ACSIJ Advances in Computer Science: an International Journal, Vol. 4, Issue 2, No.14 , March 2015
ISSN : 2322-5157
www.ACSIJ.org

Bag-of-Visual-Words, its Detectors and Descriptors; A Survey in Detail

Mehdi Faraji, Jamshid Shanbehzadeh

Department of Computer Engineering, Faculty of Engineering,
Kharazmi University of Tehran, Iran
farajimhd@gmail.com, shanbehzadeh@gmail.com

Abstract

In this survey, we investigate the Bag-of-Visual-Word technique by an up-down strategy. At the beginning, we explain the general approach and functionality of the method and then we study the combination of various high level ideas and their consequent results yielded by experts and well-known authors. Subsequently, supplementary information will be provided by comparing and discussing full details of detecting and describing interest points. Moments of inertia are also studied because of their crucial role in many computer vision approaches and also their stability over some image deformation which make them a suitable tool for object recognition methods. At the end of this paper, we draw a comparison between invariant functions and covariant function through principal axis of second moments. To provide a deeper understanding, the empirical results of the comparison have been illustrated.

Keywords: *Bag-of-Visual-Word; BoVW; Interest Point; Image Classification; Object Recognition; Region-based Detectors and Descriptors; Moments; Covariant; Invariant; SIFT;*

1. Introduction

Due to limitations of global features during concept extraction from an image, local invariant features (keypoints) are employed. The main question is what kind of keypoint has to be defined? As a computer user in a modern epoch of high tech, the word “patch” is familiar for everyone. A patch is a small repair to a program which fixes its bugs and problems. A keypoint of an image also is a patch on a part of the image which contains its surrounding information. Therefore, if we have several patches on an image, each patch contains local information of that area. Meanwhile, keypoints tend to grasp the most notable local information of their surroundings. Thus, keypoints are *salient* patches that contain rich local information about an image[1]. Extracting a number of keypoints from image and group them based on some correlation criteria like what the clustering algorithms do, represent a method called Bag of Visual Words (BoVW). BoVW has been widely used for general object recognition and image retrieval as well as texture analysis, which are parts of generic object recognition or category recognition[2]. Keypoints of each cluster construct a group called “visual word”. As a process of training, BoW gathers a great number of visual words to build a visual vocabulary and try to represent each image by a histogram of its keypoints mapped into visual words of vocabulary.

As a matter of fact, BoVW is a high level method which is combined with several different techniques to accomplish a recognition task. To achieve this, using a lot of low level methods seems inevitable. These fundamental methods provide us with the first step of obtaining results and can negatively affect performance provided those basic preparations have not been chosen wisely. Thus, a large amount of research have been carried out in order to extend our knowledge of the low level methods of image processing. Those basic approaches can be used as tools to form many desirable algorithms and models for many different types of applications. Consequently, the more robust basic methods we design, the more outstanding performance we achieve.

In field of image classification, matching, recognition, and etc. in addition to understanding of common and general image processing techniques, having knowledge of geometry such as projective geometry, camera models, and etc. can be immensely useful. Many approaches have been proposed based on the underlying concepts of geometry. Some of them obtained great results, see [3, 4]. The various available methods for the fundamental part of image classification are designed based on many arenas such as: Digital Signal Processing, Machine Learning, Computational Geometry, and Mathematics.

In fact, a combination of two vital processings on an image is needed in order to begin a high level image analysis method. In the first place, a detection phase should be performed. Detectors try to find the most appropriate locations of the image, subject to either the point itself or the information content around that point. Albeit, there are many various approaches, most of them consider high frequency variation which make edges and corners. The proposed detectors are developed based on many different ideas such as High Curvature Points, Intensity Based, Biologically Plausible methods, and many other categories. However, the most popular detectors are categorized as Viewpoint Invariant methods divided into Scale-Invariant and Affine Invariant methods[5].

In this survey, we focus on an image classification method named Bag of Visual Words (BoVW) and study the fundamental parts of the method including SIFT and its direct descendants. Following that, we investigate its very basic operation to address several questions like how the detection process is performed or what is the plausible reason for histogram based description of SIFT. The reader will become familiar with various tricks and approaches useful to deal with issues of image categorization and image matching.

This paper is organized as follows: In the next section an overview of the related literature is given. Section 3 explains BoVW and its standard phases. In section 4, we review several important notions of region based detectors and descriptors such as interest point, region or local feature, local invariant features, SIFT, gridSIFT, moment invariant, and invariant versus covariant. Section 5 is also a suggestion for future works.

2. Related Works

One of the most popular methods for text categorization which represents a document based on occurrences of its words, is known as Bag-of-Words. Joachims [6] was the first one who introduced and experimented Bag-of-Words and declared its high capability. After that, Cristianini, Shawe-Taylor [7] extended Joachims work and incorporated extra information to its kernel function. These promising improvements, encouraged machine vision researchers to develop the idea of bag-of-visual-words for generic visual categorization [8-10] and instance recognition [11]. The new approach Bag-of-Visual-Words shortly became a progenitor of a successive wave of research topic in machine vision. The two most chosen topics are image retrieval [11-13] and Image categorization [10, 14, 15].

In BoVW we first define several patches on image and then produce affine invariant features from the patches [16, 17]. To describe these features, descriptors are employed. The most popular descriptor is SIFT (Scale Invariant Feature Transform) [18] which was used at initial introducing of BoVW. It describes local information of detected points by a manipulated histogram of orientations. Several improvements have been performed on SIFT afterwards. For instance, PCA-SIFT [19] used the Principal Component Analysis on the gradient patch instead of histogram of orientation description and achieved a more discriminative results with lower dimensionality. Later on, SURF which its detection phase run at a faster clip, was proposed by Bay, Tuytelaars [20]. According to the fact that descriptors are the crucial components of BoVW, Wu [21] implemented a GPU based version of SIFT including an exhaustive SIFT matcher which multiplied the descriptor matrix on GPU and located its proper matches on GPU. Another great improvement on SIFT was gained through the simulation of different viewpoints of the patch, normalizing its translation and rotation as well, namely ASIFT [3]. It simulated the possible distortions of image by means of longitude and latitude angles of camera optical axis and rotated based on transition tilts parameter which measures the degree of viewpoint change from one view to another [3].

Before describing features of image, a detection phase seems to be necessary. Harris and Stephens [22] proposed a detector based on local auto-correlation function for edge and corner detection. However, one of the most appropriate detectors for BoVW proposed by [18, 23]. Detectors tend to identify locations in image scale space that are invariant with respect to image translation, scaling, and rotation, and are minimally affected by noise and small distortions[23]. Scaled-normalized Laplacian of Gaussian that has been completely studied under some rather general assumptions on scale invariance by Lindeberg [24], and the Gaussian kernel and its derivatives, are the most popular smoothing kernels for scale space analysis. Lowe [23] utilized the Difference of Gaussian functions which had been employed for other purposes by Crowley and Parker [25] and Lindeberg [24] as a close approximation to Scaled-normalized Laplacian of Gaussian.

