Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/090152

Image Retrieval based on VA-File and Multi-Resolution BOW

Wang Lipin[1,*] *and Pu Juncheng*[2]

[1] Engineering Technology Research Center of Optoelectronic Technology Appliance, AnHui, China
[2] Istituto Europeo di Design, Huayuan xincun 127, Tongling City, Anhui Province, (244000), China

Received: 20 Apr. 2014, Revised: 21 Jul. 2014, Accepted: 22 Jul. 2014
Published online: 1 Jan. 2015

Abstract: By the virtue of BOF to describe high-dimensional data, in this article, we propose an effective retrieval strategy employing multi-resolution BOF to accelerate the match. The main idea is to improve the overall retrieval efficiency of BOF Descriptive Vector via the construction of BOF Low-resolution Vector and the comparison under low resolution to filter high-resolution candidate vectors. Based on stratified construction, we have improved uniform quantization multi-resolution BOF and proposed a non-linear Non-uniform quantization multi-resolution BOF method, which is combined with VA-file. At last, K-nearest neighbor retrieval algorithm is given. Experiments prove that this method has effectively increased the retrieval efficiency, improved the I/O function when loading mass image datasets and raised the system efficiency.

Keywords: Image retrieval, VA-file, BOW, Quantization

1 Introduction

Vision is an important approach to perceive the world. 80% information of the nature is received via vision. In the area of Visual Media Processing, various visual feature extraction methods are developed to strengthen the discrimination of visual features. Due to the information explosion and emergence of high-dimensional and mass multimedia data emerge brought by the rapid development of multimedia and Internet technology. It has become the primary problem to study efficient visual feature description. Images are the most popular and fundamental carrier of visual information, so the extraction, classification and retrieval of image features become the hot spot in computer and multimedia researches.

Image retrieval techniques have gone through three stages: text-based, content-based and sematic-based. Early image retrieval techniques originated from text retrieval techniques. They describe image features with texts, which are usually manually labelled, including names, numbers, contents and sizes. Then they construct image retrieval databases using these data, and perform retrieval using these key words or classification categories. Essentially text-based image retrieval is the exact or probability matching of descriptive texts. The most tedious work of such retrieval is to label image contents manually, because it cannot be accomplished by computers and is subject to subjectivity of the operator. In addition, a large amount of image information and the visual information related to the subjectivity of human perception cannot be described by texts. If ambiguity occurs in image description, there will be no correct result of retrieval.

Content-based image retrieval techniques mainly construct feature vectors according to image information, such as colors, textures, shapes and spatial relations, and perform similarity retrieval on these feature vectors. Such techniques replaced the awkward manual labeling with automatic feature extraction for image description, and thus improve the efficiency as well as save labor. Content-based retrieval usually contains the detection and extraction of image features, the measurement of feature similarity, related feedback retrieval, evaluation and so on. At present many research institutes at home and abroad have started related research plans and projects and achieved a number of progresses on content-based image retrieval, such as QBIC, PhotoBook and MARS. Sematic-based image retrieval is an ideal retrieval method, which embodies the intelligent development of computers. It uses high-level sematic features to

* Corresponding author e-mail: lipinwangwst@126.com

intelligently describe images. The visual features used in content-based retrieval are low-level ones, and there is huge gap between these features to high-level semantic ones. To map low-level features to high-level semantic features, researches have forwarded three kinds of sematic extraction methods: knowledge-based, human-interacted, and extraneous information related. They form the mainstream of present semantic extraction. However, due to the limitation in the development of Computer Vision, Pattern Recognition and Image Understanding, there are many problems unsolved in semantic-based image researches and there is no satisfying approach for semantic-based retrieval. But it is undeniable that semantic-based image understand will become the focus in the image understanding area.

Simple image description is usually based on global features, such as colors, greyness values and textures. These features are easy to extract and calculate, but not effective under scalar variation and affine variation. Another descriptive method focuses on summarizing the features of local points which form local features of images. Such features are usually generated in the form of sets. The most representative one is Scale Invariant Feature Transformation (SIFT) algorithm proposed by David G. Lowe [1]. It features in a local feature description operator that is invariant to scalar, rotational and even affine changes. SIFT descriptive operator is a 128-dimensional vector, which carried the invariant local information of key points and has strong discrimination. But to describe an images, tens of hundreds key points need to be extracted, and the mass media data will further increase the burden of storage and calculation. Therefore, it is essential for scientific researches to decrease the complexity of the descriptive operators as well as intelligent analysis and efficient utilization of mass visual media data via semantic understanding. According to vision computation theories, the overall consideration on image local feature sets can effectively decrease the complexity of calculation. The ideas of overall description always employ statistic methods to process the information in local feature sets and construct high-dimensional descriptive vectors. Recently, many related researches have been carried out, and the most famous one is SIFT-based Bag of Words (BOW) Model. It has successfully phased ideas in Semantic Understanding into image processing. BOW regards local features of an image as visual words, constructs a coding library via clustering algorithms, and uses the frequencies of visual words in the library as the global feature of the image. Such idea makes it possible to describe an image with a high-dimensional vector rather than hundreds of operators. This has greatly reduced the complexity of the algorithms and served to phase a variety of high-dimensional vectors based ideas into image processing. The proposal of BOW has provided new approaches for image description, processing and semantic understanding. And the advantages of BOW ensure good performance in the area of image classification. However, BOW brings the problem of over high dimension. Experiments have shown that low-dimensional BOW lacks discriminative capacity, and to maintain enough information, proper weighting methods are needed to generate high-dimensional BOW vectors, which is typically 2000∼4000 dimensional. This, however, brings a new problem: how to make sure the efficiency to search among mass high-dimensional BOW data. High-dimensional BOW retrieval is in essence a kind of high-dimensional data retrieval, which requires effective indexing for acceleration and proper similarity measurement.

