2016 Conference on Advances in Signal Processing (CASP)
Cummins College of Engineering for Women, Pune. Jun 9-11, 2016

An Approach to Automatic Object Tracking System by Combination of SIFT and RANSAC with Mean Shift and KLT

Bhakti Baheti
Department of E&TC
SGGSIE&T, Nanded-431606
Email : bahetibhakti@sggs.ac.in

Ujjwal Baid
Department of E&TC
SGGSIE&T, Nanded-431606
Email : baidujjwal@sggs.ac.in

Sanjay Talbar
Department of E&TC
SGGSIE&T, Nanded-431606
Email : sntalbar@sggs.ac.in

***Abstract*—Object recognition and tracking are important and challenging tasks in many computer vision applications. Difficulties in object recognition arise due to occlusion, clutter and geometric transformations present between pair of images or frames. Challenges in tracking include ability to deal with abrupt object motion, nonrigid object structures, change in appearance patterns of scene and object, occlusions present and camera motion. To deal with these challenges, we have explored the effectiveness of SIFT for feature extraction and RANSAC for homography estimation in object recognition. This makes system invariant to geometric transformations, illumination variations, partial occlusions and clutter. This automatic object recognition approach is used to automatically detect the object in first frame of video and then it is tracked in subsequent frames. Object tracking is implemented by Mean Shift Algorithm and KLT tracker. These algorithms have ability to handle partial occlusion and clutter. Combination of Mean shift and KLT with SIFT and RANSAC makes the system automatic.**

***Index Terms*—SIFT, RANSAC, Mean Shift, KLT**

I. Introduction

Object recognition is a task in computer vision of finding and identifying objects in an image or video sequence. For humans, this task is very simple. Humans can recognize a multitude of objects in images easily even though they may vary in scale, size, alignment, view points or they may be partially obstructed. But algorithmic description of this task for implementation on machines has been very difficult. The aim of object tracking is to determine the position of the object in video frames continuously and reliably against dynamic scenes [1] i.e. to associate target objects in consecutive frames. Videos are actually sequence of images, called as frames so all the image processing techniques can be applied to individual frames. Thus object tracking is nothing but object recognition step in image processing [2]. From the literature, it is found that it is really a challenging task. Many approaches to object recognition and tracking have been implemented over decades but there is no winning theory.

Temporal differencing and background subtraction are two popular approaches for segmentation of moving object from video frames [3]. These approaches are easy to implement but they are based upon assumption of static background which usually is not applicable in real world environment. These methods are also sensitive to illumination changes, shadow, wind, rain etc. In 2000, Dorin Comaniciu et al. [4] proposed Mean Shift algorithm for real time tracking of non-rigid objects. The tracker had capability to handle real time occlusions and significant clutter. But for selection of target, authors suggested to draw ellipsoidal or rectangular region manually on the first frame. Huiyu Zhou et al. [1] proposed SIFT based Mean Shift algorithm for object tracking (2008). Tracking performance was improved compared to original Mean Shift algorithm but selection of region of interest was again a manual process. Lukas-Kanade algorithm was proposed for image alignment and it is widely used for object tracking [5]. In this algorithm, region of interest can be selected manually or a reference image can be used. Comparative studies show that Lukas-Kanade algorithm is more efficient than other optical flow methods and provides accurate results [6]. Selection of a particular tracking algorithm depends on the application.

Video Surveillance is one of the most active research topic in computer vision and it has wide range of applications. Detection of region of interest is the first step of information extraction and should be automatic. The manual process of target selection in existing tracking algorithms prevents them from application in automatic video surveillance system and this has motivated us to make target selection an automatic process. Scale Invariant Feature Transform proposed by Lowe [7] is an algorithm for image feature points generation. These features are widely being employed in various fields like object detection, image retrieval etc. Several variants of original SIFT have also been proposed like FSIFT, PCA-SIFT, CSIFT, GSIFT etc and each has its own advantage [8]. For example, performance of SIFT and CSIFT is best under scale and rotation changes whereas under illumination and blur change, GSIFT performs best. Using SIFT features and RANSAC for estimating geometrical transformation between two images, robust object recognition can be achieved. Our approach is to detect region of interest automatically in first frame of video by SIFT and RANSAC which is a important step towards

978-1-5090-0849-0/16/$31.00 ©2016 IEEE

automatic video surveillance.

