KIET IJCE KIET International Journal of Communications & Electronics
Volume. No. 4, Issue No. 1, January-June 2016, ISSN: 2320 - 8996

Object recognition using Robotic Vision

M.Umamurugan[1], C.S.Sundar Ganesh[2], R.T.Karthick[3]
[1]PG Student, Department of EEE,PSG College of Technology, Coimbatore
[3]Asst.Professor, Department of EEE, PSG College of Teechnology, Coimbatore
[2]Asst.Professor, Department of RAE, PSG College of Teechnology, Coimbatore
css@rae.psgtech.ac.in

***Abstract*—Stable local feature detection and representation is a fundamental component of many image registration and object recognition algorithms. Mikolajczyk and Schmid recentlyevaluated a variety of approaches and identified the SIFT algorithm as being the most resistant to common image deformations. This paper examines (and improves upon) the local image descriptor used by SIFT. Like SIFT, our descriptors encode the salient aspects of the image gradient in the feature point's neighborhood; however, instead of using SIFT's smoothed weighted histograms, we apply Principal Components Analysis (PCA) to the normalized gradient patch. Our experiments demonstrate that the PCAbased local descriptors are more distinctive, more robust to image deformations, and more compact than the standard SIFT representation. We also present results showing that using these descriptors in an image retrieval application results in increased accuracy and faster matching.**

***Index Terms*—SIFT, PCA, Feature Extraction, Matching and Euclidean distance.**

I. INTRODUCTION

Object can be recognized by appearance based method and feature based method. Appearance based methods use template images of the objects to perform recognition .In feature based method, a search is used to find feasible matches between object features and image features. The processing of object recognition has the following stages: feature extraction and feature matching. In computer vision, the Scale Invariant Feature Transform (SIFT) is an algorithm to identify and characterize local features in images. SIFT is well designed and derives distinctive image feature descriptors for image matching but it still suffers from its high cost of computation and high dimensionality. Yan ke presents a more distinctive and more compact feature descriptor, PCA-SIFT, which applies Principal Components Analysis (PCA) to the normalized gradient patch instead of using SIFT's smoothed weighted histograms which results in dimensionality reduction of the descriptors.

II. RELATED WORKS

There is a long history of research in object recognition that has modelled 3D objects using multiple 2D views. Thisincludes eigen space matching [7], which measures distance from a basis set of eigenvalue images; and histogram matching [9] which summarizes image appearance with histograms of selected properties. Earlier work [2] by the author (Lowe, 1999) extended the local feature approach to achieve scale invariance. Lowe [4, 2, 3] overcome such problems by detecting the points of interest over the image and its scales through the location of the local extrema in a pyramidal Difference of Gaussians (DOG). The Lowe's descriptor, which is based on selecting stable features in the scale space, is named the Scale Invariant Feature Transform (SIFT). In the paper [8], the dimensionality reduction of SIFT using Principal Component Analysis (PCA) on each object category is proposed to reduce computational complexity and memory requirement during training process and investigated under the proposed bag of feature object categorization framework. A new strategy to minimize the dimensionality of SIFT features is proposed [11] and the main idea is to do the Principal Component Analysis in the keypoint descriptor computation stage of the standard SIFT.

III. OVERVIEW OF SIFT ALGORITHM

SIFT (Scale Invariant Feature Transform) features are widely used in object recognition. These features are invariant to changes in scale, 2D translation and rotation transformations. To a limited extent they are also robust to 3D projection transformations. SIFT features have the following advantages. They are invariant to scale. They are invariant to 2D transformations like transformations and translation and rotation. Lowe [1], showed that practically these features are invariant to a limited amount of 3D projection transformations. SIFT features are invariant to changes in illumination

The Scale-Invariant Feature Transform (SIFT) is an algorithm to identify and characterize local features in images. It allows for correct object identification with low probability of mismatch and is easy to match against a large database of local features. The representation of the image features has a direct impact on the performance of an object recognition system. Thus, the characterization and evaluation of SIFT's performance is important to advance the research on object recognition.

The SIFT algorithm finds extrema points in scale space, and extracts position, scale, rotation invariant feature vectors. The major stages of computation of SIFT descriptors are divided into four major stages [2]: (1) scale-space extrema detection (i.e., identifying keypoints); (2) keypoint localization; (3) orientation assignment; and (4) keypoint descriptor computation. These stages are used to produce the set of image features. The sections below provide details for these stages.

.

