International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-12, October 2019

# A Hybrid Surf-Based Tracking Algorithm with Online Template Generation

**Anshul Pareek, Nidhi Arora**

***Abstract:*** ***Visual tracking is the most challenging fields in the computer vision scope. Occlusion full or partial remains to be a big mile stone to achieve .This paper deals with occlusion along with illumination change, pose variation, scaling, and unexpected camera motion. This algorithm is interest point based using SURF as detector descriptor algorithm. SURF based Mean-Shift algorithm is combined with Lukas-Kanade tracker. This solves the problem of generation of online templates. These two trackers over the time rectify each other, avoiding any tracking failure. Also, Unscented Kalman Filter is used to predict the location of target if target comes under the influence of any of the above mentioned challenges. This combination makes the algorithm robust and useful when required for long tenure of tracking. This is proven by the results obtained through experiments conducted on various data sets.***

***Keywords: Visual tracking, SURF, Mean-Shift, Lukas Kanade method, Unscented Kalman Filter predictor, GrabCut, Online template generation and Updation.***

## I. INTRODUCTION

Applications like augmented reality, human computer interaction, video indexing and surveillance, dependable robotics etc.[1],[2], all have observed the dire need of tracking and interpretation of human motion. The ability to track any human in real time condition is of great significance in visual tracking. The physical challenges faced during the real time tracking are illumination change, camera motion, pose variation, scaling and last but not the least occlusion [3],[4].

To address the above defined challenges we have developed an algorithm which is implementable in the real time. Here the Lukas Kanade tracker (LK)[5] is merged with the SURF based Mean-Shift tracker (SBMS)[6]. This combination results in a robust algorithm. This hybrid multi tracker fulfills considerable purposes like providing robust tracking, successful long term tracking and solves the issue of online generation and its regular update. word) Also deals with out of plane rotation which remains to be a big issue in interest point based methods. This hybrid tracking gives better results as compared to individual LK and SBMS tracker. This combination rectify each other over failure situations.

**Revised Manuscript Received on October 30, 2019.**
***** Correspondence Author
**Anshul Pareek***, ECE Deptt, Maharaja Surajmal Institute Of Technology, Delhi India. Email: er.anshulpareek@gmail.com
**Dr. Nidhi Arora**, CSE Department, G.D. Goenka University, Gurugram, India. Email: nidhi.arora1@gdgoenka.ac.in



Same way, if pose variation occurs and SBMS fails to converge, LK comes to rescue by localizing target region. SBMS defines a template pool of intermittently occurring SURF descriptors but the biggest task is to keep away the background descriptors from the template pool, else it results in model drifting. To overcome this problem, region growing approach is applied to the segment [7]. In our case GrabCut [8] is used for image segmentation of fore ground from background. This is applied to the SURF descriptors lying outside the target polygon, they are treated as background descriptors and washed away. The primary goal of the work presented here is to amplify the robustness of the human tracking algorithm based on interest point method [9]. It is conventional that adequate number of descriptors are obtained, for that it is expected for the target to be colossal enough. So, the work done presented may not be suitable for surveillance as the target appearance is very small [1],[2]. It would be preferable to be used for human following robots and its robotic operations.

The paper further is categorized into number of segments. Problem statement is stated in next section. In Section III tracking algorithm is explained in detail. Experimental set up is shared in Section IV. All the experimental results are stated followed by comparative analysis with existing algorithm in Section V. Conclusion is delivered in Section VI.

## II. PROBLEM STATEMENT

Assume a set of frames $F_i$, where i = 0, 1, 2….N of the video sequence. For the first frame $F_0$ a rectangular box $B_0$ is drawn over the target region to be detected. To compute SURF descriptors in box $B_i$, set of SURF descriptors of frame $F_i$ is stated as

$$L(B_i) = \{(k_1, l_1, w_1), (k_2, l_2, w_2) \ldots\ldots (k_n, l_n, w_n)\} \quad (1)$$

Where $x_i$ is the feature point location in 2D of 64-dimensional SURF features $L_i$. $w_i$ are the weights allocated to SURF descriptors $L_i$ .$l_k$ is a set of feature point locations in a given frame within a window and $l_y$ is the corresponding set of descriptors to $l_k$. $B_s$ and $B_t$ are source and target window. Now the set of good matching points between Bs and $B_t$ are computed and they are

$$L(B_s \sim B_t) = \{(k_1, l_1, w_1)^s, (k_2, l_2, w_2)^s \ldots\ldots (k_m, l_m, w_m)^s, (k_1, l_1, w_1)^t, (k_2, l_2, w_2)^t \ldots\ldots (k_m, l_m, w_m)^t\} \quad (2)$$

Where $(k_m, l_m, w_m)^t$ are SURF descriptors for target window and $(k_m, l_m, w_m)^s$ are SURF descriptors for source window .

