ISSN: 2348 - 2117

International Journal of Engineering Technology and Computer Research (IJETCR)
Available Online at www.ijetcr.org
Volume 2; Issue 6; Page No. 86-92

Survey on Visual Word Generation and Their Matching Techniques

Mr. S.N. Bhojane[1], Prof. P.R. Futane[2], Prof. K.R. Pathak[3]

[1]Student, dept. of computer Engineering, Sinhgad college of Engineering, Vadgaon, Pune 411041, India

sachin.vpcoe@gmail.com

[2] Head of Dept., dept. of computer Engineering, Sinhgad college of Engineering, Vadgaon, Pune 411041, India

prfutante.scoe@sinhgad.edu

[3] Professor., dept. of computer Engineering, Sinhgad college of Engineering, Vadgaon, Pune 411041, India

krpathak.scoe@sinhgad.edu

ABSTRACT

Content-Based Image Retrieval (CBIR), a technique which uses visual contents to search images from large scale image databases according to user's interests. Contents of image can be global features such as color, shape and texture or local features such as SIFT (Scale Invariant Feature Transform) features. Both types are complement for each other. Bag of visual words (BoVW) is a widely used approach in most of content based image retrieval. This approach is based on local features of images which forms clusters based on similarity of these features. These formed clusters are used for matching two similar images. This survey paper mainly concentrate on clustering of local features i.e. visual word generation and matching of similar visual words based on their characteristics. This paper surveys on recent studies on visual word generation and matching.

Key Words: Bag of Visual Words (BOW), CBIR, KD-tree

1 INTRODUCTION:

INTEREST in the potential of digital images has increased enormously over the last few years, by the rapid growth of imaging on the World-Wide Web. It is also discovering that the process of locating a desired image in a large and varied collection can be a source of considerable frustration. Content based image retrieval (CBIR) has been found to be a great solution in finding precise results for the given query images or finding a desired image from large collection of database images.

1.1 Content Based Image Retrieval system

In above title meaning of 'Content Based 'is finding results for given query image by analyzing contents of query image itself. Here 'Contents 'will be color, shape, texture and local features which are derived from that image itself. Content Based Image Retrieval systems are widely used in web-based image search engines as they are purely independent of meta-data. Thus a system which can search images based on their content will provide better indexing and will return more accurate results.

1.2 Image Features

CBIR system makes use of contents of query image and these contents are referred as "Features". Image features can be color, shape, texture as well as local features.

1.2.1 Color: In CBIR systems, color is the most widely feature as they are easy to extract and utilize. Color feature is independent of image size and orientation, and fairly robust to background complication. Each image added to the database is analyzed to compute a color histogram which shows the proportion of pixels of each color within the image. The color histogram for each image is then stored in the database. At search time, the user can either specify the desired proportion of each color (75 % olive green and 25% red, for example), or submit an example image from which a color histogram is calculated.

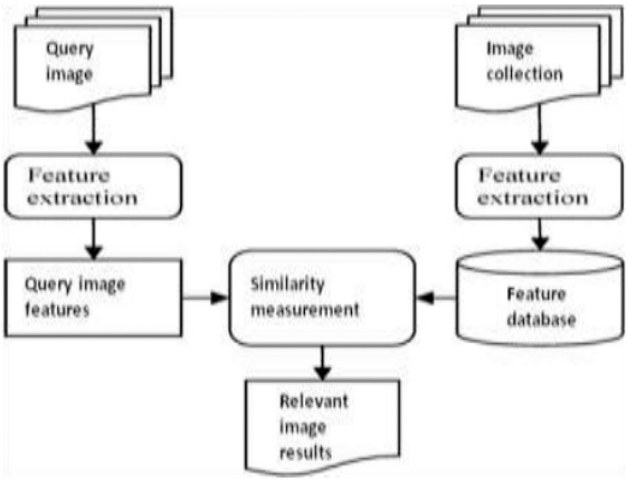

Figure 1: Block diagram of CBIR

1.2.2 Texture: Image texture gives us information about spatial arrangement of color and intensities of an image. Texture is one of the important features of natural images and refers to inborn surface properties of an entity and their association to the surrounding background. Texture can be represented by Grey Level Co-occurrence.

