid@khu.ac.ir

Abdolhossein Sarrafzadeh
Department of Computing
Unitec Institute of Technology
Auckland, New Zealand
Email: hsarrafzadeh@unitec.ac.nz

Abstract—Hashing algorithm is an efficient approximate searching algorithm for large-scale image retrieval. Learning binary code is a key step to improve its performance and it is still an ongoing challenge. The inputs of Hashing affects its performance. This paper proposes a method to improve the efficiency of learning binary code by improving the suitability of the Hashing algorithms inputs by employing local binary patterns in extracting image features. This approach results in more compact code, less memory and computational requirement and higher performance. The reasons behind these achievements are the binary nature and high efficiency in feature generation of local binary pattern. The performance analysis consists of using CIFAR-10 and precision vs. recall rate as dataset and evaluation criteria respectively. 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.

Keywords. Large-scale image retrieval, Local Binary Pattern, iterative quantization, image hashing algorithm.

I. INTRODUCTION

The widespread application of mobile devices and their image capturing capabilities results in exponential growth in creation of large scale image datasets (LSID) and consequently the need for fast large scale content based image retrieval (LSCBIR). LSCBIR requires image descriptors and fast algorithms to find similar images. The image descriptors should not only be effective in image description but also be suitable input to fast algorithms to find the nearest neighbor images according to their content with low computational cost and memory requirement. Recently Hashing algorithms have been introduced as solutions for fast LSCBIR. These algorithms embed high-dimensional vector to Hamming space, and generate a short binary code, then compare the binary codes of two images to measure their similarity based on Hamming distance. Gong and Lazebnik [1] [2] proposed a successful Hashing algorithm by iterative quantization (ITQ) where each iteration rotates the data using an optimal rotation matrix to minimize the quantization error. This algorithm showed higher performance in LSCBIR in terms of code size, memory space and computational cost. Encoding images into useful representation has a key role to enhance the ITQ performance. Local invariant image features are suitable candidates instead

of the global features in describing the image concept. An appropriate candidate is local binary pattern where it provides appropriate binary image features with high accuracy suitable for fast algorithms. The reason is its binary nature and high discriminative power.

This paper proposes a large-scale image retrieval method based on local binary patterns and ITQ. This method uses LBP to extract image features. LBP is a simple image feature extractor that uses both statistical and structural features, so it is a powerful tool for texture analysis. LBP has gained much attention due to its low computational complexity, gray-scale and rotation invariance, robustness to illumination changes, and excellent performance in many applications. It compares the gray level of a pixel and its local neighborhood and generates a binary pattern code. Binary pattern codes are often summarized into a histogram, and a bin in the histogram corresponds to a unique binary code [3]. Then these codes project to binary space by iterative quantization. ITQ [1] is one of the simple and efficient techniques to find the most appropriate rotation for the zero-centered data. It minimizes quantization error during mapping the data to the vertices of the binary Hypercube. It uses both unsupervised data embedding such as PCA and supervised embedding such as CCA to find codes with the highest variance and inconsistency between them. The rest of this paper is organized as follows. Next section explains image features extraction by LBP. Section III explains ITQ algorithm in detail. Section IV talks about GIST feature extractor. Section V explains the proposed approach. Section VI demonstrates the simulation results. Section VII is the conclusion of this paper.

II. LOCAL BINARY PATTERN

LBP operator measures the local constraint in texture analysis [4]. Its basic principle is to define the center pixel by its gray value relationship with its local neighborhood. Fig.1 shows how LBP compares the gray level of the center pixel with its 8-neighborhood ones. If the neighboring gray level value is greater than or equal to the center pixel value, this pixel takes the value 1 otherwise it takes the value 0..

Fig. 1: An example of a local binary pattern operator [3]

The limitation of the basic LBP is considering only the eight neighbors of a pixel. To overcome this limitation, the definition has been extended to include all circular neighborhoods with any number of pixels [4]. Fig.2 shows an example of the extended LBP that is called the elongated local binary pattern (ELBP) with two parameters: P and R. P is the number of circular neighborhood pixels and R is the radius of the circle.

Fig. 2: An example of an elongated local binary pattern operator [3]

In Eq.1, each pixel in the input image is compared against P equally spaced pixels forming a circle of a certain radius R. Then, the LBP value for a given pixel located at position (i, j) is computed. The mathematical formulation of LBP for a pixel is as follows:

$$LBP_{P,R}(i, j) = \sum_{P=0}^{P-1} (S(I(P) - I(i, j))2^P) \quad (1)$$

Eq.2 calculates function S:

$$S(X) = \begin{cases} 1, & X > 0 \\ 0, & X < 0 \end{cases} \quad (2)$$

In Eq.1, (i, j) is the location of center pixel, p and 2^P denote the equally spaced pixels around position (i, j) and I is the pixel intensity value .