The main contribution of this paper is combination of SIFT and RANSAC (for object recognition) is proposed with tracking algorithms Mean Shift and KLT to make tracking system fully automatic. The remainder of the paper is organized as follows. Section 2 gives an overview of the algorithms used for object recognition i.e SIFT and RANSAC. Tracking algorithms, Mean Shift and KLT are explained in section 3. In section 4, results of object recognition and tracking are discussed. Section 5 presents the conclusion.

II. Object Recognition with SIFT and RANSAC

Object Recognition is actually a labeling problem based on models of known objects. The basic steps in object recognition are feature extraction, feature matching and estimating the geometric transformation between pair of image.

A. Feature Extraction

Object recognition system requires local image features which are unaffected by partial occlusion and nearby clutter. These features should also be robust to geometrical transformations and partially invariant to change in illumination, 3D viewpoint and noise. Also they must be sufficiently distinctive to identify specific objects among many alternatives [9]. Here, we have used SIFT (Scale Invariant Feature Transform) proposed by Lowe [7] for distinctive image features generation. Following are the main steps for obtaining SIFT features [10].

1) Scale Space Construction: Scale spaced image is constructed by convolving the original image with Gaussian operator. This operation blurs the original image. In the next step the original image is resized to half size and again convolved with Gaussian operator to generate blurred out images. This process is repeated till last scale spaced image is formed. Mathematical expression of Gaussian blur is shown in eq 1. This convolution operation is carried out for each pixel. Mathematical equation for convolution of the Gaussian operator and the image is shown in eq 2.

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

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{2}$$

Eq 3 shows generation of Difference of Gaussian (DoG) pyramid from scale space.

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \tag{3}$$

where k is multiplying factor.

Feature points are nothing but the local extrema (maxima or minima) of DoG pyramid. These extrema are detected by comparison of every pixel with its 26 neighboring pixels in the scale space.

2) Keypoint Localization: The local extrema detected in above step are good candidates for keypoints. However, it is observed that either minima or maxima almost never lies exactly on a pixel. To overcome this, subpixel locations are obtained by Taylor expansion of image around the approximate keypoint. Also, only corners are selected for further processing and keypoints having low contrast or lying along edges are rejected.

3) Orientation Assignment: After stable features are determined, a main orientation is assigned to each feature point based on local image gradient to achieve rotation invariance. For each pixel $L(x, y)$ of the region around the feature location, the gradient magnitude $m(x, y)$ and orientation $\theta(x, y)$ are computed as shown in eq 4 and 5 respectively.

$$m(x, y) = \sqrt{a^2 + b^2} \tag{4}$$

$$\theta(x, y) = \tan^{-1}(\frac{b}{a}) \tag{5}$$

where,

$$a = (L(x + 1, y) - L(x - 1, y)) \tag{6}$$

and

$$b = (L(x, y + 1) - L(x, y - 1)) \tag{7}$$

Having magnitude and orientation data, orientation histogram is created and the dominant orientation is calculated for each keypoint.

4) Keypoint Descriptor: The 16 x 16 region is selected around the keypoint and is broken into sixteen 4 x 4 windows. Within each 4 x 4 window, an 8 bin histogram is formed. By doing this on all the windows, we get a 4 x 4 x 8 dimensional vector (SIFT descriptor) for one keypoint. Illumination invariance is achieved by normalizing the descriptor to unity. A keypoint is uniquely identified by its feature vector.

B. Feature Matching

After feature extraction, next step is to match them i.e. determine which features come from corresponding locations in different images. For each feature F_1^i in first image, its corresponding match F_2^j is searched in second image having smallest distance. In an object recognition task, there can be numerous objects and many features in an object. Traditionally used method for feature matching was euclidean distance based approach but it becomes quite slow when large number of features exist and hence is less efficient. To overcome this issue, cosine similarity based approach is used which closely approximates euclidean distance based approach.