The tracking window B has parameters centre (c), width (w) and height (h) and is represented as

$$B=(c, w, h) \quad (3)$$

Now our target is to find tracking window $B_i = (c_i, w_i, h_i)$ for all the given frames. In this rectangular window, both foreground and background SURF descriptors are present which may lead to false tracking. In order to avoid it an elliptical region [E0] is drawn which fits inside rectangular window removing background descriptors. The set of such descriptors in elliptical region in the first frame is given by

$$L(E_0) = \{(k_i, l_i, w_i) \mid (k_i, l_i, w_i) \in L(Bn) \wedge k_i \in E_0\} \quad (4)$$

Where $w_i=20$ and i = 1,2,3...m. This is how the segmentation of the foreground from background is done. From here object template can be quoted as

$$O_t = \{(k_i, l_i, w_i) \mid (k_i, l_i, w_i) \in L(B_n) \wedge k_i \in E_0 \wedge k_n \notin BG_R\} \quad (5)$$

$BG_R$ is the segmented background region. $O_t$ initializes the object model $O_m$. The ultimate goal is to make object model which is explained in detail in tracking algorithm section.

## III. TRACKING ALGORITHM

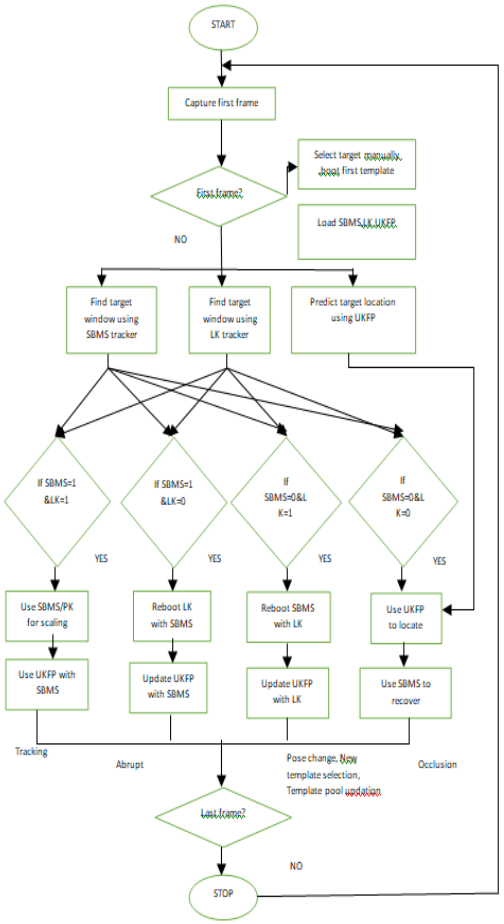


**Fig 1: Tracking Flowchart**

- The flow chart mainly works on two primary algorithms discussed previously. The first one is SBMS. Here the SURF features are combined with Mean-Shift tracker which basically is based on color Histogram matching. One can directly use SURF matching amid successive images but due to build in noise which results in fading away of descriptors in the course of matching. This makes the entire process computationally expensive and system becomes prone to failure in long run [10]. In SBMS, Histograms of source and target windows are generated. The K-means clusters are created only once for the source window and not for every frame, this reduces the computational time [11]. This is similar to bag of words approach [12] but difference is no offline database is created instead the most stable descriptor points are reprojected to provide a decent number of feature points over the entire tracking[12]. RANSAC based Homography [13], [14] is used to remove outliers. Mean-Shift computes the new centre[6].
- Second important algorithm used here is Lucas-Kanade technique. It is explained as movement of objects between successive frames in a sequence, which occurs because of relative movement that pops up between target and camera [5]. Lucas-Kanade came up with a proposal to efficiently estimate the motion of feature keypoints by comparing successive frames.
- As seen in the flowchart of proposed algorithm in Fig 1, the first case is where SBMS and LK are equal to 1, the corresponding action is tracking. In simpler words this is the condition when both the trackers are able to detect the target. This tracking by detection framework. Any target detection is said to be successful when SURF descriptor matching between target window and template is above a threshold defined by user. In this case the value of threshold is kept 0.2. Since both trackers are able to detect the target, we use matching index as benchmark to identify the higher quality window. Matching index is defined as a ratio tracking window SURF descriptor (which match with template pool) to that of the total number descriptors available in the window, with added weights of window obtained between the centres of two successive windows. The one with higher matching index gets the power of defining the scaling and repositioning of final tracking window.
- The second case is when SBMS =1 and LK=0. This accounts for abrupt motion of camera or the falling rate of frame .In this reduces [5] which ommonly observed in LK method that matching points keep on decreasing over the course of time. So a user defined threshold is declared, if matching points go below this level the LK tracker of reloaded and this happens whenever there is instant change in tracking window. In this case SBMS initializes the LK tracker by providing its current SURF feature points which acts as input estimating the location of these feature points incoming frame.When SBMS=0 and LK=1, this is the case of pose change. If pose change is detected a new template is selected.