To get texture information of image directional features were pulled out. The six image texture properties were coarseness, regularity, directionality, contrast, line likeness and roughness. Commonly texture representation uses following methods:

1.2.3 Shape: Shape features of image object have widespread application in CBIR systems. Shape may be defined as the surface pattern of an object as an outline or contour. It allows an object to be distinguished from its surroundings by its outline. Shape can be represented based on two categories:

2. VISUAL WORD GENERATION:

Visual word generation is a clustering based approach to generate visual words from clustering of local features. Clustering techniques can be classified into supervised (including semi-supervised) and unsupervised schemes. The former consists of hierarchical approaches that demand human interaction to generate splitting criteria for clustering. In unsupervised classification, called clustering or exploratory data analysis, no labeled data are available The goal of clustering is to separate a finite unlabeled data set into a finite and discrete set of "natural", hidden data structures, rather than provide an

a) Structural methods describe texture by identifying structural primitives and their placement rules using morphological operator and adjacency graph. They deal with the arrangement of image primitives, presence of parallel or regularly spaced objects.

b) Statistical methods uses statistical distribution of the image intensity to characterize the texture by the popular co-occurrence matrix, Fourier power spectra, Shift invariant principal component analysis (SPCA), Tamura feature, Multi-resolution filtering technique such as Gabor and wavelet transform.

a) Boundary-based - Only outer boundary of the shape used for shape representation. This is done by relating the considered region using its external properties as the pixels along the object boundary.

b) Region-based - Entire shape region with its internal characteristics are used to describe shape i.e. the pixels contained in that region.

accurate characterization of unobserved samples generated from the same probability distribution.

Bag of Visual words approach is based on vector quantization process as used in text retrieval domain by clustering local features of local regions or points. This approach is based on unsupervised learning algorithm which is used to group features into W words. Assigning of visual features to their appropriate visual word is more complex task as visual features are high dimensional (e.g. SIFT [128-dimensions]) and finding exact match for it is not trivial. Following subsections surveys number feature assignment techniques.

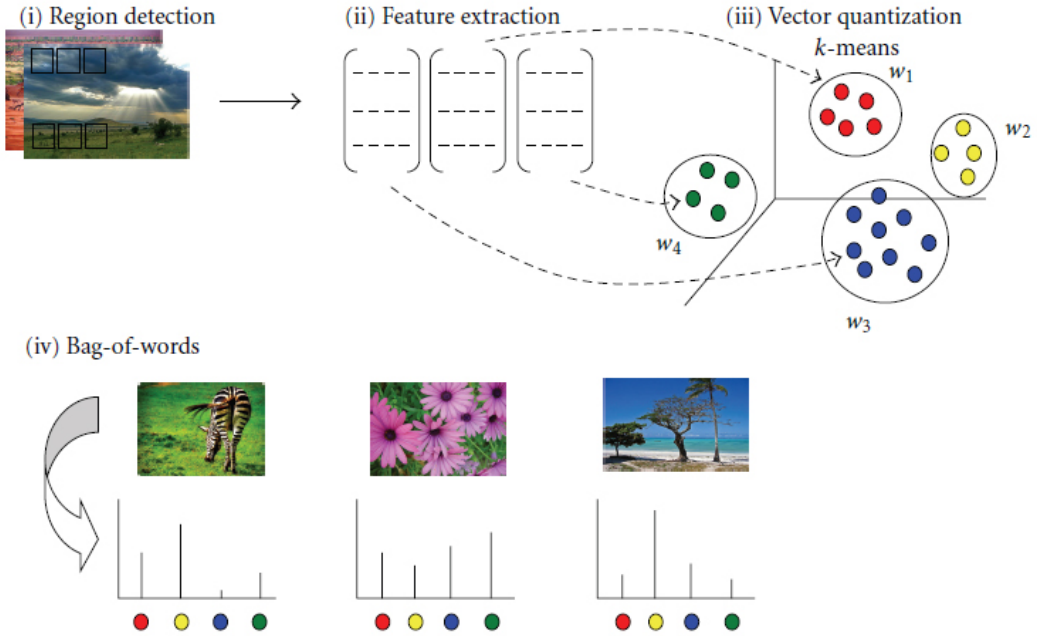

Figure 2: Visual word Generation

2.1 Codebook Approach:

Codebook Approach is basically derived from Text retrieval systems. This approach uses notion used in text retrieval systems for better retrieval of images. The objective is to vector quantize the descriptors into clusters which will be the visual word's for text retrieval.