In the area of multimedia research, images and videos play an important role. Before the proposal of SIFT by David G. Lower, global features such as colors, shapes, textures and greyness values were widely used. They were regarded indispensable in the area of image and video processing for a long time. However, they are variant to the changes in images and lights, which give way to local features. As a representative of local features, SIFT firstly detects features in the scalar space and confirms the locations and scales of key points. Then, it sets the primary direction of a key point as its direction feature to realize the independence of the operator to scales and direction. Each SIFT operator is a 128-dimensional vector. SIFT has brought researches on image description to a new level. Yanke and Sukthankar improved standard SIFT with histogram generated via smooth weighting. They used Principal Component Analysis (PCA) to quantize gradient blocks and construct low-dimensional PCA-SIFT [2]. Compared to standard SIFT, PCA-SIFT is more compact, unique and robust to transformation. The development of local features brings new problems the same time as it brings new approaches, and the rapid increase of data amount has become the bottleneck of such systems. Researchers have forwarded many strategies to accelerate the feature matching, for example in 1999 Gionis A et al. came up with Local Sensitive Hashing (LSH) [3]. LSH is mainly based on Hashing function. It projects high-dimensional data to a straight line using Random Projection and constructs Hashing function via the segmentation of this line. However, the application of LSH is always limited to main memory index and infeasible to mass data sets in external storage index. H. Lejsek et al. proposed NV-tree [4] to high-dimensional mass vectors, which aim to construct external storage index for mass data sets. It constructs effective disk-based data structure which ensures satisfying nearest neighbor query with only one disk operation and high efficiency in processing mass high-dimensional data. In the area of image detection and recognition, more and more researches have studied local feature sets as a whole. In 2003, J. Sivic et al. were the first to propose the concept of Bag of Features [5], which phased text semantic understanding into image processing. Afterwards, many further researches emerged on BOF construction factors, such as the extraction of BOF image blocks [6], the description of feature [7], the

construction of BOF coding libraries [8,9] and the design of classifiers [10]. These researches aimed at improve the discrimination and efficiency of BOF. On the other hand, many algorithms and ideas are derived from BOF, such as: 1) Binary BOF, which is a simple method to compress BOF vectors. This method leaves out the frequency of visual words and its components simply show the existence of the corresponding visual words. Binary BOF performs commonly on small-scales, but it provides satisfactory descriptive and discriminative abilities on high-dimensional BOF vectors (more than 10,000 dimensions), meanwhile it keeps high calculation capacity. 2) Mini-BOF [11]. It firstly generates many small descriptive operators from the original BOF, which are called Mini-BOF vectors. Each Mini-BOF vector provides partial information of the original BOF and has its own indexing. Then the information returned by Mini-BOF vectors is merged via a distance expectation based fusion strategy. Compared to standard, this method is not only highly efficient but also costs only hundreds bits to describe an image. There are also many researches concentrated on image retrieval and classification based BOF, such as 1) Inverted File technique, which originates in text retrieval but has good performance on BOF-based image retrieval. The construction of inverted index file will effectively speed up the matching between target images and candidate images in the database. 2) PLSA [12], which is an updated version of Potential Semantic Analysis. Through statistic ideas, it constructs s potential semantic layer between texts and words and fits this model via Expectation Maximization (EM) to get rid of synonyms and polysemants. P. Quelhas et al. [13] introduced Potential Semantic Model into BOF and accomplished BOF-based Shot Detection Algorithm. On the basis of popular high-dimensional data indexing methods, we proposed stratified methods to construct multi-solution BOF. After multi-solution reconstruction on the original BOF, we filter out some high-resolution candidates under low-resolution and thus improve the overall function of BOF-based image retrieval. We also apply VA-file into the multi-resolution BOF structure to further reduce the I/O consumption of BOF structure during mass data processing, which improves the performance of the whole system.

2 Related works

2.1 A. Descriptive features of images

Image features consist of global features and local features. Global features describe colors, textures, shapes, greyness coexistence matrixes and so on. Colors are the pixel features of images or image regions, and all pixels in these regions contribute to the construction of features. Colors are invariant to rotation and translation, and insensitive to the changes in directions. Color features are high-dimensional and not suitable for retrieval in large databases. Common color extraction methods are Color Histogram, Color Set, Color Distance and Color Clustering Vector. Texture features describe the surface of images via the statistics of multiple pixels. They cannot reflect the attributes of objects in images or obtain high-layer image contents. Textures are invariant to rotation and robust to noise, but the extraction of textures is greatly subject to the change of image resolution. Common methods of texture extraction are Feature Analysis on Greyness Coexistence Matrix, Geometric Analysis and Model Method. Shapes are features describing the shapes of the objects in an image. They can effectively retrieve via the key objects in images. However, such features lack complete mathematical models and become ineffective when the shapes of targets changes. Common methods to describe shapes are: Edge Feature Method, Fourier Shape Description, Geometric Parameter Method and Moment Invariance. The above features all provide global information and cannot reflect the object or local information.

Local features are those that discriminate a region from its neighboring regions defined by certain saliency criteria. They are usually related to changes in one or more image properties, and contain multiple forms such as points, lines and blocks. The typical procedure for local feature construction is: 1) search for a series of key points; 2) define a key region surrounding each key point; 3) extract and normalize the contents of key regions; 4) calculate local descriptive operators according to the normalized regions. Generally speaking, local features should bear sufficient descriptive and discriminative capacity to depict the image. They are always invariant to changes such as translation, rotation and brightness, which is an important strength against global features. Local features consist of two parts: key point detector and descriptive operator. These two techniques play important role in image-based 3D reconstruction, image database retrieval as well as object and location detection. Key point detector serves to select monitoring regions and key points in images and descriptive operator specify the surrounding regions of key points. The output of descriptive operator is a vector, which is invariant to common image changes and can be matched to other objects in the database according to certain similarity measurement and criteria.

The key point detector and descriptive operator are critical. In the area of computer vision, the researches on region detectors invariant to certain changes are well developed, the popular among which are detectors based on scale invariance and detectors based on affine invariance. Scale invariant detectors yield results that are invariant to scalar changes. Present methods are to search for local extremes in 3D spaces presented by (x, y, scale). The idea was proposed by Crowley and Parker in the 1980s. In scale invariant detectors the pyramid structures of images are often calculated via filters such as Gaussian Difference. When a pixel is a local 3D extreme and larger

than a given threshold, it is selected as a key point. The variation between different scale invariant detectors is often reflected by different expressions for scalar space description. The representatives of scale invariant detectors are LoG and DoG Detector.