When SBMS tracker fails to detect the target it is because when pose change occurs a large number of matching descriptors fades away but LK tracker has decent number of matches which estimates the location of feature points in next available frame This makes tracking successful under pose change even. But not all the templates provided by LK tracker are suitable to add to the template pool. For checking the quality a SURF correspondence based on Euclidean distance between descriptors of two sets is computed. Also matching index should be below the user defined threshold and scaling of bounded box should be less than 20%. Any scaling beyond this is includes background descriptors in order to achieve matching points. If the template meets these conditions then only it is included in template pool.

- The last case is when SBMS and LK both are equal to 0. This gives clear indication of an occlusion encountered. Here, the role of motion predictor steps in. In this case we are using Unscented Kalman filter (UKFP)[19]. Occlusion could be full or partial. It's easy to recover tracking in partial occlusion, because few descriptors from the template pool are recovered. But in case of complete occlusion both trackers fail due to unavailability of matching points. So, UKFP estimates a tentative window. When one such window is obtained, target is locally searched in next frame. But target could be occluded in couple of frames, then there is a possibility of model drifting. To deal with it tracking windows are expanded and local search is conducted. Lastly direct feature matching is conducted to locate target. It is always beneficial to expand the tracking window to locate target than frame to frame matching, because it contributes to low computational time.

## IV. TEST BED

The set up used for implementation requires simulation in C++using OPENCV 3.4.0. LINUX is the operating system used on i5 processor. Four datasets are chosen to show experimental results which includes all the visual tracking challenges like pose change, occlusion, abrupt motion etc. Two real-time videos are picked from Youtube (YT-1 and YT-2) and two videos belong to our data sets (PP-1 and PP-2). All videos are of 640X480 resolution.

## V. EXPERIMENTAL ANALYSIS

Here we discuss the results obtained .We have applied the algorithm on four datasets. The first step is to manually select the target in a given frame and bounded in a rectangular box. An ellipse is drawn over target and well fitted inside rectangular box to remove background descriptors. The entire body is divided into three segments of head, torso and legs still few background descriptors are a part of foreground ones. In order to remove the leftovers we have applied GrabCut which is a region growing approach. Later on this segmented image the SURF descriptors are computed. This is clearly shown in Fig 2

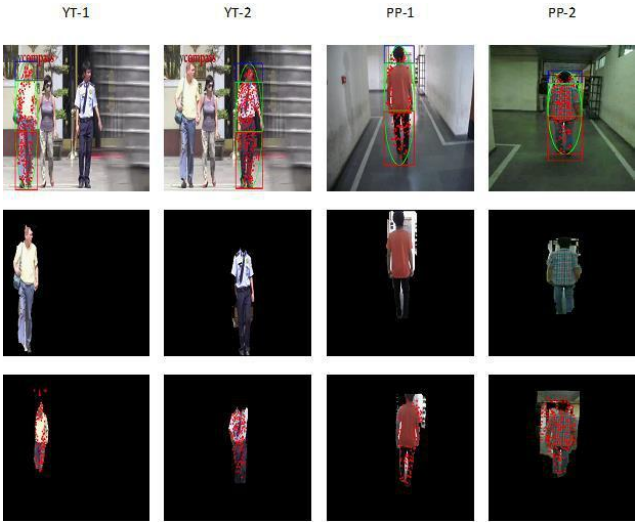


**Fig 2: The first horizontal set shows the target fitted into ellipse and divided in three portions offff head , torso and legs. The second set is obtained by GrabCut image segmentation and in third set SURF descriptor are computed over segmented image.**

Fig 3 shows presents the templates obtainedfor each dataset. The templates shows the various poses. The scaling effect is prominently seen. The occlusion templates are discarded as they could drift the object model.