2.1.1 Outline of Codebook approach

As in text retrieval each document is represented by a vector of word frequencies. However, it is usual to apply a weighting to the components of this vector, rather than use the frequency vector directly for indexing. We can use this analogy for visual words and their respective vectors. The standard weighting is known as 'term frequency document frequency', tf-idf, and is computed as follows. Suppose there is a vocabulary of k words, then each document is represented by a k-vector V_d = (t1. . . ti, . . . , tk)^T of weighted word frequencies with components

$$t_i = n_{id}/n_d \log N/n_i \qquad (1)$$

where n_{id} is the number of occurrences of word i in document d, n_d is the total number of words in the document d, n_i is the number of occurrences of term i in the whole database and N is the number of documents in the whole database. The weighting is a product of two terms: the word frequency n_{id}/n_d, and the inverse document frequency log N/n_i. The

Intuition is that word frequency weights words occurring often in a particular document, and thus describe it well, whilst the inverse document frequency down weights words that appear often in the database. At the retrieval stage documents are ranked by their normalized scalar product (cosine of angle) between the query vector V_q and all document vectors V_d in the database [12]. Query vector is given by the visual words contained in a user specified sub-part of an image, and the other images are ranked according to the similarity of their weighted vectors to this query vector.

Codebook approach is quite expensive approach to assign a visual feature to its appropriate visual word, for e.g. to assign a query feature with 128-dimensions into a codebook of size 1M and N visual words it will take $128*10^6*N$ time complexity. *Sequential index* technique improved time complexity by dividing feature space into several partitions and search entire codebook by nearest partition index. This technique reduced time complexity to $d * M/N*B$.

2.2 KD (K-Dimensional) Trees

Nearest neighbor search is an important component in many computer vision applications. KD trees are used for matching image descriptors to its nearest neighbor. In a typical application, a large number of SIFT descriptors extracted from one or many images are stored in a database. A query usually involves finding the best matched descriptor vector(s) in the database to a SIFT descriptor extracted from a query image. A useful data-structure for finding nearest-neighbor queries for image descriptors is the KD-tree, which is a form of balanced binary search tree. Notion of KD-tree is described below [9]:

1. The elements stored in the KD-tree are high-dimensional vectors.
2. At the first level (root) of the tree, the data is split into two halves by a hyper plane orthogonal to a chosen dimension at a threshold value.
3. By comparing the query vector with the partitioning value, it is easy to determine to which half of the data the query vector belongs.
4. Each of the two halves of the data is then recursively split in the same way to create a fully balanced binary tree. At the bottom of the tree, each tree node corresponds to a single point in the dataset; though in some implementation, the leaf nodes may contain more than one point.

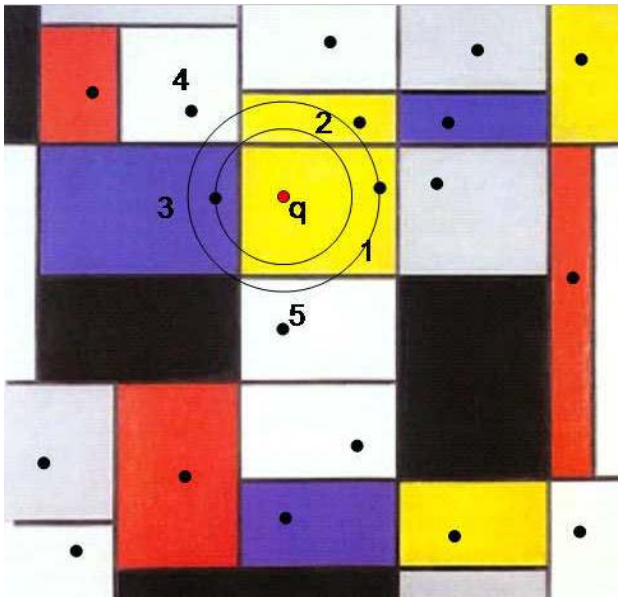

Figure 2: Priority search in KD tree

Searching in KD tree is given below with illustrative tree

1. In given fig.1, a query point is represented by the red dot and its closest neighbor lies in cell 3.

2. A priority search first descends the tree and finds the cell that contains the query point as the first candidate (label 1).

3. However, a point contained in this cell is often not the closest neighbor.

4. A priority search proceeds in order through other nodes of the tree in order of their distance from the query point that is, in this example through the nodes labeled 2 to 5.

5. The search is bounded by a hyper sphere of radius r (the distance between the query point and the best candidate).

6. The radius r is adjusted when a better candidate is found. When there are no more cells within this radius, the search terminates.

One of the disadvantages of KD-tree is that it does not allow balancing. If dimension 'd' of feature is more than complexity is no more better than brute force search. Randomized KD-trees were proposed, which will build with different parameters, with different ways to select partitioning value.

2.3 Hierarchical K-means

Basic Concept: Hierarchical K-means clustering combines the K-means algorithm and Hierarchical algorithm. The hierarchical k-means tree is constructed by splitting the data points at each level into K distinct regions using a k-means

Clustering, and then applying the same method recursively to the points in each region. We stop the recursion when the number of points in a region is smaller than K.