C. Homography Estimation with RANSAC

Now we have feature matches established between two images but there may exist some geometric transformations such as translation, rotation, scaling as well as perspective transformations between pair of images. This homography needs to be accurately estimated. Also there can be some

incorrect feature correspondences called as outliers. Presence of outliers severely distorts estimated homography from the desired one and leads to incorrect alignment between images.

Solution to this problem is RANdom SAmple Consensus (RANSAC) algorithm proposed by Fischler and Bolles [11]. RANSAC is widely used for homography estimation as it can robustly estimate the accurate homography even when upto 50 % outliers are present.

The basic steps of algorithm :

1) Randomly select any four pairs of feature matches to estimate the homography as four points are required for plane fitting.
2) Estimate homography matrix by Direct Linear Transform.
3) Calculate how many points are inliers.
4) Calculate the ratio of number of inliers to the total number of points present in set.
5) If this ratio is greater than the predefined ratio T, then again estimate the homography matrix / model parameters using all the identified inliers and terminate. Otherwise repeat steps 1 to 4 (for N iterations)

In this way, the object can be correctly recognized in a test image even if geometric transformations exist between two images. These object recognition steps are applied on first frame of video sequence to automatically detect the object and then it is tracked in subsequent frames.

III. Object Tracking with Mean Shift and KLT Tracker

The goal of object tracking is to keep track of an object's motion and position. We have implemented two modern approaches for tracking, Mean Shift and KLT tracker. In literature, it is suggested to define a rectangle manually on the region of interest (which is object to be tracked) in the first frame of video sequence and then it is tracked later. But here, we are detecting the region of interest in the first frame of video with SIFT and RANSAC, making the system automatic.

A. Mean Shift

The Mean Shift algorithm is an efficient approach for tracking objects whose appearance is defined by histograms (color, texture etc.) [1]. The simple concept of Mean Shift clustering can be extended to tracking as explained in steps below.

1) Define Region of Interest in the first frame of video. We are selecting it automatically, by SIFT and RANSAC as explained earlier.
2) Next step is to model the target in feature space. Here, the object is being modeled using weighted colour histogram. Weights are assigned using Gaussian kernel. Probability density function of target is denoted by $\mathbf{q}$.

$$\hat{\mathbf{q}} = \{\hat{q}_u\}_{u=1\ldots m} \qquad \sum_{u=1}^{m} \hat{q}_u = 1 \tag{8}$$

3) Initialize the estimated position of target in the next frame same as in current frame i.e. $y_1 = y_0$ and $x_1 = x_0$.
4) Calculate candidate probability density function (weighted color histogram) $\mathbf{p}$ in the same way as $\mathbf{q}$ is calculated.

$$\hat{\mathbf{p}} = \{\hat{p}_u\}_{u=1\ldots m} \qquad \sum_{u=1}^{m} \hat{p}_u = 1 \tag{9}$$

5) Having target and candidate probability distributions, Bhattacharya Coefficient ρ is calculated for similarity measurement as shown in eq 10.

$$\rho(\hat{\mathbf{p}}(\mathbf{y}), \hat{\mathbf{q}}) = \sum_{u=1}^{m} \sqrt{\hat{p}_u(\mathbf{y}).\hat{q}_u} \tag{10}$$

Weights for each pixel in the region of interest are calculated using the eq 11.

$$w_i = \sum_{u=1}^{m} \delta[b(\mathbf{x}_i) - u] \sqrt{\frac{\hat{q}_u}{\hat{p}_u(\hat{\mathbf{y}}_0)}} \tag{11}$$

6) Calculate the new location of target $\hat{\mathbf{y}}_1$ as shown in eq 12 .

$$\hat{\mathbf{y}}_1 = \frac{\sum_{i=1}^{n_h} \mathbf{x}_i w_i g(\| \frac{\hat{\mathbf{y}}_0 - \mathbf{x}_i}{h} \|^2)}{\sum_{i=1}^{n_h} w_i g(\| \frac{\hat{\mathbf{y}}_0 - \mathbf{x}_i}{h} \|^2)} \tag{12}$$

Again find the distance with this updated position as explained in step 5.