Although BoVW has been remarkably paid attention to, its weakness in considering spatial information is still being studied [10, 26-29]. Zhou, Zhou [30] have recently proposed an alternative solution to overcome the weakness of bag-of-features model and have achieved an improvement over the SPM method. Their empirical results on five common datasets (**1.** 8-category scenes from Oliva and Torralba [31], **2.** 13-category scenes from Fei-Fei and Perona [29], **3.** 15-category scenes from Lazebnik et al. [10], **4.** 8-category sports events from Li-Jia and Fei-Fei [32], **5.** 67-category indoor scenes from Quattoni and Torralba [33]) including indoor scenes, outdoor scenes, and sport events have demonstrated the convincing performance of their approach.

They also have incorporated the multi-resolution representation into a bag-of-feature model to achieve an effective scene classification [30]. This multi-resolution representation gives them the ability to extract local features which are common in locations but differ in resolution. As a way to represent image globally by local features obtained from all multi-resolution images, they have grouped local features (visual codebook) enjoying an unsupervised clustering algorithm (k-means). Ignoring the spatial information of local patches is the main shortcoming of this method. To mitigate this, they have adopted two modalities of horizontal and vertical partitions, in order to partition all resolution images into sub-regions with different scales[30]. Subsequently, local features of each sub-region should be mapped into the learned visual codebook to produce representation of each sub-region by a histogram of the codeword occurrences. Following that, histograms of sub-regions in the same resolution have joined together to arrange an image representation of that resolution. Based on image pyramids parameters setting in their approach, they have considered three resolutions for each image. So the outcome of image representation in a resolution has constituted a feature channel corresponding to a same resolution.

Yu-Gang Jiang [1] extended their previous works [34, 35] and improved BoVW for semantic concept detection in large-scale multimedia corpus. They examined various representation choices separately and then jointly such as feature weighting, vocabulary size, feature selection and visual bi-gram, which had not been deeply studied in other works. They found out that a weighting scheme for visual words is essential to mitigate the impact of clustering on vocabulary generation. Additionally, among five different feature selection criteria which have been examined, they figured out that Information Gain (IG) and Chi-square (CHI) are the most appropriate ones and make them able to remove half of the vocabularies without hurting the overall performance. Consequently, the computational cost was reduced especially for detecting concepts in large multimedia databases. They did not consider temporal information of video shot and emphasized on just keyframes. The temporal information has been shown to be effective particularly for the detection of event-type concepts in [36] and [37].

As an improvement to BoVW, X. Tian et al. [38] focused on making the codebook more discriminative by considering the manifold geometry of the local feature space in codebook generation process. Although, the clustering based codebook generalization is easy for implementation, it totally ignores the known labeling information of training images [38]. Several methods [17, 39-50] based on five strategies tried to conquer this problem, however, the manifold geometry had not been considered in their methods.

The strategies for constructing a supervised codebook have been described in [38]. Particularly, they enhanced learning strategy of classic unsupervised learning phase of BoVW and employed a subspace learning method for codebook generation. They fascinated by the idea of subspace learning approach to conquer the famous challenge of BoVW model. The subspace learning method finds a contextual local descriptor subspace for embedding the discriminative information[38]. They also considered two aspects of their model (codebook construction, contextual subspace learning) as an optimization problem and tried to urge them to learn simultaneously. The subspaces consisted of same class images which had been represented by BoVW, were created after a process of optimization. The fact that two images should be close together if they belong to the same classes and should be away from each other if they belong to the different classes, constituted constraints which guaranteed the discriminative ability of the optimization process.

All in all, most of the authors as well as X. Tian [38] could not dismiss the key role of the initial processing of the pure local image patches which we have referred to as fundamental processing methods earlier in this paper, and also claimed its remarkable effects on overall performance. Hence, purposeful works and valuable enhancements have been done on these methods.

In short, a specific purpose must be served: the most useful feature is the one with less varied descriptors under different variations and distortions. From initial idea of extracting features in scale space representation [51, 52] to the several developed methods like DoG detector [23], Harris-Laplace detector [53], and their affine normalizations [54, 55], and a famous segmentation based method MSER (Maximally Stable External Region) [56], all have been designed to achieve that aim.

3. Bag of visual Words

This method is inspired by text categorization algorithms (Bag of Words) and focuses on image keypoints. Each keypoint of image should be detect (by detectors) and describe (by descriptors) separately. Images can be represented by set of keypoints, but the sets vary in cardinality and lack meaningful ordering that create difficulties for learning methods[1]. Keypoint clustering is the next step of BoVW. The output of clustering process is a visual word (or codeword, visterm, visual texton) vocabulary which holds information about different local patterns (or a codebook). The size of vocabulary is the total number of clusters, varies from hundreds to over ten thousands[1]. The last step is to assign keypoints to visual words in order to represent an image as BoVW. This representation is analogous to the bag of words document representations in term of forms and semantics. Both representations are sparse and high-dimensional, and just as words convey meaning of a document, visual words reveal local pattern characteristics of the whole image[1].

Gabriella Csurka [8] considered the main step of BoVW as following:

- Detection and description of image patches
- Assigning patch descriptors to a set of predetermined clusters (a vocabulary) with a vector quantization algorithm.
- Constructing a bag of keypoints, which counts the number of patches assigned to each cluster
- Applying a multi-class classifier, treating the bag of keypoints as the feature vector, and thus determine which category or categories to assign to the image.

BoVW can be divided into three main components: detection and description of local features, visual word representation, and classification[2]. Tamaki, Yoshimuta [2] defined BoVW in more details and described the algorithm into two phases:

Training phase

a) Extracting feature points from the training images.
b) Computing feature vectors (descriptors) for each feature point.
c) Clustering feature vectors to generate visual words.
d) Representing each training image as a histogram of visual words.
e) Training classifiers with the histograms of the training images.

Test phase

a) Extracting feature points from a test image.
b) Computing feature vectors (descriptors) for each feature point.
c) Representing the test image as a histogram of visual words.
d) Classifying the test image based on its histogram.

4. Region Based Detectors and descriptors

As it can be seen from the mentioned BoVW steps, the first and second phases deal exclusively with feature points. The two most used operations on feature points are *detection* and *description*. This survey article therefore sets out to review these operations belong to the region based methods.

4.1 Interest Point

By paying a little attention, it is obvious that interest point are coordinates of suitable candidate points in the Cartesian Space or row and column indices of appropriate selected pixels of the image. Apart from the pixel itself, no information can be fetch from the interest point since it does not include its neighborhood pixels. Camera calibration and 3D reconstruction applications are instances of those which employ the interest points (the geometric location of the specified point is the main attraction rather than its neighborhood information) to use in their further processing algorithms. This interest points can be obtained by various feature extraction methods depends on the essence of the application.

4.2 Region or Local Feature

The surrounding pixels of an interest point with any geometric formation adjacent to the interest point which carry some locality information about the interest point, called local feature or region in some applications. Therefore, to describe a local feature, both the location of an interest point and the geometric properties of their enclosing pixels such as size, area, shape, and etc. have to be specified by the desired descriptor.