Laplacian-of-Gaussian (LoG) Detector [14] was proposed by Lindeberr as an algorithm to detect key regions via the detection of 3D maximum in LoG scalar spaces. The scalar space of LoG is constructed by the consecutive smoothing of high-resolution image under different scales of Gaussian Kernel. Such LoG operation is cyclosymmetric and the detection result is of block structure. LoG Detector realizes scale invariance mainly through automatic scale selection, which is also employed by Bretzner and Lindebery on image tracking technology.

Difference-of-Gaussian (DoG) Detector [1] is an effective algorithm proposed by Lowe in 1999. It detects local 3D extremes in the scalar pyramid constructed by Gaussian Difference Filter. The input image is consecutively smoothed and sampled via Gaussian Kernel and presented by the Gaussian Difference obtained by the subtraction of two consecutively smooth images. So all DoG layers are constructed by the combination of smoothing and subtractive sampling. The local 3D extremes in the pyramid structure determine the scales and locations of key points. DoG operation is an approximation of LoG, but it can significantly speed up the calculation. With DoG method several imaged can be processed per second. The common disadvantages of DoG and LoG is the potential detection of local extremes on the edges and line boundaries. The signal changes of points in these regions are unidirectional, and their locations are too sensitive to the surrounding noise and small changes to be stable. Accordingly, the detection results of DoG and LoG are unstable.

Affine invariant detectors are Harris Detector, Harris-Laplace Regional Detector, Hessian-Laplace Regional Detector, Harris-Affine Regional Detector and so on.

Harris Detector is invariant to image translation and rotation. It detects the values of greyness via the minor displacement to random directions. The detection regions given by Harris Detector are 41×41 pixel blocks centered at the key points. This detector is simple with homogeneous and proper feature, but not adaptive to scalar changes.

Harris-Laplace Regional Detector [15] is invariant to rotation and scalar changes. The algorithm consists of two steps: multi-scale point detection and iteration of scales and locations. First of all, it selects the scale of key points according to the scalar extremes of LoG, and constructs spatial pyramid structures consisting of images of different resolutions under each selected scales via Harris Function. On each layer of the pyramid structure, decide whether a pixel point belongs to the candidate set according to whether it is the extreme value against 8 surrounding points. Rule out some candidate points via Non-maximum Suppression in the scalar space. Then, a candidate point is set as a key point if it is the Laplace local maximum along the scalar direction, and its scale is set as the feature scale. The structure given by Harris-Laplace detection is of corner structure.

Hessian-Laplace Regional Detector [16] is invariant to rotation and scalar changes. Key points are located as the local maximum via Hessian Determinant Operator in the space and via Gaussian-Laplace Operator in the scalar space. Such Detector is similar to DoG detector, but yields more precise results in the scalar space. The precision of locations will influence the performance of descriptive operators, thus Hessian-Laplace performs better in regional detection.

Harris-Affine Regional Detector is invariant to affine changes. The locations and scales of key points are obtained by Harris-Laplace Detector estimation. Affine neighboring regions are obtained via second-order matrix based Affine Adaptor.

2.2 SIFT local feature descriptor

The BOF technique studied in this article is usually constructed on the basis of SIFT descriptor, it is necessary to introduce the basic concepts and construction process of SIFT algorithm. On this basis, we will analyze the strengths and short comings of SIFT, which casts light to the selection of it for high-level image description method.

SIFT is obtained by image block standardization suggested by Lowe. It is a 3D histogram of gradient locations and directions. Locations are quantized as location grids and the angles of gradients as 8 directions. The final descriptor is a 128-dimensional vector.

Scale Invariant Feature Transform (SIFT), proposed by David G. Lowe in 1999 and improved in 2004. It is a local feature description method based on invariant feature detection technique and invariant to scaling, rotation, brightness changes and even affine changes in the scalar space. The following is the detailed steps of SIFT algorithm:

1. Construction of scalar space and detection of extreme point

   The primary step of key point detection is to determine the locations and scales of key points. Scale is the description of the same object from different views. The first step of detection is scalar space construction, during which the construction of Gaussian Pyramids is needed. There are o groups of Gaussian Pyramids and s layers in each group. Plenty experiments proved that $s = 6$ is the optimal selection, and values larger than 6 will cause instability to the selected key points. After the selection of the numbers of groups and layers, greyness processing is performed on the original image and the image on the first layer of the first group is generated via Gaussian Smoothing. The image on a higher layer is obtained from Gaussian Smoothing of the image on the previous layer. The image on the first

layer of each group (except for the first group) is obtained from 1/2 sampling of the image on the last layer of the previous group. In Gaussian Pyramids constructed in this way, the scales of images are increasing by k as the ascending of layers. Each layer of DoG is from the subtraction of previous two layers. DoG Pyramid is the scalar space for key point detection.

2. Determination of key point locations
   To determine the locations of key points, the value of every pixel in every layer of image should be compared with its surrounding 26 pixels. If the value is a local extreme, then it is included into the candidate set for key points. After extreme detection, we get the candidate set. Then, to get robust key points, two steps are taken. The first step is to rule out low-contrast points, i.e. the pixel values of key points should differ greatly from its neighboring points. The second step is to rule out edge points, because DoG Operator yields strong edge response, which results in unstable edge points. The deletion of edge points will increase the matching stability and resistance to noise.
3. Determination of key point directions
   The construction of direction parameters of key points ensures rotation invariance. In SIFT algorithm, the value and direction of the gradient of a key point are its direction parameters. They are defined as follows:

$$m(x,y)=\sqrt{(L(x+1,y)-L(x-1,y))^2+(L(x,y+1)-L(x,y-1))^2} \quad (1)$$

$$\theta(x,y)=\tan^{-1}((L(x,y+1)-L(x,y-1))/(L(x+1,y)-L(x-1,y))) \quad (2)$$

   The algorithm samples in the window centered at the key point with the radius of r and calculate the direction of gradient in the neighboring area via histogram. The range of gradient histogram is 0∼360 degree, in which each column spans 10 degrees and there are 36 columns. The peak values of the histogram are the principal direction of the neighboring gradient, i.e. the direction of this key point. If a peak value that is 80% of the principal peak exits, the corresponding direction is the auxiliary direction of this key point. A key point may be assigned to several directions (one principal and more than one auxiliary), which can increase the robustness of matching.
4. Generation of descriptor
   First of all, rotate the axis to the direction of key point to ensure rotation invariance. Make the gradient histograms in 8 directions in each 4×4 pixel block and achieve the sum of each gradient direction to form a seed point. For each key point, 2×2 i.e. 4 seed points are picks, each contains the information of 8 directions, thus, a 32-dimensional descriptive vector. In actual calculation, Lowe suggested to describe each key point using 4×4 seed points to increase the stability of matching. Thus, 128 data points, i.e. a 128-dimensional SIFT vector, are generated for one key point. The SIFT feature of an image is formed through the above four steps and it contains information such as key point locations, directions, scales and descriptors. The most frequently used information in this article is the descriptor.

The SIFT vector has the following advantages: a) SIFT is a local feature, which is invariant to rotation, scaling and change of light and stable in a certain extent of changes in visual angle, affine transformation and noise. b) It contains abundant information and has great distinctiveness, so it is applicable to fast and accurate matching among mass feature data. c) It has a large quantity that even a few objects can generate a number of SIFT features. d) It is high-speed, which makes extraction convenient and fast and satisfies the requirement of real-time. e) It is extensible and easy to combine with other feature vectors.

The above advantages secure the importance of SIFT in the area of image and video processing. However, its high-resolution and stability increase the complexity. To describe an image, hundreds and thousands of feature vectors are needed. Such complicated description causes the problems of high complexity and storage demand during mass image database retrieval and video frame detection. In addition, the variation of the number of vectors in different images brings difficulty in matching algorithms. Therefore, researchers have started to regard these features as a whole and use semantic method to describe images via some statistic ideas.

2.3 Vector Approximation

With the rapid increase of multimedia data, it becomes an urging demand to obtain more profits in multimedia data storage, browsing, retrieval and search. Many applications such as digital museums, online entertainment and shopping as well as multimedia data retrieval have concentrated active searching more on the indexing and retrieval of image database. Meanwhile, image retrieval is an important part of multimedia information management. Retrievable images should be indexed via the content, which is usually realized by labelled key words, automatically detected Visual Cues or visual words. Sematic based image retrieval via labelled key words is a simple method. But for large database, the performance of this method loses rapidly with the expanding of data.

The aim of indexing structure in multimedia information retrieval is to receive the query results as soon as possible. With the increase of dimensions, Dimensional Disaster is unavoidable in tradition indexing structures. Weber et al. forwarded Vector Approximation File (VA-file) [17] on the basis of Cost Model in high-dimensional spaces. This method employs not tree indexing but rather a sequential access searching algorithm. But it is not the original vectors that are visited, but rather the approximate vectors after compression. During retrieval, the minimum and

maximum limit of the distance between the query vector and the original feature vector is calculated via the approximate vector, and according to the bounds most data are filtered out. The storage of approximate vectors are far less demanding than the original vectors, so VA-file will effectively reduce the I/O duration during the searching process and greatly improve the searching efficiency.

VA indexing maintains two data files: sequence file for approximate vectors and sequence file for original vectors. Assume the dimension of the vector space is d, to perform approximation to vectors, the space of each dimension j is allocated with certain approximate digits b_j ($\Sigma b_j = b$) and divide the axis equally into 2^{bj} intervals and the vector space into 2^b hypercubes. The approximate vectors of all vectors in the same hypercube are the same. As shown in Figure 1, the space of each dimension in the 2D space is assigned 2-bit numbers, and then the approximate vector of $\boldsymbol{p}_i$ is [11]. Thus, vectors that were originally presented by two floats (8 bytes) are now presented only via 4 bits, and thus the vectors are compressed. The sequential file of all approximate vectors is a VA-file. Suppose the approximate vector of $\boldsymbol{p}_i$ is $\boldsymbol{a}_i$, then the maximum limit of Vector $\boldsymbol{p}$ and Vector $\boldsymbol{q}$ is u_i and the minimum limit is l_i.

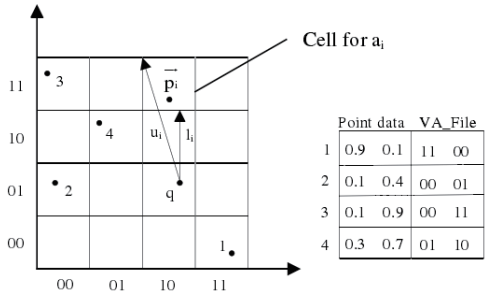


Fig. 1: Vector approximation file diagram in 2-dimension.

Label the 2^{bj} breakpoints of j-axis as $f_{l,j}, l = 0, 1, \ldots, 2^{bj} - 1$, and the lth interval is presented as $[f_{l,j}, f_{l+1,j}]$. Then u_i and l_i is calculated as:

$$l_i = (\sum_{j=1}^{d} (l_{i,j})^p)^{1/p} \tag{3}$$

$$u_i = (\sum_{j=1}^{d} (u_{i,j})^p)^{1/p} \tag{4}$$

Therein:

$$l_{i,j} = \begin{cases} q_j - f_{l+1,j} & q_j > f_{l+1,j} \\ 0 & q_j \in [f_{i,j}, f_{l+1,j}] \\ f_{l+1,j} - q_j & q_j < f_{l,j} \end{cases} \tag{5}$$

$$u_{i,j} = \begin{cases} q_j - f_{l+1,j} & q_j > f_{l+1,j} \\ \max(q_j - f_{l+1,j}, f_{l+1,j} - q_j) & q_j \in [f_{i,j}, f_{l+1,j}] \\ f_{l+1,j} - q_j & q_j < f_{l,j} \end{cases} \tag{6}$$

There are two steps in k-nearest Neighbor based on VA-file. The first step is to calculate the maximum limit u_i and minimum limit l_i between Approximate Vector $\boldsymbol{a}_i$ and Query Vector $\boldsymbol{q}$. When l_i is larger than the present k^{th} smallest maximum limit, this vector will be deleted, for there are at least k vectors that are more suitable for the requirement. Those remaining vectors will enter the next step. In the second step, the original vectors are visited. Visit the candidate vectors remained from the first step sequentially according to the increase of corresponding minimum limits and calculate their distances from the query vector. Not all candidate vectors are visited. If the distance minimum limit of a vector is higher than the k^{th} nearest neighbor, then, without reaching an end, the present k neighbors are the results. Under high dimensional conditions, VA-based indexing is the only method better than sequential searching for high precision nearest neighbor query. As for other methods, they cannot ensure that they are better than sequential searching on any format of dataset in high-dimensional situations. However, VA-based Nearest Neighbor Searching employs exhaustive algorithm and needs to visit all data, so the CPU operation complexity and I/O complexity become the limiting factor when the dataset is large.