7) Compare the distance with predefined distance threshold. If the distance is less than the threshold distance, update this target position and move to next frame of video for target localization. Else, we have not found the correct position of object yet so repeat steps 3 to 6. Maximum number of iterations are set to 5.

Mean Shift can track the object even in presence of occlusion and clutter. However, it is not scale adaptive. Next, we will see KLT tracker which overcomes this drawback.

B. KLT Tracker

Lukas-Kanade algorithm is widely being used for various applications including object tracking, mosaicing, medical image registration etc. Later, Baker and Matthews suggested some improvements in original Lukas-Kanade algorithm to reduce computational complexity [12]. Modified KLT algorithm for object tracking is illustrated below.

The Goal of Lukas-Kanade algorithm is to align a template or reference image $T(\mathbf{x})$ to an input image I, where $\mathbf{x} = (x, y)^T$. Let $\mathbf{W}(\mathbf{x}; \mathbf{p})$ be the warping function where, $\mathbf{p} = (p_1, \ldots p_n)^T$ is a set of warping parameters. We have to estimate a transformation such that when it is applied to image I, it becomes very similar to template image pixel by pixel.

Mathematically, it is a problem to estimate warping parameters such that following is minimized:

$$error = \sum_{\mathbf{x}} [I(\mathbf{W}(\mathbf{x}; \mathbf{p} + \triangle \mathbf{p})) - T(\mathbf{x})]^2 \quad (13)$$

We assume that current estimate of $\mathbf{p}$ is known and then iteratively solve for increments to the parameters $\triangle p$. Next, we illustrate how these parameters are estimated.

1) Assume that we have some initial estimate of warping function parameters. Here, we have considered affine warping function. From earlier step of object recognition, homography estimated using RANSAC gives us initial set of warping parameters.

2) Subtract template from warped image. This is indicated in eq 14.

$$error = I(\mathbf{W}(\mathbf{x}; \mathbf{p})) - T(\mathbf{x}) \quad (14)$$

3) Compute the gradient of template image ∇T in x and y direction.

$$\nabla T = \left[T_x, T_y\right] \quad (15)$$

4) Obtain Jacobian of warping function $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$. Jacobian is nothing but derivative of a vector value function.

5) Compute steepest descent images by matrix multiplication of gradient of template image and Jacobian of warping function.

$$steepest \;\; descent = \nabla T \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \quad (16)$$

6) Compute Hessian matrix from steepest descent as shown in eq 17.

$$H = \sum_{\mathbf{x}} \left[\nabla T \frac{\partial \mathbf{W}}{\partial \mathbf{p}}\right]^T \left[\nabla T \frac{\partial \mathbf{W}}{\partial \mathbf{p}}\right] \quad (17)$$

Also, compute inverse Hessian matrix.

7) Multiply steepest descent obtained in eq 16 with error obtained in eq 14 as shown in eq 18. This process is called steepest descent parameter update.

$$update = \sum_{\mathbf{x}} \left[\nabla T \frac{\partial \mathbf{W}}{\partial \mathbf{p}}\right]^T [I(\mathbf{W}(\mathbf{x}; \mathbf{p})) - T(\mathbf{x})] \quad (18)$$

8) Calculate the change in parameters i.e. $\triangle$ $\mathbf{p}$ needed to minimise the error between template image and input image as shown in eq 19.

$$\triangle \mathbf{p} = H^{-1} \sum_{\mathbf{x}} \left[\nabla T \frac{\partial \mathbf{W}}{\partial \mathbf{p}}\right]^T [I(\mathbf{W}(\mathbf{x}; \mathbf{p})) - T(\mathbf{x})] \quad (19)$$

9) Now, as we have change in the parameters $\triangle \mathbf{p}$ needed, update the warping parameters as shown in eq 20.

$$\mathbf{p} \leftarrow \mathbf{p} + \triangle \mathbf{p} \quad (20)$$

Above steps are performed iteratively to estimate warping parameters of a single feature point. And the same steps are to be executed on each feature point in parallel for alignment estimation between a pair of images. To track a object in video which is nothing but sequence of images, alignment is to be estimated between every pair of frames. Some features may disappear over a period of time. So, features are detected after every 10 or 15 frames and alignment is estimated using new and old features.