**Fig 3:Template selection for each database.**

Fig 4 shows how algorithm recovers from occlusion. White tracking window is predicted location UKFP. The blue tracking window emerges from re-initialization of LK tracker. Tracking results are shown in Fig 5. Tracking video is available online for verification [15].

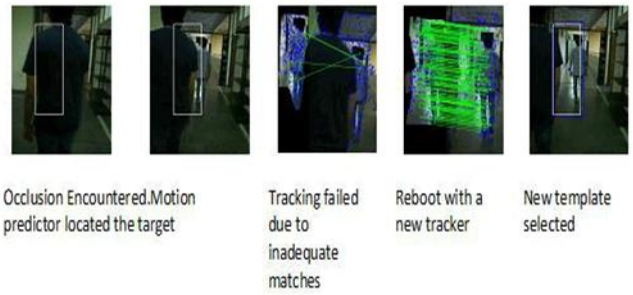


**Fig 4: Occlusion recovery steps**


**Fig 5: Tracking snapshots.White, blue and green tracking windows are for UKFP, LK and SBMS trackers.**

**TABLE-I**

| Dataset | Total no. Of frames | Camera motion and Pose change | Scaling upto | No. of descriptors | No. of templates generated | Average time (ms) | Number of Occlussions detected |
|---|---|---|---|---|---|---|---|
| YT-1 | 260 | Smooth, Yes | 160% | 180-550 | 9 | 202 | 8 |
| YT-2 | 260 | Smooth, Yes | 165% | 150-500 | 11 | 431 | 6 |
| PP-1 | 368 | Abrupt, No | 220% | 160-600 | 19 | 199 | 9 |
| PP-2 | 368 | Abrupt, No | 180% | 180-300 | 10 | 168 | 5 |

Table I presents overall performance for the various datasets. Robustness of algorithm is reflected in the results against the occlusion, abrupt camera motion and pose change. The maximum descriptor range is 600 which is quite less than obtained in other algorithms, leading to less memory requirement in order to store templates.

**TABLE-II**

| DataSet | Para-meters | Algorithms Comparative Analysis | | |
|---|---|---|---|---|
| | | Reprojection based MeanShift | RF-Mean- Shift based object model | Proposed algorithm |
| PP-1 | SR | 45.64 | 16.20 | 86.23 |
| | AOL | 49.64% | 25.23% | 67.24% |
| | AT | 121 ms | 1020ms | 199 ms |
| PP-2 | SR | 64.24 | 30.58 | 83.74 |
| | AOL | 61.9% | 34.34% | 63.2% |
| | AT | 130 ms | 900ms | 168 ms |

To prove the efficiency of the proposed algorithm, a transparent comparison is done with other existing algorithms. Here, we compare our proposed algorithm with reprojection based Mean-Shift-SURF algorithm [6] and SURF Mean-Shift based object model [16]. The comparison is done using Success Rate, Computational time and percentage of overlap. Results are displayed in Table II .Computational time is average time taken to process. The other important parameter to evaluate efficiency of any tracking algorithm is the overlap percentage. To obtain this the ground truth is manually computed for each frame. The covered region is then compared with tracking window. The percentage area which is common is termed as overlap percentage. When the common area is greater than 50 %, successful tracking is attained else it is considered to be failed. Success rate is obtained from this overlap percentage [17].Mathematically it is presented as

$$SR = (n/N) \text{ X } 100 \quad (6)$$

Where n is the total number of successfully tracked frame and N is of the total number of frame. The results shown in Table II are only for two data sets. Highest success rate is achieved by the proposed algorithm. The minimum time is taken by Mean-Shift but it cannot deal with pose change and occlusion. The average time is less than 200 ms and that makes it useful in real time tracking.

## VI. CONCLUSION

The proposed algorithm is interest point based challenges like pose change, illumination change, scaling, occlusion etc. The proposed hybrid algorithm is able to overcome all these challenges .It also overcomes the shortcomings of fixed template-based tracking by offering online generation and update of templates. The average time for computation is less than 200 ms which makes it adequate for real time implementation without compromising over success rate. Also, SURF is the only visual feature used no other features like blob, edges or color is required [20].