A) *Scale-Space Extrema Detection*

The first stage of calculation is to search over all scales and image locations. The difference-of-Gaussian function is used to detect stable key point locations in scale space. This stage attempts to find those locations and scales that are identifiable from different views of the same object. This can be efficiently achieved by using a scale space function. Furthermore, it has been shown under reasonable assumptions that it must be based on a Gaussian function. The scale space of a two-dimensional image is defined by equation (1) as shown below:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad \ldots\ldots\ldots\ldots(1)$$

Where * is the convolution operator, G(x, y, σ) is a variable-scale Gaussian, and I (x, y) is the input image. The parameter σ is the scale of the key point and is also the standard deviation of the Gaussian function, equation (2).

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-}\,(x^2+y^2)/2\sigma^2 \quad \ldots\ldots\ldots(2)$$

The difference of Gaussians function, $D(x, y, \sigma)$, is used to detect stable keypoint locations in scale space; $D(x,y,\sigma)$ is computed by using the difference between two images, one with scale k times the other. Then, $D(x, y, \sigma)$ is given by equation (3).

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \quad \ldots\ldots\ldots\ldots (3) \end{aligned}$$

To detect the local maxima and minima of $D(x, y, \sigma)$, each point is compared with its 8 neighbors at the same scale, and its 9 neighbors up and down one scale. If this value is the minimum or maximum of all these points then this point is an extrema. The extrema is used as a SIFT key point.

B) Key point Localization

In order to enhance the stability of the follow-up image feature matching and increase the algorithm's anti-noise ability, we need to remove the low-contrast and unstable key points. This stage attempts to eliminate these unstable key points from the final list of key points by finding those that have low contrast or are poorly localized on an edge. This may be achieved by calculating the Laplacian value for each keypoint found in stage one. The location of extrema, z, is given by equation (4).

$$z = -\frac{\partial^2 D^{-1}}{\partial x^2}\frac{\partial D}{\partial x} \quad \ldots\ldots\ldots\ldots (4)$$

C) Orientation Assignment

This step aims to assign a consistent orientation to the key points based on local image properties. The key point descriptor can then be represented relative to this orientation, achieving invariance to rotation. The gradient magnitude, m, and orientation, μ, of (x, y) are given in equations (5) and (6)

$$m(x,y) = \sqrt{(L(x+1, y - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \quad ..(5)$$

$$\mu(x,y) = \tan^{-1}((L(x, y\mp 1) - (L(x, y-1)/(L(x+1, y) - L(x-1, y))) \quad ..(6)$$

D) Key point Descriptor Calculation

The local image gradients are measured at the selected scale in the region around each key point. These are transformed into a representation that allows for significant levels of local shape distortion and change in illumination. The local gradient data, used above, is also used to create key point descriptors. The gradient information is rotated to line up with the orientation of the key point and then weighted by a Gaussian with a variance of 1.5 * the key point scale. These data are then used to create a set of histograms over a window centered on the key point. Key point descriptors typically use a set of 16 histograms, aligned in a 4×4 grid, each with 8 orientation bins, one for each of the main compass directions and one for each of the mid-points of these directions. This process results in a feature vector containing 128 elements.

IV. NEED FOR PCA

SIFT features are invariant to scale and 2D transformations like transformations and translation and rotation. These features are also invariant to a limited amount of 3D projection transformations. However SIFT has its own problems. First, the number of SIFT features that are generated from an image cannot be controlled. The second problem is computational as SIFT features are of high dimension. Thus the major problem in SIFT is its very high dimension. The large computational effort associated with matching all the SIFT features for recognition tasks, limits its usage to many applications.

There are two main reasons that we take the PCA-SIFT descriptor instead of the SIFT descriptor in this paper. First, the standard SIFT feature vector will contain a great number of redundant information, which are undesirable for describing the objects. PCA-SIFT will apply the PCA method to minimize this redundancy which is mainly derived from the background features in the local image patches. Also the shows the dimensionality is reduced for PCA SIFT compared to other methods [2].

V.PRINCIPAL COMPONENT ANALYSIS

Principal Component Analysis (PCA) is a standard technique for dimensionality reduction and has been appliedto a broad class of computer vision problems, including feature selection object recognition and face recognition. While PCA suffers from a number of shortcomings such as its implicit assumption of Gaussian distributions and its restriction to orthogonal linear combinations, it remains popular due to its simplicity. The contribution of this paper lies in rigorously demonstrating that PCA is well-suited to representing key point patches and that this representation significantly improves SIFT's matching performance.

The first three stages of PCA SIFT is same as that of SIFT algorithm. The difference arises in the third stage which is the key point descriptor stage. PCA-SIFT applies Principal Components Analysis (PCA) to the normalized gradient patch instead of using SIFT's smoothed weighted histograms. This algorithm for local descriptors accepts the same input as the standard SIFT descriptor: the sub-pixel location, scale, and dominant orientations of the key point. We extract a 41×41 patch at the given scale, centered over the key point, and rotated to align its dominant orientation to a canonical direction.