Algorithm: The algorithm is described as follows [3]:

1. Set X = $\{x_i | i = 1, ..., r\}$ as each data of A, where A = $\{a_i | i = 1, ..., n\}$ is attribute of n-dimensional vector.

2. Set K as the predefined number of clusters.

3. Determine p as numbers of computation.

4. Set i = 1 as initial counter.

5. Apply K-means algorithm.

6. Record the centroids of clustering results as Ci = $\{c_{ij} | j = 1, ...,K\}$

7. Increment i = i + 1

8. Repeat from step 5 while i < p.

9. Assume C = {C_i|i = 1, , p} as new data set, with K as predefined number of clusters.

10. Apply hierarchical algorithm.

K-means algorithm fails for non-linear dataset and unable to handle noisy data and outliers. Tree structures are all local optimum algorithms which cannot reach global optimum without considerable amount of backtracking.

2.4 Summery of Visual word assignment

Comparative analysis of above discussed methods is given in table 1

Table 1: Comparative Analysis

Method	Time Complexity	Parameter
K-means	O(dM)	single Image in database
Sequential Index	O(d*M/N* B)	N partitions, Single Image
KD tree	O(NlogN) OR O(KN)	If feature dimension is less else time complexity is same as brute force search

3. VISUAL WORD MATCHING

Visual word matching is next step in generating results based on generated visual words. Matching deals with finding best matching visual word for query image with database image visual word. This matching will decide result set for given query image. Following figure depicts process of visual word matching

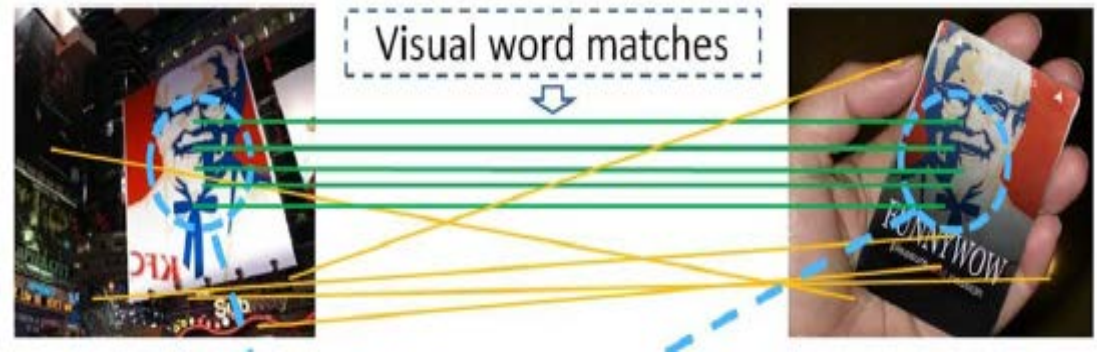

Figure 3: Visual word matching

3.1 Weak Geometric consistency

The key idea of this method is to verify the consistency of the angle and scale parameters for the set of matching descriptors of a given image. This method can be thought of as extension of bag of features in which initial search results are found out by mapping difference between database image descriptor and query image descriptors. Scores between each pair are found by following eq. (2.2.1) [7]

$$s_i := s_i + f(x_i, x_i') \qquad (3)$$

Where s_i =matching score for database image I $f(x_i, x_i')$ =Distance between image i and query image i'

For a given image i, the entity si then represents the histogram of the angle and scale differences, obtained from angle and scale parameters of the interest regions of corresponding descriptors. This is obtained by modifying the update step of above eq.(2.2.1) as follows:

$$s_i(\delta_a, \delta_s) := s_i(\delta_a, \delta_s) + f(x_i, x_i') \qquad (4)$$

Where δ_a and δ_s are the quantized angle and log-scale differences between the interest regions. Weak geometric constraint cannot effectively deal with affine changes.dP-WGC which deals with affine changes and non-rigid deformation.

3.2 Spatial Coding

The spatial relationships among visual words in an image are critical in identifying special duplicate image patches. After SIFT quantization, matching pairs of local features between two images can be obtained Spatial coding encodes the relative positions between each pair of features in an image. Two binary spatial maps, called X-map and Y-map, are generated. The X-map and Y-map describes the relative spatial positions between each feature pair along the horizontal (X – axis) and vertical (Y – axis) directions, respectively [4]. For instance, given an

image I with K features $\{v_i\}$, (i = 1, 2, · · ·, K), its X − map and Y − map are both K × K binary matrix defined as follows,

$$Xmap(i,j) = \begin{cases} 0 & \text{if } x_i \le x_j \\ 1 & \text{if } x_i \ge x_j \end{cases}$$

$$Ymap(i,j) = \begin{cases} 0 & \text{if } y_i \le y_j \\ 1 & \text{if } y_i \ge y_j \end{cases}$$

where (*xi*, *yi*) and (*xj*, *yj*) are the coordinates of feature *vi and vj* , respectively.