2.4 BOW Model

In text classification area, there is a model called Bag of Words (BOW) Model. It is a simplified model of natural language processing and retrieval. In this model, texts (paragraphs and files) are simplified as the set of non-sequential words without grammars and sequences. The appearance of words is independent. BOW plays a critical role in text classification. It proposes an idea to describe texts based on statistics. In this idea, words in the training texts are trained and a unique vocabulary is generated. Then, the target text is processed via the vocabulary and the frequency of each word in this text is summarized to construct the descriptive vector of the text. Afterwards, text classification or retrieval is performed based on the vectors. BOW is efficient as well as simple. In 2003, Sivic et al. from Oxford University were the first to forward BOF [18]. The idea is mostly similar to semantic classification based on texts. Local features are quantized as labelled visual words, and visual words from the training set are clustered into a codebook. A local feature is mapped to a histogram reflecting the frequencies of key words, i.e. a BOF vector. This generation of BOF has provided new an approach in the area of image semantic understanding. The basic idea of BOF is to regard an image as a set of independent image blocks, and from each image block a descriptive vector (descriptor) is constructed. Cluster the descriptors of the training set to form a codebook containing visual words. Perform Weighting Statistics analysis on the descriptor of target image according to the codebook to generate a feature histogram vector whose dimensional number is

1000∼4000, and this high-dimensional vector presents the image. Afterwards, generate a classifier according to the high-dimensional vectors of images from the training set and perform image classification. The main steps of BOF construction algorithm is as follows:

1) Detect image blocks and generate descriptors. Common methods to detect image blocks are dense sampling algorithms, random sampling algorithm and key point based sampling algorithms, such as Harris-Laplace, Laplacian and Gaussian. Common methods for descriptor generation are SIFT, PCA-SIFT and GLOH.
2) Allocate descriptors of image blocks into different clusters via a clustering algorithm. The centers of clusters are called visual words, and the set of visual words is a codebook. Common clustering algorithms are K-mean and so on.
3) Allocate the descriptors of target image into the clusters in the codebook using a weighting strategy. Construct a key point frequency vector according to the number of descriptors in each cluster. According to the specific application, the original vector is further processed, such as weighting and normalization.

3 Multi-resolution BOW

To achieve better classification and discrimination via BOF, the number of dimensions should reach a certain value, which is as small as 2000∼4000 and as large as tens of thousands. The description of mass data via such highly dimensional vectors is time-consuming. BOF-based image retrieval is high-dimensional data retrieval at the same time. Theoretically, methods for high-dimensional data processing, such as dimension reduction, are applicable to BOF. Inspired by MRSA Stratification, this article phases this method into BOF processing and adapts it to this new application area.

3.1 Multi-resolution BOF constructed by MRSA

Researchers has found an effective multi-resolution searching algorithm for fast exhaustive searching, namely MRSA. To delete improper candidates, this algorithm has developed a Sum Pyramid Structure for image features (or histograms). For Histogram $\boldsymbol{X}$, the Summation Pyramid is defined as a sequence of histograms $\{\boldsymbol{X}^0, \ldots, \boldsymbol{X}^l, \ldots, \boldsymbol{X}^L\}$. $\boldsymbol{X}^L = \boldsymbol{X}$ and there are 2^l components in $\boldsymbol{X}^l$. The number of components is cut by half in low-resolution situations. A pyramid structure is generated via the accumulation of neighboring components in high layers, i.e. $\boldsymbol{X}^l(i)$ is generated via:

$$X^l(i) = X^{l+1}(2i-1) + X^{l+1}(2i) \quad 1 \leqslant i \leqslant 2^l \tag{7}$$

For a given query Histogram $\boldsymbol{Q}$, the following formula can be proved:

$$d(X,Q) \equiv d^L(X,Q) \geqslant \ldots \geqslant d^l(X,Q) \geqslant \ldots \geqslant d^0(X,Q) \tag{8}$$

$d^l(\boldsymbol{X}, \boldsymbol{Q})$ means $d(\boldsymbol{X}^l, \boldsymbol{Q}^l)$. It should be emphasized that the theory is built on the basis of the selection of proper similarity measurement function d. Only similarity measurement functions meeting Formula (8) are suitable for pyramid structure construction, such as Hamilton Distance and χ^2 Distance. In this article we mainly use Hamilton Distance. MRSA pre-calculates the Summation Pyramid of each candidate. When searching for the best match for a target image, the Summation Pyramid of $\boldsymbol{Q}$ is constructed at the first place. Then for each layer of the pyramid, calculate the distance $d^L(\boldsymbol{X}_n, \boldsymbol{Q})$ between Candidate L and the target image and compare it with the minimum distance $d_{\min}$. If $d^L(\boldsymbol{X}_n, \boldsymbol{Q})$ is larger than $d_{\min}$, $\boldsymbol{X}_n$ is excluded and spared of the distance calculations of higher layers. The time consumption of higher layer calculation is larger than lower layer, so MRSA can effectively reduce the searching duration. Though the above introduction of MRSA, it is shown that BOF is also a statistic histogram structure. It can be constructed and compared through Hamilton Distance, so MRSA can be easily applied to BOF. MRSA requires the number of components in a histogram to be an even number or 2 to the N^{th} power, but the components of some visual features are not 2 to the N^{th} power. Although Zero-Padding can solve this problem, the size of histograms increases obviously due to this method.