In every aspects of science, to compare experimental results of different methods, several criteria have been specified. In visual object recognition area also, for distinguishing good features from poor incapable ones, having some special properties is essential, such as *Repeatability*, *Distinctiveness* / *informativeness*, *Locality*, *Quantity*, *Accuracy*, and *Efficiency*. These metrics are usually used in the evaluation process to illustrate the capability of a proposed method. For further information about the evaluation process see [57].

4.3 Local Invariant features

A local feature is an image pattern which differs from its immediate neighborhood[5]. Some usual image properties like intensity, color, and texture which are defined as changing criteria, can conclude points, edges or tiny image patches to be measured from the center point of the patch in order to be converted into descriptors.

Local features have been employed in three categories of image processing and machine vision applications:

Local features for Special Applications like processing aerial images: Dealing with this types of images requires some previous knowledge of the related field to become aware of different interpretations for specific information and geometrical shapes in the image. As there is a priori definition for tools of feature extraction in image (such as edge, corner, blob, etc.), the most suitable terminology for naming the action of this type of features is "*Detect*". For instance edge detector, corner detector, and etc. Evidently, in this category, every edges, corners, and blobs have been predefined semantically, hence *detecting* those points give us the ability to discern "*False Detections*" or "*Missed Detections*" [5].

Local features for Matching and Tracking Applications, Pose Estimation, Image Alignment or mosaicking, Camera Calibration, and 3D reconstruction: Detected features in this category not only must be meticulously localized and accurately located but also have to be uniquely and permanently found in a steady way. Reaching a pertinent representation of the feature is not as important as meeting the latter two requirements [5].

Local features for Object Recognition, Scene Classification, Image retrieval, and video mining: Since the aim of describing this kind of features is not matching, it does not need to be precisely localized like the features which were used for matching and tracking. Therefore resulting an appropriate image representation by a collection of the local features will satisfies the need of preliminaries factors (such as statistical analysis of features) for achieving object recognition without starting with the image segmentation phase [5].

Since no prior knowledge is available for tools (edges, corners, blobs, and etc.) in image, the most appropriate terminology for the features of scenario 2 and 3 is '*Extract*'[5]. To summarize, if the chief characteristics of tools in image has been realized before starting to process, there is something to '*Detect*' in image, otherwise the term *Extract* is substituted, nevertheless the term detector is commonly used for all scenarios logically incorrect [5].

Undoubtedly, if an extracted feature of an image shows its robustness for an application belongs to one of the above three categories, it does not mean that the feature is also suitable for employing in applications of other categories. Mathematically speaking, each problem needs to be solved according to the necessary satisfactions for its constraints and variables, and different kinds of problems needs different clues to reach their solution.

4.4 SIFT

To achieve more robust results, Lowe [23] introduced new class of local image features which showed substantial improvements over its previous approaches. While its prior methods suffered from variance in scale and were vulnerable of projective distortion and illumination changes (Invariance Problem), described features by SIFT were invariant to image scaling, translation and rotation and partially invariant to illumination changes and affine or 3D projection [23]. In fact SIFT is a transformation of desired image into a huge set of local feature vectors with respect to invariance problem. The method have originated based on a model of behavior of complex cells in the cerebral cortex of mammalian vision and shares a number of properties in common with responses of neuron in inferior temporal (IT) cortex in primate vision [23]. In the elementary version of SIFT, Lowe tried to ease the major failing of corner detectors methods [58] which obtained the features only from one specified scale, by enjoying different scaling of an image and determining an extra explicit scale for each point to provide an opportunity for sampling the image description vector at a commensurate scale for each image. For key localization phase, Lowe used the proposition of Lindeberg [24] and convolved a window of Gaussian Kernel as smoothing method on images two times with some considerations in each scale.

$$g(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \tag{4.4.1}$$

Where $g(x)$ is the 1D Gaussian Kernel which was convolved first to image in horizontal and then in vertical direction to attain 2D Gaussian Kernel. After that, he made two smooth images from base image with determining $\sigma = \sqrt{2}$ to give image A1. This process was repeated for second time resulted image A2. The difference of Gaussian function was obtained by subtracting image A2 from image A1.

He extracted image gradients $M_{ij}$and orientation R_{ij} from the obtained smooth image A_{ij} at each level of the pyramid.

$$M_{ij} = \sqrt{\left(A_{ij} - A_{i+1,j}\right)^2 - \left(A_{ij} - A_{i,j+1}\right)^2} \tag{4.4.2}$$

$$R_{ij} = atan2\left(A_{ij} - A_{i+1,j}, A_{i,j+1} - A_{ij}\right) \tag{4.4.3}$$

To make the image descriptors invariant to rotation, he specified a histogram of orientations (R_{ij}) and assigned a canonical orientation to each key location. He defined 36 bins, each bin was contained a range of 10 degree to cover all 360 range of rotations.

Lowe presented a more in-depth development and analysis of his earlier work [23] while obtaining more enhancements in stability and feature invariance, and introduced a new local descriptor that provided more distinctive features including being less sensitive to local image distortions such as 3D viewpoint change [18]. He employed a technique called cascade filtering approach to reduce the cost of feature extractions by applying the more costly computations only at some appropriate tested locations as well. He split the process of extracting local invariant features into four major stages: *scale-space extrema detection*, *keypoint localization*, *orientation assignment* and *keypoint descriptor*.

To explain in more detail, the process is embarked on by searching through all images in every scale to find interest points with substantial potentiality of being invariant to scale and orientation. To achieve this Lowe [18] used the proposition of Lindeberg [24] and employed DoG to obtain the most efficient results and convolved a Gaussian function $G(x, y, \sigma)$ on image $I(x, y)$ to make a smooth image $L(x, y, \sigma)$ (*Scale-space extrema detection*).

$$G(x, y, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{4.4.4}$$

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{4.4.5}$$

To compute DoG, he proposed to subtract a Gaussian function with standard deviation σ from another Gaussian function multiply by a constant multiplicative factor k and then convolve the result on original image and make new DoG image $D(x, y, \sigma)$.

$$D(x, y, \sigma) = G\big((x, y, k\sigma) - G(x, y, \sigma)\big) * I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (4.4.6)$$

After that, the maxima or minima of all available Difference of Gaussian images must be found by comparing each sample point to its eight neighbors in current image and nine neighbors in scale above and below [18]. Experiments have shown that although there is a large number of extrema points, it is possible to choose the most stable and useful subset of them.