Spatial coding is better than RANSAC but this approach is not rotation invariant.

3.4 Summery of Visual word matching

Hamming Embedding and weak geometric constraints were initially used to match visual words. These techniques were invariant to scaling and used SIFT as local descriptors, but WGC cannot effectively deal with affine changes.RANSAC has a full geometric verification method which defines set of valid key point matches. This method is so time consuming and can be applied on few images which makes this technique can be applied on ranked lists obtained from initial BOF matching. Spatial coding measures global relative position consistency which has better performance than RANSAC but is rotation invariant. Geometric coding is further improvement in spatial coding and is rotation invariant. This method is affected by detection error of SIFT orientation hence is less effective than spatial coding in retrieving non-rotated images. Both above methods measure similarity by number and proportion of hard quantization, but this will lead to loss in spatial information. Soft quantization removed drawbacks of hard quantizations.Complex graph model is computationally expensive which leads to slow matching speed.

4 CONCLUSION:

Visual word generation techniques are mainly dependent on dimension of visual feature used for clustering. This clustering can be carried out by tree structures or sequential approach, but it has been seen that sequential approaches are more expensive than tree structures. Tree structures used for finding nearest neighbor are likely to obtain local optimum for given query feature which fails in obtaining in finding global optimum. Global optimum for given query feature can be found out by considering co-occurrence of its neighboring features.

Visual word matching is required in finding precise result from given dataset of images. Matching considered spatial relations of both matching images and this matching can found through different data structures.

REFERENCES:

1. Shi, Miaojing, Ruixin Xu, Dacheng Tao, and Chao Xu. "W-tree indexing for fast visual word generation", In *IEEE Transactions on Image Processing*, Vol. no. 3 Page(s): 1209-1222, March 2013.
2. Lingyang Chu, Shuqiang Jiang,Shuhui Wang,Yanyan Zhang, and Qingming Huang, "Robust Spatial Consistency Graph Model for Partial Duplicate Image Retrieval", In *IEEE Transactions on Multimedia,* VOL. 15, NO. 8 Page(s): 1982-1996, DECEMBER 2013.
3. Miaojing Shi, Ruixin Xu, Dacheng Tao,Chao Xu, "Fast visual word quantization via spatial neighborhood boosting", In *IEEE International Conference on Multimedia(ICME)*,Page(s):1 - 6,11-15 July 2011.
4. Zhou, Wengang, Yijuan Lu, Houqiang Li, Yibing Song, and Qi Tian,"Spatial coding for large scale partial-duplicate web image search.", In *Proceedings of the international conference on Multimedia*, Page(s): 511-520. ACM, 2010.
5. Liu, Hairong, and Shuicheng Yan,"Common visual pattern discovery via spatially coherent correspondences ",In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*,Page(s):1609-1616, 2010.
6. Muja, Marius and Lowe, David G,"Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration.", In *International conference on Computer vision Theory and Applications*, Page(s):331-340,2009.
7. Jegou, Herve, Matthijs Douze, and Cordelia Schmid,"Hamming embedding and weak geometric consistency for large scale image search", In *European conference on Computer vision* ,Page(s):304-317,2008.
8. Bosch, Anna, Andrew Zisserman, and XavierMuoz, "Scene classification using a hybrid generative/discriminative approach,"*IEEE Transactions on Pattern Analysis and Machine Intelligence*, Page(s): 712-727,2008.
9. Silpa-Anan, Chanop, and Richard Hartley, "Optimized KD trees for fast image descriptor matching", In *IEEE Conference on Computer Vision and Pattern Recognition* , Page(s):1-8., 2008.
10. Pavan, Massimiliano, and Marcello Pelillo, "Dominant sets and pairwise clustering." *IEEE Transactions on*

Pattern Analysis and Machine Intelligence, Page(s):167-172, 2007.

11. D. G. Lowe, "Distinctive image features from scale invariant key points", *International journal of computer vision,vol. 60*, Page(s):91-110, Nov. 2004.

12. Swain, Michael J., and Dana H. Ballard, "Color indexing ",*International journal of computer vision 7.1*,Page(s): 11-32, 1991