3.2 Multi-resolution BOF constructed via Non-uniform Quantization method

Rather than through the summation of two neighboring components to obtain low-resolution components, we can construct low-resolution histograms through the generation of low-resolution components via the combination of several neighboring components. In this area of information retrieval, if Word t appears frequently in a text, it will be allocated a high weighting factor. This can be applied to the area of image retrieval to regard Word t as an image feature. During the construction of multi-resolution BOF, we hope to maintain the components with high weighting factors when generating low resolution. So a non-linear function deciding the grouping of neighboring components is needed during the non-uniform quantization. This function varies according to different image datasets. First of all, include all BOF components in a histogram $\boldsymbol{X}_{acc}$. Suppose there are Z key frames in a database, Accumulative BOF Histogram is defined as:

$$X_{acc}(i) = \frac{1}{Z}\sum_{j=1}^{Z} X_j(i), \quad i = 1, \ldots, b \tag{9}$$

$\boldsymbol{X}^j$ is the histogram of the j^{th} frame. The weighting factors $\boldsymbol{W}^{\text{TF}}$ and $\boldsymbol{W}^{\text{TF-IDF}}$ are defined as:

$$W^{\text{TF}} = \frac{X_{acc}(i)}{\sum_{i=1}^{b} X_{acc}(i)} \quad i = 1,\ldots,b \tag{10}$$

$$W^{\text{TF-IDF}} = W^{\text{TF}} \times (\log(\frac{Z}{z_i}) + 1) \quad i = 1,\ldots,b \tag{11}$$

z_i is the number of non-zero key frames in the i^{th} component and Z is the number of all key frames. $\boldsymbol{W} = \{w_i | i = 1,\ldots,b\}$ is the set containing the weighting factors of all components. The non-linear function can reconstruct the weighting histogram $\boldsymbol{W}$:

$$W(i) = \sum_{j}^{i} w(j), \quad i = 1,\ldots,b \tag{12}$$

New components are generated from present vectors via the following formula:

$$i_{new} = round(m \times W(i)/W(b)), \quad 1 \leqslant i \leqslant b \tag{13}$$

In this formula, m is the number of new low-resolution components. The formula provides the non-linear function needed to construction multi-resolution BOF via non-uniform quantization methods, and it also determines the neighboring components to be processed for the components in the low-resolution histogram.

4 Image retrieval algorithm based on VA-file and Multi-resolution BOF

4.1 Multi-resolution BOF using VA-file

The multi-resolution structure of BOF is a pyramid structure, which can be retrieved via MRSA. But on this basis, we have made some improvement in this article. Multi-resolution processing can effectively reduce the time consumption, but it increases the redundant storage. Multi-resolution BOF consumes too much storage space and I/O function during mass data processing and thus influences the performance of the whole system. Thus we include VA-file to further improve Multi-resolution BOF and make better performance.

Among the histograms under various resolution $\{\boldsymbol{X}^0, \ldots, \boldsymbol{X}^l, \ldots, \boldsymbol{X}^L\}$, each histogram component $\boldsymbol{X}_i^h$ can be related to an approximate vector ah via VA-file. Thus a VA sequence $\{\boldsymbol{a}^0, \ldots, \boldsymbol{a}^h, \ldots, \boldsymbol{a}^H\}$ is constructed. Calculate the minimum limit of ah under different resolutions, l_i^h. Then $d(\boldsymbol{p}_i^h - \boldsymbol{Q}^h) \geqslant l_i^h$. During K-NN searching, compare the minimum limit under low resolution l_i^h with the minimum distance $d_{\min}[M-1]$, if $l_i^h \geqslant d_{\min}[M-1]$, then:

$$d(p,Q) = d(p_i^H, Q^H) \geqslant d(p_i^h, Q^h) \geqslant l_i^h \geqslant d_{\min}[M-1] \tag{14}$$

4.2 Image retrieval algorithm based on multi-resolution BO

We have introduced the image retrieval algorithm based on multi-resolution BOF with VA-file. In this algorithm, array *knn* contains the distances between k approximate labels i of query image $\boldsymbol{Q}$ and the query point. Array *knn_u* stores the present k nearest labels i and their maximum distances from the query image $\boldsymbol{Q}$. Array *knn_l* stores the present k nearest labels i and their minimum distances from the query image $\boldsymbol{Q}$. The above array is sorted ascending. We divide the algorithm into calculation stage and filtering stage, the detailed process is as follows:

1. Vector calculation stage:
   1) Initialize array *knn_u*, set the distance as MAXREAL.
   2) Pop-up the top elements l_i and i in *heap_l*,
   If $l_i^h \geqslant$*knn_u*[k]*dist*, goto step 3).
   else use $||p_i - q||$ and i to replace the element in *knn*[k].
   Sort *knn*[k].
   Continue step 2).
   3) Algorithm end, the results are kept in *knn* array.
2. Vector filtering stage:
   1) Initialize array *knn_u*, set the distance as MAXREAL.
   Initialize *heap_l*, and set $i = 1, h = 1$.
   2) If $i > N$, goto step 2).
   Repeat each candidate vector.
   Calculate maximum limit distance u_i^h and minimum limit distance l_i^h between the approximate value $\boldsymbol{a}_i^h$ and query vector $\boldsymbol{q}$.
   if $l_i^h >$*knn_u*[k]*dist*, eliminate the current candidate vector.
   $i = i + 1$, set $h = 1$.
   else $h = h + 1$, goto step 2).
   3) Calculate maximum limit distance u_i^H and minimum limit distance l_i^H between the approximate value $\boldsymbol{a}_i^H$ and query vector $\boldsymbol{q}$.
   If $l_i^H >$*knn_u*[k]*dist*, eliminate the current candidate vector.
   else use u_i^H and I to replace element in *knn_u*[k].
   Sort *knn_u*[k].
   Insert l_i^H and $i$into *heap_l*.
   Goto step 2) and $i = i$+1, set h =1.

