Robust Object Tracker in Video via Discriminative Model

Mehrez ABDELLAOUI[1,2]*, Ali DOUIK[2]

[1,2] Higher Institute of Applied Sciences and Technology, University of Kairouan,

Avenue Beit El Hikma, Kairouan, 3100, Tunisia

mehrez.abdellaoui@enim.rnu.tn (*Corresponding author*)

[2] NOCCS Laboratory, National Engineering School of Sousse, University of Sousse,

Pôle technologique de Sousse, Route de Ceinture Sahloul, Sousse, 4054, Tunisia

ali.douik@enim.rnu.tn

Abstract: In this paper a new method for object tracking in complex video scenes is presented. Video object tracking is one of the most challenging tasks in the field of image and video processing. Due to the complexity and the variability of the scenes and objects, there are a lot of challenges one faces. Recently, many tracking approaches have used discriminative models to create robust trackers of specific pixels in video frames called interest points. These trackers allow users to extract the target object from the complex background of the scene in real time. Most of the proposed robust trackers are facing bad or wrong predictions of the interest points positions related to the target object. This greatly decreases the performances of the tracker. To overcome this problem, both the interest point detection methods and online boosting discriminative model were combined in order to obtain a robust tracker. The proposed approach was experimented on video clips presenting several aspects of intrinsic and extrinsic factors of real challenges that tracking algorithms may face. The obtained results using two evaluation metrics have shown that the present robust tracker outperforms the recent state-of-the-art techniques in terms of Spatial Robustness Evaluation (SRE) and Temporal Robustness Evaluation (TRE).

Keywords: Object tracking, Discriminative model, Interest points detection, SRE, TRE.

1. Introduction

Video processing and understanding is one of the most active research topics in the field of image processing and computer vision. Target object tracking in video is one of the major tasks in video analysis. It remains an open and challenging research problem even after several years of research in this field. There are several applications of target object tracking including human activity recognition (Alahi et. al, 2017), visual robot control (Ak et. al, 2018), vehicle navigation (Lee et. Al, 2012), event detection and video retrieval (Liu et. al, 2008), face detection (Vrejoiu, 2017). Great challenges still exist in increasing the robustness of the proposed trackers especially when both extrinsic and intrinsic factors producing modifications of the target object appearance exist. Extrinsic factors are illumination changes, partial and full occlusions, dynamic background clutter and motion blur, whereas, common intrinsic factors include pose, appearance and scale variations.

The proposed method aims to track an arbitrary object for which only its initial position in the first frame is known. The target present in this paper is represented by the development of a robust method of updating the position of an object based on an adaptive appearance model which is able to manipulate important intrinsic and extrinsic factors with acceptable deviation of the target. The proposed tracker uses the interest point (IP) flexibility and the discriminative learning paradigm robustness. It represents the target object by using a set of interest points described using Haar like features ([17], [6]) and extracted from small image patches surrounding it. To localize correctly the target object in any frame in the sequence, IPs are-obtained-from a predefined-search patch in the current frame.

The rest of the paper is organized as follows. In section 2, related work to robust trackers is presented. The section 3 is dedicated to briefly describe the adaptive appearance models based on object tracking. In Section 4, the tracking algorithm is introduced. The experimental results of the proposed method on a set of challenging video sequences are exposed in section 5. Finally, a conclusion is given in Section 6.