## REFERENCES


1. Zhigang Bing, Yongxia Wang, Jinsheng Hou, Hailong Lu, and Hongda
2. Chen " Research of tracking robot based on surf features". International
3. Conference on Natural Computation (ICNC) IEEE Yantai Shangdong(2010) 3523–3527.
4. W. Hu, X. Zhou, W. Li, W. Luo, X. Zhang, S. Maybank, "Active contour-based visual tracking by integrating colors shapes and motions",(2013) IEEE Trans. Image Process., vol. 22, no. 5, pp. 1778-1792. K. Rasool Reddy, K. Hari Priya, N. Neelima "Object detection and
5. tracking: A survey". International Conference on Computational Intelligence and Communication Networks (CICN)(2015).
6. Alper Yilmaz, Omar Javed, and Mubarak Shah "Object tracking: A survey," ACM Computing Surveys (CSUR)(2006) 38(4): Article 13.
7. B. D. Lucas and T. Kanade (1981), An iterative image registration technique with an application to stereo vision. Proceedings of Imaging Understanding Workshop, pages 121--130(1981)
8. Sourav Garg and Swagat Kumar, "Mean-shift based object tracking algorithm using surf features," in Recent Advances in Circuits, Communications and Signal Processing. 2013, pp. 187–194, WSEAS.
9. Asano, T. and N. Yokoya (1981). Image segmentation schema for low-level computer vision. Pattern Recognition 14 (1-6), 267-273.
10. C. Rother, V. Kolmogorov, and A. Blake, GrabCut: Interactive foreground extraction using iterated graph cuts, ACM Trans. Graph., vol. 23, pp. 309–314, 2004
11. Schmid, Cordelia; Mohr, Roger; Bauckhage, Christian (1 January 2000). "Evaluation of Interest Point Detectors" (PDF). International Journal of Computer Vision. 37 (2): 151–172. doi:10.1023/A:1008199403446
12. H. Bay et al(2008) "Speeded-up robust features (SURF) ." Computer Vision and Image Understanding, 110(3): 346-359, 2008
13. C. Aggarwal, J. Han, J. Wang, and P. S. Yu. A framework for clustering evolving data streams. In Proceedings of the International Conference on Very Large Data Bases, pages 852–863, 2003
14. Sivic, Josef (April 2009). "Efficient visual search of videos cast as text retrieval" (PDF). IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 31, NO. 4. IEEE. pp. 591–605.
15. Martin A. Fischler & Robert C. Bolles (June 1981). "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography" (PDF). Comm. ACM. 24 (6): 381–395. dio:10.1145/358669.358692.
16. Pareek Anshul and Arora Nidhi (2018), "Evaluation of Feature Detector-Descriptor Using Ransac for Visual Tracking " International Conference on Sustainable Computing in Science, Technology and
17. Management (SUSCOM-2019). Available at SSRN: https://ssrn.com/abstract=3354470.[15]Human tracking-Anshul Pareek https://youtu.be/GhQ8cjEuzcQ
18. Meenakshi Gupta, Sourav Garg, Swagat Kumar, and LaxmidharBehera, "An online visual human tracking algorithm using surf-based dynamic object model," in International Conference on Image Processing (ICIP), Australia, 2013, IEEE.
19. F. Bashir, F. Porikli. "Performance evaluation of object detection and tracking systems", IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS), June 2006.
20. T. Song, J. Speyer, "The modified gain extended Kalman filter and parameter identification in linear systems", Automatica, vol. 22, pp. 1, 1986.
21. WAN Li, LIU Yan-chun, PI Yi-ming, "Comparing of Target-Tracking Performances of EKF,UKF and PF" RADAR Science and Technology,2007,vol 1
22. US 2009238460, Ryuji Funayama, Hiromichi Yanagihara, Luc Van Gool, Tinne Tuytelaars, Herbert Bay, "ROBUST INTEREST POINT DETECTOR AND DESCRIPTOR", published 2009-09-24


## AUTHORS PROFILE

**Ms. Anshul Pareek**, is B.E. in Electronics and Communication from University of Rajasthan, and M.Tech. in Digital Communication from Rajasthan Technical university. Currently working as an Assistant Professor in Maharaja Surajmal Institute of Technology, New Delhi. Her research interests in fields of Artificial intelligence, Machine learning are Human Computer interaction, mainly Human motion tracking.

**Dr Nidhi R. Arora,** holds a PhD in the field of Information Retrieval from INHA University South Korea. Her dissertation work focused on designing a ranking algorithm to produce top-k search results for keyword query on data graphs. She is currently working as Associate Professor in GD Goenka University. Her research interests are in the field of Deep Learning, Natural Language Processing and Machine Learning. She has publications in top conferences and journals such as DEXA, DASFAA, Expert Systems With Applications (ESWA) and New Generation Computing.