After efficient interest points (keypoints or candidate locations) were detected, he employed the proposed method of [59] to fit a mathematical model on the obtained keypoints (*keypoint localization*). The principal measure of the selection in this stage is the keypoint's quality of being stable. Their approach includes using the Tylor expansion of the scale space function for the offset of the sample point $\mathbf{x} = (x, y, \sigma)^T$:

$$D(\mathbf{x}) = D + \frac{\partial D^T}{\partial \mathbf{x}}\mathbf{x} + \frac{1}{2}\mathbf{x}^{\mathbf{T}}\frac{\partial^2 D^T}{\partial \mathbf{x}^2}\mathbf{x} \quad (4.4.7)$$

And then obtaining the location of the extremum of the function $\hat{\mathbf{x}}$ by setting the first derivative of it with respect to $\mathbf{x}$ to zero and calculate $\mathbf{x}$:

$$\hat{\mathbf{x}} = -\frac{\partial^2 D^{-1}}{\partial \mathbf{x}^2}\frac{\partial D}{\partial \mathbf{x}} \quad (4.4.8)$$

To reject unstable extrema with low contrast, he [18] substituted $\hat{\mathbf{x}}$ into the Tylor expansion and obtained the function value at extremum $D(\hat{\mathbf{x}})$:

$$D(\hat{\mathrm{x}}) = D + \frac{1}{2}\frac{\partial D^T}{\partial \mathrm{x}}\mathrm{x} \quad (4.4.9)$$

Therefore, any selected extrema which has the function value $D(\hat{\mathbf{x}})$ less than a threshold (Lowe has employed a threshold equal to 0.03) will be rejected.

To achieve invariance to image rotation, he assigned consistent orientation to each keypoints based on local image properties (gradient directions) and represented the keypoint descriptor relative to the consistent orientation [18] (*orientation assignment*). Furthermore, he sampled image gradient magnitudes and orientation around the keypoint location and weighted them by a Gaussian window and then described them by an orientation histogram.

$$m(x, y) = \left((L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2\right)^{\frac{1}{2}} \quad (4.4.10)$$

$$\theta(x, y) = tan^{-1}\Big(\big(L(x, y+1) - L(x, y-1)\big)/\big(L(x+1, y) - L(x-1, y)\big)\Big) \quad (4.4.11)$$

To find an appropriate form of representation, he employed the idea of [60] that is a model based on human complex cells of biological vision in primary visual cortex. The final result was a transformation to a new representation in which significant levels of local shape distortions and changes in illumination were allowed [18] (*keypoint descriptor*).

Applying SIFT features on image matching and recognition needs to first build a huge database of extracted SIFT features from many images and then match SIFT features of the desired image with the stored SIFT features by comparing their Euclidean distance of their SIFT vectors. Lowe [18] considered fast nearest-neighbor algorithm to perform this computations rapidly against large databases. One difficulty of the SIFT keypoint descriptors which weakens its distinctiveness appears during dealing with cluttered images and causes only a few correct matches in the database for many features from the background. To ease this problem, the correct matches can be obtained by identifying the subsets of keypoints to match with scale, location and orientation of the new image. The determination of these consistent clusters can be performed rapidly by using an efficient hash table implementation of the generalized Hough transform [18]. Subsequently, clusters with either three or more than three matches are chosen for verification. The verification phase is divided to first applying a least-squared estimate to make affine approximation to the object pose in order to being capable of removing outliers which are not consistent to the affine approximation. Next step is to compute a probability measure for a particular set of features that considers the accuracy of fit and number of probable false matches as well.

4.5 gridSIFT

Although the keypoints which is extracted by SIFT as the result of using DoG, are fruitful to perform recognition task, the problem of the sparse generated descriptors still has adverse effects on its overall performance. Therefore, several researchers focused on detecting keypoints from fixed locations of image [9, 16, 29] and claimed to have better results if dense features are extracted instead of interest point features [10]. They combined global features with the use of local features by first segmenting the image into several sub-images and then computing the histogram of each sub-region by exploiting local patches of that sub-region[10].

Fei-Fei, L. and P. Perona [29] have tested following different ways of extracting local regions:

- *Lowe Dog Detector* [18].
- *Kadir and Brady Saliency Detector* [61].
- *Random Sampling:* During a random process, several patches is selected from the image with a randomly chosen size between 10 to 30 pixels. Nowak [9] compared random sampling over five different image datasets with two other samplers (Harris-Laplace and Laplacian of Gaussian) and showed with experimental results that random sampling outperformed two other samplers. They declared that the number of sampled patches is the most prominent parameter influencing overall performance. However, if fixed small number of patches are sampled, none of the samplers will dominate the other's performance [9].
- *Evenly Sampled Grid*: An evenly sampled grid spaced at 10×10 pixels for a given image. The size of the patch is randomly sampled between scale 10 to 30 pixels.[29]

The latter approach has been named gridSIFT. It segments the image into a grid and samples every patch of the grid independently and obtains densely SIFT descriptors.

Fei-Fei, L. and P. Perona have also tested all of the four different ways on two dissimilar representation of patches: normalized 11×11 pixel gray values [29] and a 128-dimension SIFT vector[23]. Their experimental results reported that in each ways of extracting local regions, the performance of the 128-Dimensional SIFT vector is significantly greater than normalized 11×11 pixel gray values and is more robust.

To perform grid sampling two factors are necessary to be considered. First, which is referred to Grid Spacing, determines the distance between each two features. If more number of features are needed, the space between samples must be decreased. It affects how densely the features are extracted [2]. The scale of a patch around each center of the sample is the second concern. GridSIFT needs a range of scale to be specified around each point in order to participate the appropriate spatial information involved in the feature at the point [2].

In particular, to reach an eventual SIFT descriptor of a sample point, two methods have been proposed. To explain in more details, consider: D as the *Dimension* of the feature vector, P as the number of sample *Point*, sp as the current *sample point* or in other word gridSIFT descriptor, S as the number of *Scales*, v as the feature *vector* and f as the *feature*. Each feature vector is formulated as:

$$v_{sp} = \left(f_{sp.1}, f_{sp.2}, \dots, f_{sp.D}\right) \tag{4.5.1}$$

In first method, sp is set as below:

$$1 \le sp \le S \times P \tag{4.5.2}$$

Therefore it generates the following feature vectors:

$$\begin{gathered} v_1 = (f_{1.1}, f_{1.2}, \dots, f_{1.D}) \\ v_2 = (f_{2.1}, f_{2.2}, \dots, f_{2.D}) \\ \vdots \\ v_{S \times P} = (f_{S \times P.1}, f_{S \times P.2}, \dots, f_{S \times P.D}) \end{gathered} \tag{4.5.3}$$

Obviously, it is specifying that the dimension of feature vectors are independent from the number of scales, however the number of gridSIFT descriptors is changed by defining more or less scales for each sample point. To conclude, a D dimensional gridSIFT descriptor is extracted for each scales of each sample points without any correlation with other feature in the scales of the same sample point.