IV. Results and Discussion

The proposed approach for automatic object tracking has been applied to the three different test sequences. The object of interest is automatically detected in the first frame of each sequence and then it is continuously tracked in the remaining part of sequence. This task of object recognition is implemented by exploring effectiveness of SIFT and RANSAC. Fig. 1 shows sample results of person recognition. Fig. 1(a) is used as training image for the person recognition

(a) (b) (c) (d) (e)

Fig. 1: Results of Person Recognition

in a set of images. We can see that person is correctly recognized when it is at different scale than the reference image as shown in fig. 1(b). Fig. 1(c) shows a case when there is change of pose. In fig. 1(d), projective transformation exists i.e. there is change in 3D viewpoint. Result of recognition in presence of partial occlusion is shown in fig. 1(e).

The same approach can be used for face recognition. Sample results of face recognition are shown in fig. 2. Training image

(a) (b) (c) (d) (e)

Fig. 2: Results of Face Recognition

of the person used for face recognition is shown in fig. 2(a). Fig. 2(b)-(e) show that face is correctly recognized irrespective of illumination changes that are present among the images as well as facial expression and pose variation. Even though part of face is covered by hand or goggle in fig. 2(d) and 2(e), face is successfully recognized.

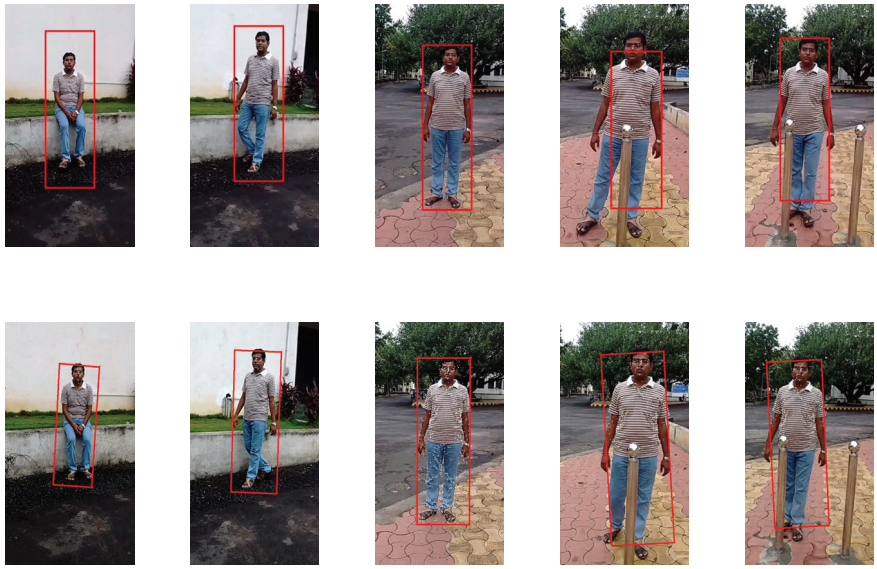

Fig. 3: Sequence 1: Tracking results of Mean Shift (1st row) and KLT Tracker (2nd row)