VI. PCA SIFT DESCRIPTION

PCA-SIFT can be summarized in the following steps: (1) pre-compute an Eigen space to express the gradient images of local patches; (2) given a patch, compute its local image gradient; (3) project the gradient image vector using the Eigen space to derive a compact feature vector. This feature vector is significantly smaller than the standard SIFT feature vector, and can be used with the same matching algorithms. The Euclidean distance between two feature vectors is used to determine whether the two vectors correspond to the same keypoint in different images.

A) Computation of local image gradient for patches

PCA enables us to linearly-project high-dimensional samples onto a low-dimensional feature space. For our application, this projection (encoded by the patch Eigen space) can be pre-computed once and stored. The input vector is created by concatenating the horizontal and vertical gradient maps for the 41×41 patch centered at the key point. Thus, the input vector has 2×39×39=3042 elements.

More precisely, each of the patches satisfies the following properties: (1) it is centered on a local maximum in scale-space; (2) it has been rotated so that one of the dominant gradient orientations is aligned to be vertical; (3) it only contains information for the scale appropriate to this key point – *i.e.*, the 41×41 patch may have been created from a much larger region from the original image. The remaining variations in the input vector are mainly due to the "identity" of the keypoint or to unmodeled distortions. These remaining variations can be reasonably modelled by low-dimensional Gaussian distributions, enabling PCA to accurately represent them with a compact feature representation.

B) *Projection of gradient image vector*

More importantly, projecting the gradient patch onto the low-dimensional space appears to retain the identity related variation whilediscarding the distortionsinduced by other effects. Each was processed as described above to create a 3042-element vector, and PCA was applied to the covariance matrix of these vectors. The matrix consisting of the top n eigenvectors was stored on disk and used as the projection matrix for PCA-SIFT. The images used in building the Eigen space were discarded and not used in any of the matching experiments.

C)Feature representation

To find the feature vector for a given image patch, we simply create its 3042-element the normalized image gradient vector and project it into the feature space using the stored Eigen space. It is empirically determined good values for the dimensionality of the feature space, n to be 20. The standard SIFT representation employs 128-element vectors whereas PCA SIFT results in significant space benefits. Thus the PCA SIFT feature descriptor obtained is dimensionally reduced compared to that of SIFT descriptor.

VII. IMAGE FEATURE MATCHING

Image matching, also referred to as image correspondence, plays an important role in many aspects of computer vision. For instance, objects recognition, image retrieval, stereo correspondence, building panoramas and so on. Two main problems prevent us from matching progress. First, the number of elementary primitives in an image is large. The other problem is the possible variance between matching image pairs. Translations, rotations, scales and luminance changes can cause the difference of two pictures. It is virtually impossible to compare two images using traditional methods such as a direct comparison between grey values. Two methods enabling a more reliable comparison have been developed: correlation-based methods and feature-based methods. The correlation-based methods still involve all the pixels in images but all pixels will be grouped as windows with certain sizes. On the contrary, feature-based methods just focus on sparse sets of features.

Feature point matching is done as follows. Given an interest feature point in one image, the matches of that point in other image are to be found. The definition of match depends on the matching strategy. The distance between the descriptors is the main similarity criterion. The results for distance threshold-based matching reflect the distribution of the descriptors in space, so this strategy is used. If the distance between the particular pair of feature points falls below the chosen threshold t, this pair is termed as a match.

The Euclidean distance between two feature vectors is calculated to determine whether the two vectors belong to the same keypoint in different images. For each descriptor in image A the distance to all descriptors in image B is calculated. If ratio of the distance of the second closest distance to the closest is greater than or equal to 1.5 (threshold), the descriptor from A is matched to the one from B, otherwise the descriptor in A is not at all matched. This criterion is to avoid having too many false matches for points in image A which are not present in image

VIII. EXPERIMENTAL RESULTS

The matching is performed for real images pairs using both SIFT and PCA SIFT by distance ratio method for various thresholds. Also under different transformations, including rotation, jpeg compression and noise added. For every catalogue, a pair of images are taken in the range from small image transformations to large ones and the transformations are significant enough to illustrate the features of SIFT and PCA SIFT. The test images are illustrated in Figure 2a and Fig 2b.