III. ITERATIVE QUANTIZATION

A. Algorithm

ITQ [1] [2] is a simple and efficient method to find appropriate rotation of the zero-centered data. It minimizes the quantization errors during data mapping to the vertices of the binary Hypercube. After centering the input feature vectors, in order to find codes with maximum variance and pairwise uncorrelated, input data $X \in R^{(n \times d)}$ is projected using an unsupervised data embedding such as PCA, even though it can be used with supervised embedding such as CCA. PCA algorithm applies on the data points and formulate the problem of learning a good and appropriate binary code in the form of minimizing the quantization error of the projected data to

each of the vertex of the binary cube. If $W \in R^{d \times q}$ is the coefficient matrix that is obtained by PCA method, then Eq.3 encodes each bit of $k = (1, \dots, q)$.

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

The entire encoding process would be:

$$Y = \text{sgn}(XW) = \text{sgn}(V) \quad (4)$$

If W is an optimal solution, then WR , where R is an orthogonal matrix of $q \times q$ is also optimized. Therefore the projected data $V = XR$ also transform into orthogonal. ITQ orthogonally transform the projected data to minimize the quantization error.

$$Y = \text{sgn}(XWR) = \text{sgn}(VR) \quad (5)$$

Suppose $v \in X^q$ is a vector in the projected space. It is easy to show that $\text{sgn}(v)$ is one of the vertex of the hypercube $\{-1, 1\}^q$ which is close to v in the Euclidean distance. Quantization loss is the difference between v and real projected v into binary hypercube $\{-1, 1\}^q$.

$$\|\text{sgn}(v) - v\|_2 \quad (6)$$

Better binary codes produce when the amount of quantization loss is less, therefore they are looking for orthogonal rotation, so that the projected points are closest as possible to their binary quantization.

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

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

This section refers to the issue of learning the binary codes without any information. First, a linear dimension reduction is applied to the data and then the binary quantization is applied to the result space.

B. Maximize the variance

They are looking for efficient codes in which the variance of every bit is maximum and the bits are mutually uncorrelated, this goal can achieve by maximizing the following objective function:

$$\tau(w) = \sum_k \text{var}(h_k(x)) = \sum_k \text{var}(\text{sgn}(xw_k)) \quad (9)$$

$$\frac{1}{n} B^T B \quad (10)$$

Variance is maximized only when the coding functions exactly produce balanced bits. That will happen when for half of the data $h_k(x) = 1$ and for other half of the data $h_k(x) = -1$.

1) *The condition of maximizing the variance:* A hash function with a maximum entropy of $H(h_k(x))$ should maximize the variance of hash values and vice versa:

$$H(h_k(x)) \leftrightarrow \max var(h_k(x)) \quad (11)$$

Let h_k be the probability of assigning $h_k(x) = 1$ to a data point is equal to p and the probability of assigning $h_k(x) = -1$ is equal to $1 - p$, the entropy value is obtained through the following relationship:

$$H(h_k(x)) = -p \log(p) - (1 - p) \log(1 - p) \quad (12)$$

It is easy to show that the entropy value is maximum when the balanced segmentation is done. It leads to maximizing the variance of bits.

$$E(x) = M = 2P - 1 \quad (13)$$

$$var[h_k(x)] = (h_k(x) - M)^2 = 4P(1 - P) \quad (14)$$

The maximum amount of variance occurs in $P = 1/2$ and this is concept of balanced segmentation.

IV. GIST GLOBAL DESCRIPTOR

Oliva and Torralba [5] proposed The GIST descriptor. GIST develops a low dimensional representation of the scene without segmentation by a set of perceptual dimensions (naturalness, openness, roughness, expansion, ruggedness) that represent the dominant spatial structure of a scene. They show that these dimensions may be reliably estimated using spectral and coarsely localized information.

The following steps show the computation of GIST descriptor:

- The image convolves with 32 Gabor filters at 4 scales and 8 orientations and producing 32 feature maps of the same size of the input image.
- Each feature map divides into 16 regions (by a 4x4 grid), and then average the feature values within each region.
- Concatenate the 16 averaged values of all 32 feature maps, resulting in a 16x32=512 GIST descriptor.

V. PROPOSED APPROACH

Fig.3 shows diagram of the proposed approach. The steps are as follows:

- 1) Image features extract by LBP.
- 2) Centering the input feature vectors at first, then use them as inputs of ITQ hashing algorithm and generate compact code for images.
- 3) Use hamming distance to measure similarity between query image compact code and compact code of images in dataset.
- 4) Sort the hamming distance results and show 25 top results that are closest to query image.

Fig. 3: Diagram of the proposed method

VI. EXPERIMENTAL RESULTS

We examined three different aspects of the proposed approach: Hashing code size, memory space and computational cost.

A. Database

The proposed approach has been evaluated on CIFAR-10. It consists of a total of 60,000 32×32 color images in 10 classes with different natural scenery or objects.

B. Measures

We evaluate performance of proposed method by Precision vs. Recall curves.

Precision vs. Recall curves: Precision is ratio of relevant images to the total number of images retrieved in the query and Recall is ratio of relevant images retrieved in a query to the total number of images in the database. These curves are used to represent and compare the performance of a system. The curves show, in general, how precision decreases as larger parts of the image database are retrieved.