The second method is also called *variant multi scale gridSIFT*, set the sp as:

$$1 \le sp \le P \tag{4.5.4}$$

Subsequently, it defines each vector feature by combining every scale's vector features which is belonged to a same sample point:

$$\begin{gathered} v_1 = \begin{pmatrix} f_{1.1}, f_{1.2}, \dots, f_{1.D}, \\ f_{2.1}, f_{2.2}, \dots, f_{2.D}, \\ \dots, \\ f_{S.1}, f_{S.2}, \dots, f_{S.D} \end{pmatrix}_{S \times D} \\ v_2 = \begin{pmatrix} f_{S+1.1}, f_{S+1.2}, \dots, f_{S+1.D}, \\ f_{S+2.1}, f_{S+2.2}, \dots, f_{S+2.D}, \\ \dots, \\ f_{2S.1}, f_{2S.2}, \dots, f_{2S.D} \end{pmatrix}_{S \times D} \\ \vdots \\ v_P = \begin{pmatrix} f_{(P-1)S+1.1}, f_{(P-1)S+1.2}, \dots, f_{(P-1)S+1.D}, \\ f_{(P-1)S+2.1}, f_{(P-1)S+2.2}, \dots, f_{(P-1)S+2.D}, \\ \dots, \\ f_{PS.1}, f_{PS.2}, \dots, f_{PS.D} \end{pmatrix}_{S \times D} \end{gathered} \tag{4.5.5}$$

Consequently, it shows that the dimension of each vector is strictly determined based on the number of scales surrounding that sample point and the dimension of each scale's feature. By employing this strategy the number of feature vectors is decreased to P, in contrast, the dimension of each feature vector (D) is increased to $S \times D$.

To boost the performance and make the feature specifically adapted for use by that type of the problem which is being studied, some researchers employed Spatial Pyramid Matching [10] and Pyramid Histogram of Visual Words (PHOW) [16]. The orientation information of SIFT descriptor was not used in [2] since they have claimed that spatial information and orientation in NBI[1] images are less

[1] Narrow Band Imaging

informative than those in images used for category recognition.

4.6 Moment Invariants

Obtaining *invariant* results is the ultimate goal of many studies. To explain the issue in more detail, consider the nth-moment of inertia:

$$m_n = \int_{-\infty}^{+\infty} x^n I(x)dx \tag{4.6.1}$$

Where *I(x)* denotes a 1D vector or a distribution of one random variable *x*. Different values of *n* determine a variety of notions such as total area under the function of *I(x)*, the mean (Expected) value of random variable *x* for zeroth-moment (*n=0*) and the first moment (*n=1*) respectively.

$$m_0 = \int_{-\infty}^{+\infty} I(x)dx \quad \textbf{Zeroth-Moment} \tag{4.6.2}$$

$$m_1 = \int_{-\infty}^{+\infty} xI(x)dx \quad \textbf{The First Moment} \tag{4.6.3}$$

To describe the variation of the distribution about the mean, central moment is used.

$$\mu_n = \int_{-\infty}^{+\infty} (x-\bar{x})^n I(x)dx \tag{4.6.4}$$

Where *E(x)* is the mean value of the distribution, and similarly different values of *n* defined the 'spread out' of the probability distribution function (in this case *I(x)*, since we use image intensity later). For instance, if *n=2*, the formula (4.6.4) explains the *Variance of distribution* which is the most common central moment.

$$\mu_2 = \int_{-\infty}^{+\infty} (x-\bar{x})^2 I(x)d(x-\bar{x}) \tag{4.6.5}$$

Obviously, for manipulating the grayscale image intensities, two random variables (*x,y*) is needed. Therefore, the probability density functions or image intensity matrix is denoted by *I(x,y)* and the moment and central moment formulas are substituted by the following.

$$m_{pq} = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} x^p y^q I(x,y)dxdy \tag{4.6.6}$$

$$\mu_{pq} = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} (x-\bar{x})^p (y-\bar{y})^q I(x,y)d(x-\bar{x})d(y-\bar{y}) \tag{4.6.7}$$

As the moment defined by two random variables, the image produced by *I(x,y)* describes a mass on the Cartesian Space. To attain the coordinates of the Center of Mass (CoM), first ordered moments (m_{00}, m_{01}, m_{10}) are employed. Note that the order of the moments is the summation of *p* and *q* (order of moment = *p+q*). Hence the coordinates of CoM are:

$$\boldsymbol{Coordiantes\ of\ CoM} = \begin{cases} \bar{x} = \frac{m_{10}}{m_{00}} \\ \bar{y} = \frac{m_{01}}{m_{00}} \end{cases} \tag{4.6.8}$$

The main idea of performing visual pattern recognition by moments of inertia was first presented by [62] and attracted a lot of interest. After half a century, this field of study has been also fully investigated for complex mathematical concepts, see [63, 64].

In the case of employing moments in image and visual signals processing, Hu [62] presented six absolute orthogonal invariant for second and third order moments and also one skew orthogonal invariant which is useful in distinguishing mirror images:

$$\Phi_1 = \mu_{20} + \mu_{02} \tag{4.6.9}$$

$$\Phi_2 = (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2 \tag{4.6.10}$$

$$\Phi_3 = (\mu_{30} - 3\mu_{12})^2 + (\mu_{21} + 3\mu_{03})^2 \tag{4.6.11}$$

$$\Phi_4 = (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2 \tag{4.6.12}$$

$$\Phi_5 = (\mu_{30} + 3\mu_{12})(\mu_{30} + \mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] + (3\mu_{21} - \mu_{03})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] \tag{4.6.13}$$

$$\Phi_6 = (\mu_{20} - \mu_{02})[(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] + 4\mu_{11}(\mu_{30} + \mu_{12})(\mu_{21} + \mu_{03}) \tag{4.6.14}$$

$$\Phi_7 = (3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] - (\mu_{30} - 3\mu_{12})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \tag{4.6.15}$$

The moments of inertia or the second ordered moments (m_{20}, m_{11}, m_{02}) are useful tools to specify some object features.

4.6.1 Principal axis

Second order moments are able to specify the principal axes which are a pair of axes where the second moments of inertia are minimum (minor principal axis) and maximum (major principal axis) [62]. An important feature of an image (orientation of an image), which is the direction of principal axes of image, can be obtained by computing the angle of closest principal axis to *x* axis:

$$\theta = \frac{1}{2}\tan^{-1}\left(\frac{2\mu_{11}}{\mu_{20}-\mu_{02}}\right) \tag{4.6.1.1}$$

4.7 Invariant versus Covariant

To have a more accurate understanding, we should notice to the specific discrimination between the *Invariant* and *Covariant* functions. For instance, the area of a 2D shape is *invariant* under 2D rotation because its value never changed with this kind of transformation [5]. Therefore, if applying a transformation to the argument of the function does not make new output values for that function, it is considered as an invariant function:

$$F\{I(x,y)\} = F\{Trnsf[I(x,y)]\} \tag{4.7.1}$$

For rotation the transformation function $Trnsf(I)$ is:

$$Trnsf\{I(x,y)\} = Rot\{I(x,y)\}: \begin{bmatrix} x' \\ y' \end{bmatrix}$$
$$= \begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad where\ \varphi\ is\ the\ angle\ of\ rotation \tag{4.7.2}$$

The following formula shows the definition of the *covariant* function:

$$Trnsf\{F[I(x,y)]\} = F\{Trnsf[I(x,y)]\} \tag{4.7.3}$$

Obviously, if the transformation is applied to the argument of the function, the same effect will be obtained as if it is applied to the output of that function. Again, we consider the rotation for transformation function and orientation of the major axis of inertia (θ) as a substitution for function F:

$$Rot\{\Theta[I(x,y)]\} = \Theta\{Rot[I(x,y)]\} \tag{4.7.4}$$

We illustrate some experimental results in table 4.7.1, it is obvious that the formula is true in any conditions. Therefore, the orientation of the major axis of inertia (θ) is *covariant* under the rotation of the image.