2. Related Works

Many online tracking methods have been developed by making use of generative or discriminative models. Generative tracking methods typically search the target object with minimal reconstruction errors in the image region by modelling it. Black et al. [9] developed a

method using an off-line subspace learning model to characterize the target object. The mean shift method realizes the object tracking in two steps [13]. The first one consists of using nonparametric distributions features for object description such as color distribution, whereas the second is the location object search using mean shift method. To be more robust, a Continuous Adaptive Mean Shift (CAMS) [10] can overcome the problem of color distribution changing by varying the size of the search window and updating the color distributions at each frame. The Frag tracker [1] uses histograms of local patches to model appearance of the object. It overcomes several difficulties, which is lost in traditional histogram-based algorithms, (e.g. mean shift). First, by robustly combining multiple patch votes, partially occluded objects or pose changes are handled. Second, the color spatial distribution of pixels which is lost in traditional histogram-based algorithms is considered due to the geometric relations between patches. The Incremental Visual Tracker IVT [25] adapts appearance variations using an incremental subspace model. It learns incrementally a low dimensional subspace representation, and efficiently adapts online the appearance changes in the target object. Kwon et al. [19] use sparse principal component analysis (SPCA) of a template feature set to construct multiple observation models. Each observation model represents a particular object's appearance. Most of the described methods use holistic representations for the description of the objects and therefore do not manipulate occlusions correctly. Also, their performances decrease when the background is cluttered.

Discriminative techniques formulate the tracking problem as a binary classification problem in order to separate the target object from the background. This separation was done by Grabner et al. [17] and Babenko et al. [6] using online learning. On the other hand, Avidan [5], Collins et Liu [12] and Grabner et al, [16] describe the target object using a set of features. To develop a robust tracking method, Avidan [5] combines a set of weak classifiers to obtain a strong one. In [12], Collins et al. separate the target object from the background by an online discriminative features learning. Grabner et al. [16] introduce an online

boosting algorithm to update discriminative features. Babenko et al. [6] consider examples within positive and negative bags by developing Multiple Instance Learning "MIL" into online object tracking.

As it is commonly known, an ideal tracking process depends on three elements: the first one is an appearance model, which can refine the tracking of objects by finding the most likelihood location of the appearance model. The second one is a motion model, that links the locations of the object over time, and the final one is a search strategy to get the most probable location in the actual frame.

3. Object Tracking Based on Adaptive Appearance Models

Adaptive appearance modelling methods have been developed to effectively address actual challenges of object tracking. These approaches are based on the appearance models structure. They only model the target [25] or both the target and the background ([4], [21], [22], [28]). Better results were obtained using these methods that isolate the object from the background through a discriminative classifier.

These methods extract the neighbourhood around the tracker location for negatives samples and consider the current location as one positive sample to update the classifier. Bad prediction of object location can decrease progressively the model update performances and cause false detection. Further, sampling multiple examples from a small neighbourhood around the current tracker location as positive samples leads to confusion in the appearance model and deterioration in its discriminative strength.

To avoid confusion that causes problems for supervised classic learning approaches, Babenko et al [6] involve Multiple Instance Learning (MIL) [14] method for object tracking, analogous to Viola et al. [26] for object detection. In MIL, the form of training data is $\{(X_1, y_1), \ldots, (X_n, y_n)\}$ where a bag $X_i = \{x_{i1}, \ldots, x_{im}\}$ and y_i is a binary bag vector. The bag vectors are set using the following measure (1).

$$y_i = \max_j \left(y_{i,j} \right) \qquad (1)$$

where $y_{i,j}$ are the instance labels. In other words, a bag is considered as positive one if it includes at least one positive instance. To train a boosting classifier that enlarges the log likelihood of bags as mentioned in (2), the MIL Boost approach based on the gradient boosting framework is used.

$$S = \sum_i (\log(p(y_i | X_i))) \qquad (2)$$

The following terms used in on-line boosting algorithm are briefly defined:

Weak classifier: A weak classifier achieves slightly better results than random guessing. The hypothesis f^{weak} is a feature generated by applying a defined learning algorithm.

Selector: A selector pick out exactly one of M weak classifiers with hypothesis: $F^{weak} = \{f_1^{weak}, .., f_M^{weak}\}$

$$f^{sel}(x) = f_n^{weak}(x) \qquad (3)$$

where n is selected depending on the estimated error e_j of each weak classifier $f_j^{weak} \in F^{weak}$ such that $n = \arg\{\min_j (e_j)\}$.