5 Ry experiemnts and analysis

In this article, we have designed many experiments to evaluate the performance of multi-resolution structure in image and video retrieval. The data used are from UQLIPS system of University of Queensland, as shown in Figure 1. This data set contains 10,000 different video segments, including TV advertisements, movie clips, new clips and documentaries. In the experiments, we use key frames to identify each video segment. A key frame is

usually a shot from the video. We employ a color-based method to automatically extract key frames and combine all key frames together to form key frames data set.

Fig. 2: Key frames extracted for video segments.

The experiments are carried out under the environment of Intel Core 2 2.2G Hz and 4G RAM and averages of several calculations are obtained. DoG Detector and SIFT Descriptor are unanimously used for local feature extraction, and K-means algorithm is used to generate public codebook. Weighting methods TF and TF-IDF are used to generate BOF vectors. Without specification, all experiments in this article are carried out on the mentioned video data set, and 1000 key frames are randomly picked as query images.

5.1 Evaluation of the accuracy of Multi-Resolution BOF

In the experiment, the accuracies of two multi-resolution structures generated via uniform quantization and non-uniform quantization. We randomly select K key frames from the key frame set (K is 10, 20, 30, 40 and 50 respectively). Evaluations are made through Correctly Retrieved Ration (CRR). When CRR is 1.0, all target images are correctly retrieved. To the highest layer of resolution, i.e. the original BOF vector layer, no quantization method is used. Thus, with the same similarity measure method, the CRR value is the same. The mechanism Frame Number is employed for corrective evaluation. In a video segment, each frame including key frames has a unique Frame Number. If the distance between Frame Numbers of target frame and the resulting frame is smaller than a certain threshold, then the target frame is correctly retrieved. In the experiment, the threshold is set to 20. The similarity measurement function in the experiment is Hamilton Distance. If the correct result appears in the first K images, the retrieval is regarded as correct and the algorithm is ended. These experiments mainly evaluate the impact of multi-solution structures on retrieval results, so we do not include VA-file to further process BOF. Multi-resolution employs two-layer structures, i.e. the original BOF and the combinations of different low-resolutions. The original BOF is 2048 dimensional, and low-resolutions are 256, 512 and 1024. Figure 3 shows the result.

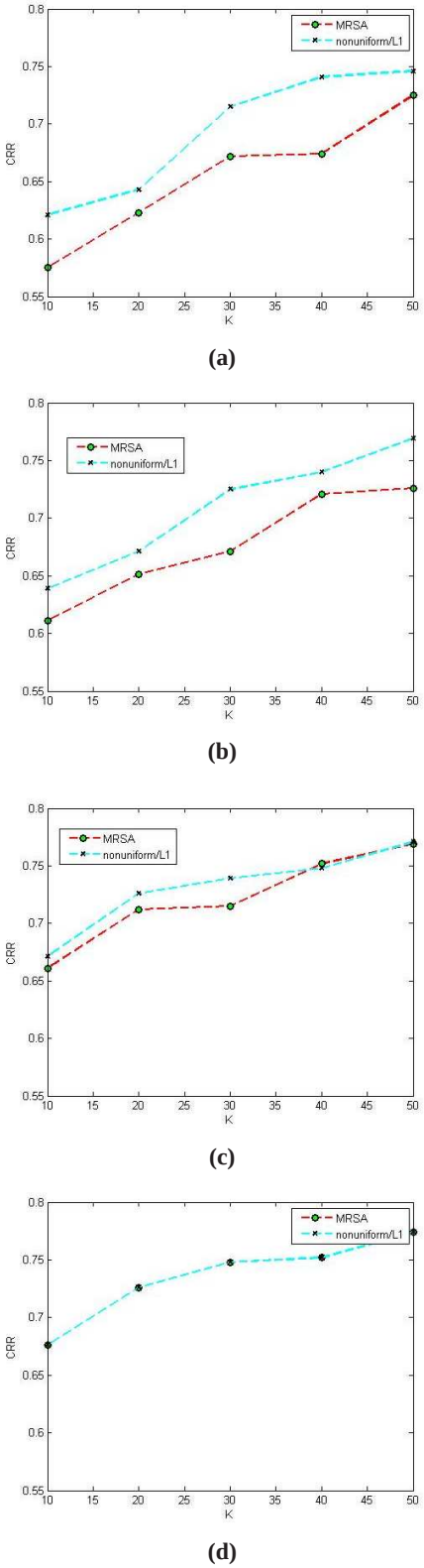


Fig. 3: The impact of BOF structures with different resolutions on retrieval accuracies, the dimension of different low resolutions are (a) 256 (b) 512 (c) 1024 (d) 2048.

The abscissa in Figure 3 is the value of optimal matching number K and the ordinate is CRR. It is shown in the figure that the multi-resolution BOF structure constructed by non-uniform quantization is more accurate than the uniform quantization method MRSA on all low-resolution layers. When the dimension number of the original BOF is 2048, the variation in low resolutions has an impact on the results. Figure 3(d) is the result given by original BOF, i.e. the accuracy curve without multi-resolution structure. It is shown that the accuracy is lowered when multi-resolution structure is included, but such loss can be reduced to a minimum when a proper low resolution is selected. When applying multi-resolution BOF structure, the hierarchical resolution structure should be tested according to different datasets and numbers of target images to reach the optimal effect.

5.2 Comparison on candidate filtering efficiency

To show the improvement of multi-resolution BOF on candidate filtering, we calculate the visiting rates on multiple resolutions to evaluate the performance of uniform and non-uniform quantization on a candidate set. In this experiment, the codebooks are of size 1024 and the original BOF histogram is generated via TF weighting. Then uniform and non-uniform quantization is used to generate different resolution structures. The dimension of the original BOF is 1024, so the highest layer used in the experiment is 10. The experiment is carried out under two conditions: with and without VA-file. Figure 4 and Figure 5 show the results of candidate filtering by multi-resolution BOF with and without VA-file.