In the other hand, if we consider $\Phi_1 = \mu_{20} + \mu_{02}$ instead of function F, we see that Φ_1 satisfies the first definition, hence it is *invariant* under the rotation of the image.

$$\Phi_1\{I(x,y)\} = \Phi_1\{Rot[I(x,y)]\} \tag{4.7.5}$$

Function $\Phi_1\{I(x,y)\}$ is formulated by normalized central moments which proposed by [65]:

$$\Phi_1\{I(x,y)\} = {\eta_{20}}^{I(x,y)} + {\eta_{02}}^{I(x,y)} \tag{4.7.6}$$

Where: $$\eta_{pq} = \frac{{\mu_{pq}}^{I(x,y)}}{m_{00}^{\frac{p+q}{2}+1}} \quad for\ p+q > 1 \tag{4.7.7}$$

Where ${\mu_{pq}}^{I(x,y)}$ is the second order moments of image denoted by $I(x,y)$.

The empirical results which are demonstrated in table 4.7.1, aim to compare *invariant* and *covariant* functions in respect to image rotation. To achieve this goal, the function $\theta\{I(x,y)\}$ (formula 4.7.9) and $\Phi_1\{I(x,y)\}$ (formula 4.7.6) with considering the stated constraint ($|\theta| < \frac{\pi}{4}$) by [62], are defined as:

$$\theta\{I(x,y)\} = \frac{1}{2}\tan^{-1}\left(\frac{2{\mu_{11}}^{I(x,y)}}{{\mu_{20}}^{I(x,y)} - {\mu_{02}}^{I(x,y)}}\right) \tag{4.7.8}$$

$$\boldsymbol{\Theta}\{\boldsymbol{I(x,y)}\} = \theta\{I(x,y)\} + k\frac{\pi}{2}$$
$$for\ specified\ \boldsymbol{k}\ illustrated\ in\ fig\ 4.7.1 \tag{4.7.9}$$

To draw a comparison between the deviation from the mean of both invariant and covariant examples in the table, we consider the Δ function as:

$$\Delta_\theta = Rot\{\Theta[I(x,y)]\} - \boldsymbol{\Theta}\{Rot[I(x,y)]\} \tag{4.7.10}$$

$$\Delta_\Phi = \Phi_1\{I(x,y)\} - \Phi_1\{Rot[I(x,y)]\} \tag{4.7.11}$$

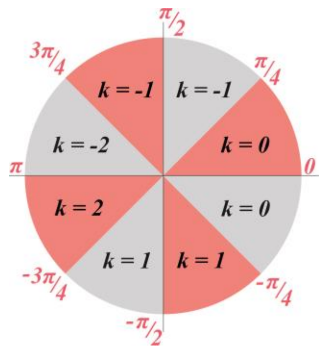

Fig. 4.7.1 Values for k according to the location of rotation angle for formula (4.7.9)

Based on the similarity of the tilt angles (because of the ambiguity[63]), Table 4.7.1 is divided up into several categories which are illustrated by a thick line. Each category has considerably similar results in both covariant and invariant function. The two last columns obtained from formulas 4.7.11 and 4.7.12 respectively, indicate the error and emphasize that the difference of outputs are significantly small and can be omitted in some special applications.

Finally, the function $Rot\{I(x,y)\}$ indicates a standard image rotation algorithm. To come to a greater understanding, Fig. 4.7.2 illustrates the vectors of orientations which are also drawn on Cartesian space by substituting the image with an ellipse rotated through the image's angle of rotation. The column (1) of Fig. 4.7.2 illustrates the rotated image including its principal axis (specified by red arrows) and major and minor axis in respect with the two conditions adapted to resolve the ambiguity by [63] (specified by green arrows). One condition stated that the tilt angle must be an angle between the semimajor axis and the x axis [63]. The main reason why we consider k as a coefficient of $\frac{\pi}{2}$ in formula 4.7.9 is to consider aforementioned conditions. The column (2) of Fig. 4.7.2 shows the image ellipse of the main image which is rotated through the specified angle. In fact, the first and the second columns of Fig. 4.7.2 are the results of formulas $\boldsymbol{\Theta}\{Rot[I(x,y)]\}$ and $Rot\{\Theta[I(x,y)]\}$ respectively.

5. Future works

Many efforts have been performed to simulate as many forms of distortions (such as change in viewpoint angle) as possible in addition to normalize rotation. All in all, aim to reach an invariant representation and description of image which is currently a challenging area. The available approaches suffer from being inaccurate and time consuming. The ultimate goal is to create an invariant methods which run as quick as possible and produce fully affine invariant descriptors.

Table 4.7.1 Comparison of Invariant and Covariant functions.
In each row of the table, the value of Δ_θ and $\Delta_\Phi$specify the error of the function. One reason for that can be the neglect of dx during the conversion from the continuous function to a discrete function.

Angle of Rotation	$\theta\{Rot[I(x,y)]\}$	$Rot\{\Theta[I(x,y)]\}$	$\Phi_1\{Rot[I(x,y)]\}$	$\Phi_1\{I(x,y)\}$	Δ_θ	Δ_Φ
30	-30.0764	-30.0456	0.3467	0.3466	0.0308	-0.0002
120	-30.0739	-30.0456	0.3467	0.3466	0.0283	-0.0002
-150	-30.0709	-30.0456	0.3467	0.3466	0.0252	-0.0002
-60	-30.0702	-30.0456	0.3467	0.3466	0.0245	-0.0002
105	-15.0352	-15.0456	0.3465	0.3466	-0.0104	0.0002
-75	-15.0352	-15.0456	0.3465	0.3466	-0.0104	0.0002
-165	-15.0352	-15.0456	0.3465	0.3466	-0.0104	0.0002
15	-15.0352	-15.0456	0.3465	0.3466	-0.0104	0.0002
0	-0.0456	-0.0456	0.3466	0.3466	0	0
-90	-0.0456	-0.0456	0.3466	0.3466	0	0
-180	-0.0456	-0.0456	0.3466	0.3466	0	0
180	-0.0456	-0.0456	0.3466	0.3466	0	0
90	-0.0456	-0.0456	0.3466	0.3466	0	0
75	14.9588	14.9544	0.3466	0.3466	-0.0044	0
-15	14.9588	14.9544	0.3466	0.3466	-0.0044	0
-105	14.9588	14.9544	0.3465	0.3466	-0.0044	0
165	14.9588	14.9544	0.3465	0.3466	-0.0044	0
60	29.9428	29.9544	0.3464	0.3466	0.0116	0.0002
-120	29.9428	29.9544	0.3464	0.3466	0.0116	0.0002
-30	29.9435	29.9544	0.3464	0.3466	0.0108	0.0002
150	29.9469	29.9544	0.3464	0.3466	0.0075	0.0002
135	44.9193	44.9544	0.3462	0.3466	0.0351	0.0004
-45	44.9198	44.9544	0.3462	0.3466	0.0345	0.0004
-135	44.9199	44.9544	0.3462	0.3466	0.0345	0.0004
45	44.9199	44.9544	0.3462	0.3466	0.0345	0.0004