Fig. 2a Image used for test

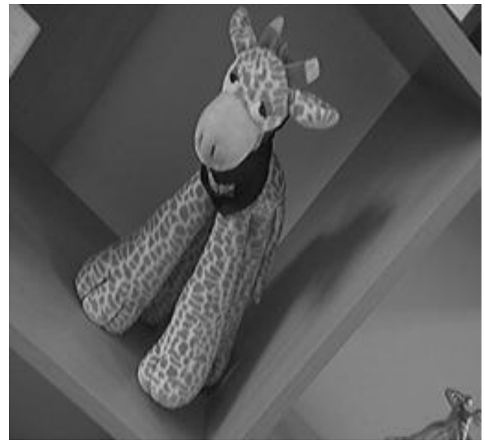

Fig. 2b Image used for test.

The number of keypoints obtained for the original and transformed image pairs for various transformations are shown in the table 2.

Image	**Keypoints Detected**
Giraffe Rotated giraffe	1528 1421

Table 2. Number of Keypoints obtained for image pairs

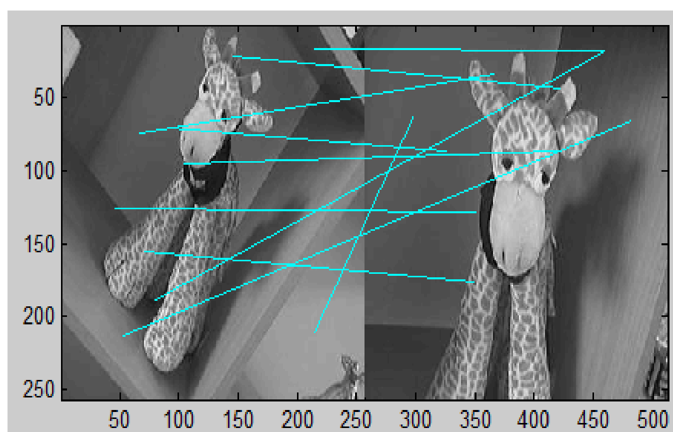

Fig. 3 Matching for the rotated image pair for threshold of 0.8 . Matches obtained by PCA SIFT

	SIFT	PCA SIFT
Dimensionality	High	Low
Dimension Value	128	20 or less
Computation	Less	Low
Memory Requirement	High	Low

Table3. Comparison between SIFT and PCA SIFT

IX. CONCLUSION

In this paper, local feature descriptor matching based on SIFT and PCA SIFT are done. By implementing PCA in SIFT, the dimensionality of the features is reduced. The matching points are also increased so as the accuracy as shown in the table 3. The performance evaluation and the matching local image descriptors by Singular Value Decomposition method instead of Euclidean distance method are considered to be the future work.

REFERENCES

[1] C. Schmid, R. Mohr, "Local Gray value Invariants for Image Retrieval", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.19,PP.530-534,1997

[2] D.G. Lowe, "Distinctive Image Features from Scale Invariant Keypoints", Int'l J. Computer Vision, vol. 2, pp. 91-110, 2004

[3] D.G. Lowe, "Local Feature View Clustering for 3D Object Recognition", Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, Hawaii, 2001 ,pp.682-688.

[4] D.G. Lowe, "Object Recognition from Local Scale Invariant Features", Proc. Seventh Int'l Conf. Computer Vision,Kerkyra, 1999,pp. 1150-1157.

[5] H. Bay, T. Tuytelaars, L. Van Gool, "SURF, Speeded Up Robust Features", Journal of Computer Vision and Understanding,Vol. 110, pp. 346-359, May 2006

[6] K. Mikolojczyk and C. Schmid, "A Performance Evaluation of local descriptors", IEEE Transaction on Pattern Analysis and Machine Intelligence, 2005, pp.1615-1630

[7] Murase, Hiroshi, and Shree K. Nayar, "Visual learning and recognition of 3-D objects from appearance", International Journal of Computer Vision, Vol. 14, pp. 5-24,1995

[8] N. Watcharapinchai , S. Aramvith , S. Siddhichai and S. Marukatat, "Dimensionality Reduction of SIFT using PCA for Object Categorization", International Symposium on Intelligent Signal Processing and Communication Systems, Bangkok, Feb 2008,pp. 1-4.

[9] Schiele, Bernt, and James L. Crowley, "Recognition without correspondence using multidimensional receptive field histograms", International Journal of Computer Vision,Vol.36,pp. 31-52, 2000 .

[10] Sai K. Vuppala, Sorin M. Grigorescu, DanijelaRistic, and Axel Graser, " Robust color Object Recognition for a Service robotic Task in the System FRIEND II", 10th International Conference on Rehabilitation Robotics - ICORR'07,Noordwijk,June 2007,pp. 704-713.

[11] Y. Ke and R. Sukthankar, "PCA-SIFT: a more distinctive representation for local image descriptors", IEEE Computer Society Computer on Computer Vision and Pattern Recognition, June 2004, pp. 506-513.