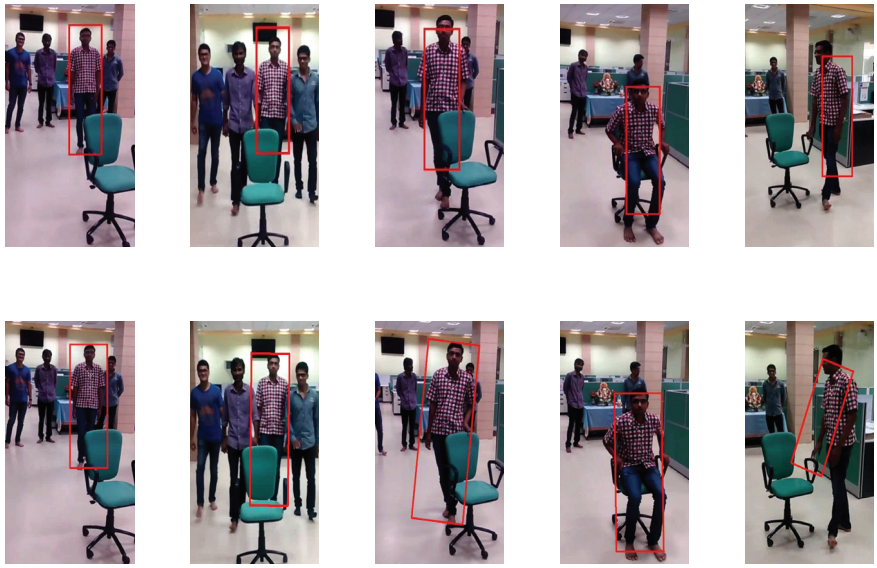

Fig. 4: Sequence 2: Tracking results of Mean Shift (1st row) and KLT Tracker (2nd row)

Thus region of interest can be efficiently and automatically detected in the first frame and it is tracked in subsequent frames by Mean Shift and KLT tracker. Results are tested on 3 frame sequences and their details are summarized in Table I.

TABLE I: Details of Three Image Sequences

Sequence	Size	Frame Number	fps	Object-Number
Single Person	864x480	523	25	1
Four Persons	864x480	541	25	4
Face	480x640	413	30	1

First of all, sequence Single Person is tested. In this sequence, person conducts several casual movements in outdoor environment. Results of this sequence are illustrated in fig. 3. First row shows outputs of Mean Shift tracker and second row shows outputs of KLT tracker. We can observe that person is being tracked when he is moving, handling the pose changes and partial occlusion.

Secondly, four person sequence is tested which consists of indoor environment. This sequence is a bit complicated scenario as an object's tracking is distracted and occluded by other objects. Here, the goal is to track the person wearing white shirt with red and black checks. Tracking multiple person is not our intention for now. Challenge in this case is to handle the occlusion by other objects and pose changes during tracking. Fig. 4 shows the tracking results for this sequence.

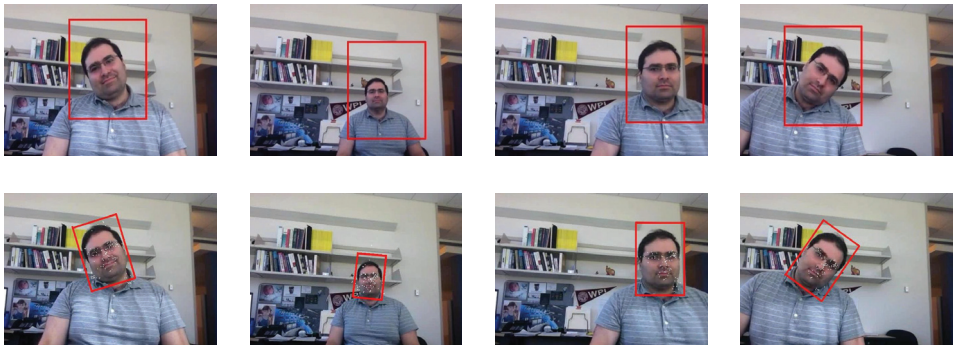

Fig. 5: Sequence 3: Tracking results of Mean Shift (1st row) and KLT Tracker (2nd row)

Next, algorithm is tested for face tracking on a video from MATLAB database and results are shown in fig. 5. We can see that face is tracked successfully when it is rotated to left or right as well as when person is moving from left to right. When person is moving back and forth, there is scale change of object but in spite of that face is recognized and tracked successfully.