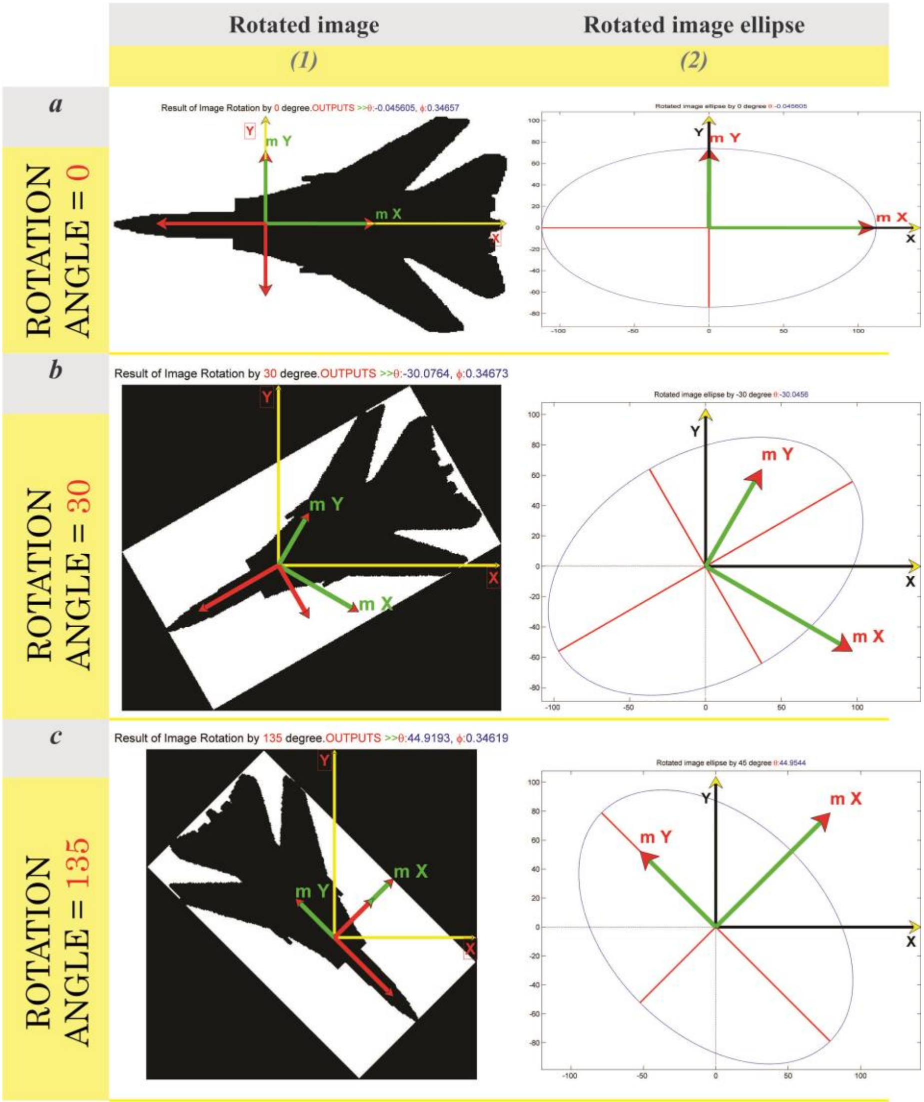

Fig. 4.7.2 Drawing of principal axis on the rotated image plus the rotated image ellipse for three angles.

References

1. Yu-Gang Jiang, J.Y., Chong-Wah Ngo, Member, IEEE, and Alexander G. Hauptmann, Member, IEEE, *Representations of Keypoint-Based Semantic Concept Detection: A Comprehensive Study.* IEEE TRANSACTIONS ON MULTIMEDIA, 2010. 12.
2. Tamaki, T., et al., *Computer-aided colorectal tumor classification in NBI endoscopy using local features.* Med Image Anal, 2013. 17(1): p. 78-100.
3. Morel, J.-M. and G. Yu, *ASIFT: A new framework for fully affine invariant image comparison.* SIAM Journal on Imaging Sciences, 2009. 2(2): p. 438-469.
4. Mikolajczyk, K. and C. Schmid, *Scale & affine invariant interest point detectors.* International journal of computer vision, 2004. 60(1): p. 63-86.
5. Tuytelaars, T. and K. Mikolajczyk, *Local invariant feature detectors: a survey.* Foundations and Trends® in Computer Graphics and Vision, 2008. 3(3): p. 177-280.
6. Joachims, T., *Text categorization with support vector machines: Learning with many relevant features*. 1998: Springer.
7. Cristianini, N., J. Shawe-Taylor, and H. Lodhi, *Latent semantic kernels.* Journal of Intelligent Information Systems, 2002. 18(2-3): p. 127-152.
8. Gabriella Csurka, C.R.D., Lixin Fan, Jutta Willamowski, Cédric Bray, *Visual Categorization with Bags of Keypoints*. Workshop on statistical learning in computer vision, ECCV., 2004. 1.
9. Nowak, E., F. Jurie, and B. Triggs, *Sampling strategies for bag-of-features image classification*, in *Computer Vision–ECCV 2006*. 2006, Springer. p. 490-503.
10. Lazebnik, S., C. Schmid, and J. Ponce, *Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories*, in *IEEE Conference on Computer Vision & Pattern Recognition*2006. p. 2169-2178.
11. Josef Sivic, A.Z., *Video Google: A Text Retrieval Approach to Object Matching in Videos.* Ninth IEEE