C. Results

1) *Comparisons with state-of-the-art hashing methods:* Fig.4 compares the performance of the proposed method with the following state-of-the-art hashing algorithms:

- 1) LSH [6] uses random matrix with Gaussian distribution to project data onto binary space.
- 2) RR [7] balances the variance between different dimensions of the data by rotating the PCA-projected data with a random matrix.
- 3) SKLH [8] is based on random Fourier features for estimating the Gaussian kernels.

As shown in Fig.4 our method outperforms the other three state of the art methods. By employing suitable inputs we can achieve to high precision by small size of hashing codes that leads to low computational cost and memory requirement.

(a) Precision vs. Recall @16bit

(b) Precision vs. Recall @32bit

(c) Precision vs. Recall @64bit

Fig. 4: Comparison proposed method with state-of-the-art methods

(a) Precision vs. Recall @16bit

(b) Precision vs. Recall @32bit

(c) Precision vs. Recall @64bit

Fig. 5: Comparison LBP&ITQ and GIST&ITQ

Feature extractor method	LBP	LBP	GIST
Data projection method	PCA	CCA	PCA

precision	64%	96%	44%

precision	60%	88%	48%

TABLE I: Top 25 returned images

2) *Comparisons LBP with GIST as image features extractor*: In Fig.5, we want to show that the useful representation has a key role to enhance the ITQ performance. For this purpose we used LBP and GIST as image features extractor and compared the results. As shown in Fig.5, LBP as input vector of ITQ can achieve higher precision with small code length. The reason for this result is high efficiency in feature generation of local binary pattern.

3) *Image retrieval results*: Table.I demonstrates the image retrieval results of query sample in the CIFAR-10. The top 25 images are shown as the query results. The red rectangles denote false results.

VII. CONCLUSION

In this paper, we propose a method to improve the efficiency of learning binary code by improving the Hashing algorithms inputs. This method uses local binary patterns to generate feature vector as input of ITQ. Computational simplicity and discriminative power are the most distinctive characteristic of LBP. Its high efficiency in image features extraction leads to enhance ITQ performance. By generating Suitableness inputs, we can achieve to higher precision with small size of code. The main factors in LSCBIR are speed of retrieval and memory usage. This approach can represent an image by small code length that leads to speed of retrieval and loss of memory. Experiments show that, the accuracy of retrieval has been improved by employing LBP as input of ITQ and we can achieve to higher precision by small code length.

REFERENCES

- [1] Y. Gong and S. Lazebnik, "Iterative quantization: A procrustean approach to learning binary codes," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 817–824.
- [2] Y. Gong, S. Lazebnik, A. Gordo, and F. Perronnin, "Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 12, pp. 2916–2929, 2013.
- [3] S. Liao and A. C. Chung, "Face recognition by using elongated local binary patterns with average maximum distance gradient magnitude," in *Computer Vision-ACCV 2007*. Springer, 2007, pp. 672–679.
- [4] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 971–987, 2002.
- [5] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International journal of computer vision*, vol. 42, no. 3, pp. 145–175, 2001.
- [6] A. Andoni and P. Indyk, "Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions," in *Foundations of Computer Science, 2006. FOCS'06. 47th Annual IEEE Symposium on*. IEEE, 2006, pp. 459–468.
- [7] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 3304–3311.
- [8] M. Raginsky and S. Lazebnik, "Locality-sensitive binary codes from shift-invariant kernels," in *Advances in neural information processing systems*, 2009, pp. 1509–1517.
- [9] A. Kamel, Y. B. Mahdi, and K. F. Hussain, "Multi-bin search: improved large-scale content-based image retrieval," in *Image Processing (ICIP), 2013 20th IEEE International Conference on*. IEEE, 2013, pp. 2597–2601.
- [10] H. Fu, X. Kong, and Z. Wang, "Binary code reranking method with weighted hamming distance," *Multimedia Tools and Applications*, pp. 1–18, 2014.
- [11] P. Indyk and R. Motwani, "Approximate nearest neighbors: towards removing the curse of dimensionality," in *Proceedings of the thirtieth annual ACM symposium on Theory of computing*. ACM, 1998, pp. 604–613.
- [12] Y. Weiss, A. Torralba, and R. Fergus, "Spectral hashing," in *Advances in neural information processing systems*, 2009, pp. 1753–1760.
- [13] R. Fergus, Y. Weiss, and A. Torralba, "Semi-supervised learning in gigantic image collections," in *Advances in neural information processing systems*, 2009, pp. 522–530.
- [14] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [15] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, 2003, pp. 1470–1477.
- [16] F. Perronnin and C. Dance, "Fisher kernels on visual vocabularies for image categorization," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 2007, pp. 1–8.
- [17] H. Zhao, Z. Wang, and P. Liu, "The ordinal relation preserving binary codes," *Pattern Recognition*, 2015.