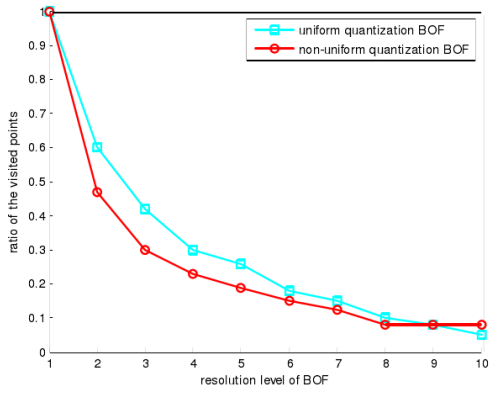


Fig. 4: Filtering rate of multi-resolution BOF with VA-file.

The abscissa shows the number of resolution layers, the ordinate show the visiting rates on candidate images. It is shown that on the highest layer, i.e. the layer with the abscissa value of 1, the visiting rate is 1, which means all candidate images are visited. This is normal because no multi-resolution processing is carried out on this layer.

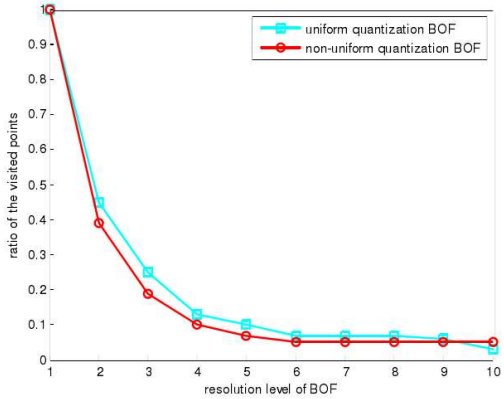


Fig. 5: Filtering rate of multi-resolution BOF without VA-file.

Generally speaking, the filtering efficiency of non-uniform quantization is better than uniform. This is mainly because the former maintains more histogram information. Even without VA-file for further processing, many candidates are filtered due to the properties of multi-resolution BOF. It is shown in the experiment that the more the resolution layers are, the better filtering can be reached. However, the application of multi-resolution structures increases the complexity of calculation and storage and in a way influences the accuracy. Thus the number of resolution layers and low-resolution dimensions should be determined according to different datasets to achieve the optimal result. It is also shown that the inclusion of VA-file can further increase the filtering rate, and, according to the properties of VA-file, higher approximated bit number will further improve the filtering.

6 Conclusions

In this article, we have introduced a simple but new method to process high-dimensional BOF vector. The comparison of multi-resolution BOF structures under low resolutions effectively filters some candidates and reduces the time consumption of this algorithm. We have included VA-file to further process multi-resolution BOF to improve the filtering rate and the I/O function when dealing with mass data. First of all, we theoretically proved the effectiveness of the multi-resolution structure, and then experiments further proved that K-NN based on multi-resolution BOF is better in searching speed than tradition BOF algorithm.

Acknowledgement

The work was supproted by the fund of Collaborative Innovation Center of Tongling University.

References

[1] D. Lowe. Distinctive image features from scale invariant keypoints. International Journal on Computer Vision, 2004, 60(2):91-110.

[2] Y. Ke, R. Sukthankar. PCA-SIFT: A more distinctive representation for local image descriptors. Computer Vision and Pattern Recognition, 2004, 506-513.

[3] A. Gionis, P. Indyk, R. Motwani. Similarity search in high dimensions via hashing. Very Large Data Base. 1999, 518-529.

[4] H. Lejsek, F. H. Asmundsson, B. P. Jonsson, L. Amsaleg. NV-tree: An efficient disk-based index for approximate search in very large high-dimensional collections. IEEE Trans Pattern Analysis and Machine Intelligence. 2009, 31(5):869-863.

[5] C. Dance, J. Willamo wski, L. Fan, C. Bray, G. Csurka. Visual categorization with bags of keypoints. ECCV Workshop on Statistical Learning in Computer Vision. 2005.

[6] E. Nowak et al. Sampling strategies for bag-of-features image classification. Lecture Notes in Computer Science, 2006, 490-503.

[7] K. Mikolajczyk, C. Schmid. A performance evaluation of local descriptors . Pattern Analysis and Machine Intelligence, 2005, 27(10):1615-1630.

[8] J. Farquhar, S. Szedmak, H. Meng. Improving bag-of-keypoints image categorisation. Technical report, University of Southampton, 2005.

[9] F. Jurie, B. Triggs. Creating efficient codebooks for visual recognition. IEEE International Conference on Computer Vision, 2005, 1:604-610.

[10] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. Machine Learning, 2004, 42:177-196.

[11] H. Jégou, M. Douze, C. Schmid. Packing bag-of-features. IEEE International Conference on Computer Vision, 2009, 2357-2364.

[12] Th. Leung, J. Malik. Representing and Recognizing the Visual Appearance of Materials using Three-dimensional Textons. International Journal of Computer Vision, 2001, 43(1): 29-44.

[13] P. Quelhas, F. Monay, J. Odobez, D. Gatica-Perez, T. Tuytelaars. A Thousand Words in a Scene. Patten Analysis and Machine Intelligence, 2007, 1575-1589.

[14] T. Lindeberg. Feature detection with automatic scale selection. International Journal of Computer Vision. 1998, 30:79-116.

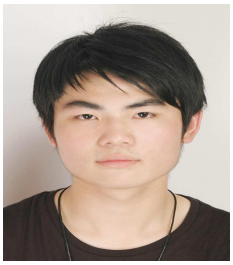
[15] K. Mikolajczyk, C. Schmid. Scale and affine invariant interest point detectors. International Journal of Computer Vision, 2004, 60:63-86.

[16] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, L. Van Gool. A comparison of affine region detectors. International Journal of Computer Vision, 2005, 65:43-72.

[17] R. Weber, H. J. Schek, S. Blott. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. Very large Data Base, 1998, 194-205.

[18] J. Sivic, A. Zisserman. Video google: A text retrieval approach to objects matching in videos. InProceedings of 9th International Conference on Computer Vision, 2003, 1470-1478.

Wang Lipin is an Associate Professor at Tongling University. Her M.Sc. in Project Management from Hefei University of Technology. Her research interests include electronic information, image processing and information security.



Pu Juncheng is a student at Istituto Europeo di Design. His research interests include image processing, curves and its computational aspects.