International Conference on Computer Vision (ICCV 2003) 2003. 2-Volume Set.
12. Nister, D. and H. Stewenius. *Scalable recognition with a vocabulary tree*. in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*. 2006. IEEE.
13. Wu, Z., et al. *Bundling features for large scale partial-duplicate web image search*. in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. 2009. IEEE.
14. Wang, C., D. Blei, and F.-F. Li. *Simultaneous image classification and annotation*. in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. 2009. IEEE.
15. Si, Z., et al. *Learning mixed templates for object recognition*. in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. 2009. IEEE.
16. Bosch, A., X. Muñoz, and R. Martí, *Which is the best way to organize/classify images by content?* Image and Vision Computing, 2007. 25(6): p. 778-791.
17. Zhang, J., et al., *Local features and kernels for classification of texture and object categories: A comprehensive study.* International journal of computer vision, 2007. 73(2): p. 213-238.
18. Lowe, D.G., *Distinctive image features from scale-invariant keypoints.* International journal of computer vision, 2004. 60(2): p. 91-110.
19. Ke, Y. and R. Sukthankar. *PCA-SIFT: A more distinctive representation for local image descriptors*. in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*. 2004. IEEE.
20. Bay, H., T. Tuytelaars, and L. Van Gool, *Surf: Speeded up robust features*, in *Computer Vision–ECCV 2006*. 2006, Springer. p. 404-417.
21. Wu, C., *SiftGPU: A GPU implementation of scale invariant feature transform (SIFT)*, 2007.
22. Harris, C. and M. Stephens. *A combined corner and edge detector*. in *Alvey vision conference*. 1988. Manchester, UK.
23. Lowe, D.G. *Object recognition from local scale-invariant features*. in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*. 1999. Ieee.
24. Lindeberg, T., *Scale-space theory: A basic tool for analyzing structures at different scales.* Journal of applied statistics, 1994. 21(1-2): p. 225-270.
25. Crowley, J.L. and A.C. Parker, *A representation for shape based on peaks and ridges in the difference of low-pass transform.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, 1984(2): p. 156-170.
26. Liu, J. and M. Shah. *Scene modeling using co-clustering*. in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. 2007. IEEE.
27. Battiato, S., et al., *Spatial hierarchy of textons distributions for scene classification*, in *Advances in Multimedia Modeling*. 2009, Springer. p. 333-343.
28. Bosch, A., A. Zisserman, and X. Muoz, *Scene classification using a hybrid generative/discriminative approach.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2008. 30(4): p. 712-727.
29. Fei-Fei, L. and P. Perona. *A bayesian hierarchical model for learning natural scene categories*. in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. 2005. IEEE.
30. Zhou, L., Z. Zhou, and D. Hu, *Scene classification using a multi-resolution bag-of-features model.* Pattern Recognition, 2013. 46(1): p. 424-433.
31. Oliva, A. and A. Torralba, *Modeling the shape of the scene: A holistic representation of the spatial envelope.* International journal of computer vision, 2001. 42(3): p. 145-175.
32. Li, L.-J. and L. Fei-Fei. *What, where and who? classifying events by scene and object recognition*. in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. 2007. IEEE.
33. Quattoni, A. and A. Torralba, *Recognizing indoor scenes.* 2009.
34. Jiang, Y.-G., C.-W. Ngo, and J. Yang. *Towards optimal bag-of-features for object categorization and semantic video retrieval*. in *Proceedings of the 6th ACM international conference on Image and video retrieval*. 2007. ACM.
35. Yang, J., et al. *Evaluating bag-of-visual-words representations in scene classification*. in *Proceedings of the international workshop on Workshop on multimedia information retrieval*. 2007. ACM.
36. Sun, J., et al. *Hierarchical spatio-temporal context modeling for action recognition*. in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. 2009. IEEE.
37. Wang, F., Y.-G. Jiang, and C.-W. Ngo. *Video event detection using motion relativity and visual relatedness*. in *Proceedings of the 16th ACM international conference on Multimedia*. 2008. ACM.
38. Tian, X. and Y. Lu, *Discriminative codebook learning for Web image search.* Signal Processing, 2013. 93(8): p. 2284-2292.
39. Jurie, F. and B. Triggs, *Creating efficient codebooks for visual recognition*, in *10th International Conference on Computer Vision*2005. p. 604-610 Vol. 1.
40. Perronnin, F., et al., *Adapted vocabularies for generic visual categorization*, in *Computer Vision–ECCV 2006*. 2006, Springer. p. 464-475.
41. Liu, J., Y. Yang, and M. Shah. *Learning semantic visual vocabularies using diffusion distance*. in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. 2009. IEEE.
42. Wu, L., S.C. Hoi, and N. Yu. *Semantics-preserving bag-of-words models for efficient image annotation*. in *Proceedings of the First ACM workshop on Large-scale multimedia retrieval and mining*. 2009. ACM.
43. Moosmann, F., W. Triggs, and F. Jurie, *Randomized clustering forests for building fast and discriminative visual vocabularies.* 2006.
44. Perronnin, F. and C. Dance. *Fisher kernels on visual vocabularies for image categorization*. in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. 2007. IEEE.
45. Mairal, J., et al., *Supervised dictionary learning.* arXiv preprint arXiv:0809.3083, 2008.
46. Lazebnik, S. and M. Raginsky, *Supervised learning of quantizer codebooks by information loss minimization.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2009. 31(7): p. 1294-1309.
47. Marszalek, M. and C. Schmid. *Semantic hierarchies for visual object recognition*. in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. 2007. IEEE.

48. Lian, X.-C., et al. *Probabilistic models for supervised dictionary learning*. in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. 2010. IEEE.
49. Wang, L. *Toward a discriminative codebook: codeword selection across multi-resolution*. in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. 2007. IEEE.
50. Yang, L., et al. *Unifying discriminative visual codebook generation with classifier training for object category recognition*. in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. 2008. IEEE.
51. Lindeberg, T., *Detecting salient blob-like image structures and their scales with a scale-space primal sketch: a method for focus-of-attention.* International Journal of Computer Vision, 1993. 11(3): p. 283-318.
52. Lindeberg, T., *Feature detection with automatic scale selection.* International journal of computer vision, 1998. 30(2): p. 79-116.
53. Mikolajczyk, K. and C. Schmid. *Indexing based on scale invariant interest points*. in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*. 2001. IEEE.
54. Lindeberg, T. and J. Gårding, *Shape-adapted smoothing in estimation of 3-D shape cues from affine deformations of local 2-D brightness structure.* Image and vision computing, 1997. 15(6): p. 415-434.
55. Lindeberg, T. *Direct estimation of affine image deformations using visual front-end operations with automatic scale selection*. in *Computer Vision, 1995. Proceedings., Fifth International Conference on*. 1995. IEEE.
56. Matas, J., et al., *Robust wide-baseline stereo from maximally stable extremal regions.* Image and vision computing, 2004. 22(10): p. 761-767.
57. Mikolajczyk, K. and C. Schmid, *Performance evaluation of local descriptors.* IEEE Trans Pattern Anal Mach Intell, 2005. 27(10): p. 1615-30.
58. Schmid, C. and R. Mohr, *Local grayvalue invariants for image retrieval.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, 1997. 19(5): p. 530-535.
59. Brown, M. and D.G. Lowe. *Invariant Features from Interest Point Groups*. in *BMVC*. 2002.
60. Edelman, S., N. Intrator, and T. Poggio, *Complex cells and object recognition.* 1997.
61. Kadir, T. and M. Brady, *Saliency, scale and image description.* International Journal of Computer Vision, 2001. 45(2): p. 83-105.
62. Hu, M.-K., *Visual pattern recognition by moment invariants.* Information Theory, IRE Transactions on, 1962. 8(2): p. 179-187.
63. Teague, M.R., *Image analysis via the general theory of moments*.* JOSA, 1980. 70(8): p. 920-930.
64. Wallin, Å. and O. Kubler, *Complete sets of complex Zernike moment invariants and the role of the pseudoinvariants.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, 1995. 17(11): p. 1106-1110.
65. Chen, C.-C., *Improved moment invariants for shape discrimination.* Pattern recognition, 1993. 26(5): p. 683-686.