From the results of above frame sequences, we can see that Mean Shift and KLT both have capability to handle partial occlusion. We also observe that Mean Shift is not scale adaptive i.e. size of rectangle remains constant. Also, Mean Shift tracking algorithm is based upon the colour information of the object. If the object of interest and background have similar colour, then Mean Shift may fail to identify and track true object. On the other hand, KLT can handle the scale changes of object. In other words, size of the rectangle and its orientation changes with change in scale and orientation of object. It can also track the object successfully even when object and background are of similar colour. But computational complexity of KLT is higher and hence its execution time is also higher than that of Mean Shift. These tracking algorithms earlier relied on manual selection of object which we have replaced with automatic selection by SIFT and RANSAC. However it is still an issue that real time performance has not been achieved.

V. Conclusion

In this paper, an automatic object recognition and tracking approach is presented and implemented. We have tried to deal with few challenges in this domain by exploring and combining effectiveness of SIFT and RANSAC with Mean Shift and KLT for tracking. Feature matching is done by cosine similarity approach as an alternative to euclidean distance which reduces computational complexity.

Object recognition results show that object can be successfully recognized in an image even if geometric transformations like rotation, scaling, translation, projective transformation and illumination differences are present. Same approach is also applied on face database for face recognition problem. Results are invariant to illumination changes and partially invariant to pose and facial expressions of the person.

This automatic object recognition approach is used to automatically detect the object in first frame of video and then it is tracked in subsequent frames. Object tracking is implemented by Mean Shift as well as KLT tracker and the results are compared. Mean shift is computationally less expensive but it is not scale adaptive. KLT is scale adaptive and is robust to noise and dynamic scene variations.

References

[1] Huiya Zhou, Yuan Yuan and Chunmei Shi, *Object tracking using SIFT features and mean shift*, Computer Vision and Image Understanding, vol. 113, pp.345-352, Mar. 2009.

[2] U. K. J. Himani, S. Parekh and Darshak G. Thakore, *A Survey on object detection and tracking methods*, International Journal of Innovative Research in Computer and Communication Engineering, Vol.2, pp. 2970-2978, Feb. 2014.

[3] Arnold W. M. Smeulders and Mubarak Shah, *Visual Tracking: An Experimental Survey*, IEEE Transactions on Pattern Analysis and Machine Intelligence, VOL. 36, NO. 7, JULY 2014.

[4] D. Comaniciu, V. Ramesh and P. Meer, *Real time tracking of non-rigid objects using mean shift*, pp. 142-49, 2000.

[5] B. D. Lukas and T. Kanade, *An iterative image registration technique with an application to stereo vision*, International Joint Conference on Artificial Intelligence, Volume 2, pp.674-679, 1981.

[6] Tobias Senst, Volker Eiselein and Thomas Sikora, *Robust Local Optical Flow for Feature Tracking*, IEEE Transactions on Circuits and Systems for Video Technology, VOL. 22, NO. 9, SEPTEMBER 2012.

[7] D. G Lowe, *Distinctive image features from scale-invariant keypoints*, International Journal of Computer Vision, VOL. 60, no. 2, pp. 91 to 110, November 2004.

[8] Wan-Lei Zhao and Chong-Wah Ngo, *Flip-Invariant SIFT for Copy and Object Detection*, IEEE Transactions on Image Processing, VOL. 22, NO. 3, MARCH 2013.

[9] D. G. Lowe, *Object Recognition from local scale invariant features*, International Conference on Computer Vision, Volume-2, pp.1150-1157, 1999.

[10] Bhakti Baheti, Ujjwal Baid and Sanjay Talbar, *A Novel Approach for Automatic Image Stitching of Spinal Cord MRI Images using SIFT*, International Conference on Pervasive Computing (ICPC), pp.15, 2015.

[11] M. Fischler and R. Bolles *Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography* , Communications of the ACM 24:381 to 395, 1981.

[12] S. Baker and I. Matthews, *Lukas - Kanade 20 years on : A unifying framework*, International Journal of Computer Vision, vol. 56, pp. 221-155, 2004.

[13] O. J. Alper Yilmaz and M. Shah, *Object Tracking: A Survey, ACM Computing Surveys*, vol. 38, pp.1-45, Dec. 2006.

[14] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2003.