Strong classifier: A strong classifier is obtained by a linear combination of selectors as mentioned in (4) and (5)

$$F^{strong}(x) = sign(conf(x)) \qquad (4)$$

$$conf(x) = \sum_{n=1}^{N} \alpha_n . f_n^{weak}(x) \qquad (5)$$

4. Proposed Tracker

The proposed tracker is developed as a detection method. Two steps are used for this detection. The first one is feature detection and the second one is the online boosting.

Detected features are usually related to locations of IPs which are particular image pixels. These pixels are less sensitive to several deformations and various noises than other pixels of the image.

Online boosting uses the tracking window in the current frame. Positive patches which are close to the current target object and the negative patches which are far away from the object center are extracted to update the classifier result.

To estimate the object's location in the next frame, small patches [5x5] are extracted around these points and the Haar like features [26] is computed within these patches to generate weak hypotheses. The detected patch is the one with the best classification score. Combining both online boosting and feature detection, a robust tracking algorithm was obtained.

4.1 Feature Detection

To detect features that are used in the proposed online boosting, the Noble corner detector was applied based on background suppression and imposing spatial constraints [15].

Complex scenes are characterized with highly textured background. That led to undesirable space interest points. In order to get only the efficient set of interest points, the surround suppression mask (SSM) is applied on each pixel, and then the effect of its neighbourhoods is computed.

After that, an inhibition term is calculated for each point Cσ(Ix,Iy) according to equation (6), where w and z are defined as the horizontal and vertical gradient of the SSM. $\Theta_\sigma(Ix)$ and $\Theta_\sigma(Ix_{w,z})$ are the gradients at $X \equiv (I_x, I_y)$ and $X_{w,z} \equiv (I_x - w, I_y - z)$ points respectively. σ is the scale level of the gradient function.

$$\Delta_{\Theta,\sigma}(X, X_{w,z}) = \left| \cos\left(\Theta_\sigma(X) - \Theta_\sigma(X_{w,z})\right) \right| \qquad (6)$$

The relation (7) is used to control the strength of the surround suppression. SIPs number relies on the rate α. Then, non-maxima suppression is used to get the final group of interest points from the Cα, σ.

$$C_{\alpha,\sigma}(X) = E\left(C_\alpha(X) - \alpha \times l_\alpha(X)\right) \qquad (7)$$

$$E(k) = \begin{cases} k & if \ k \geq 0 \\ 0 & if \ k < 0 \end{cases} \qquad (8)$$

$$l_a(X) = \iint_\Omega C_\sigma(X_{w,z}) \times \Delta_{\Theta,\sigma}(X, X_{w,z}) \, dw \, dz \qquad (9)$$

where l_α is a suppression term and Ω is the image domain.

As mentioned in [11], after suppressing background interest points, the local constraint is applied using the surround suppression mask and non-maxima suppression.

4.2 Features Description

After interest points detection, small patches [5x5] are extracted around these points and the Haar like feature [26] is computed within these patches to generate weak hypotheses. The feature value is then a weighted sum of the pixels, an integral image is used in order to make the computation of Haar like feature easy and efficient.

4.3 Classifier Update

Each weak classifier, used in the present approach, contains a set of Haar like features $f=(f_1, f_2, ..., f_n)$ computed around interest points and four parameters $(\mu_1, \sigma_1, \mu_0, \sigma_0)$ representing the average and the standard deviation. The feature proposed in this paper is modelled as mentioned in [6].

$$GC_k(x) = \log\left[\frac{\prod_{i=1}^{n} p(f_i|y=1) \, p(y=1)}{\prod_{i=1}^{n} p(f_i|y=0) \, p(y=0)}\right]$$
$$= \sum_{i=1}^{n} \log\left(\frac{p(f_i|y=1)}{p(f_i|y=0)}\right) \quad (10)$$

The uniform prior $p(y=1) = p(y=0)$ is considered, where y is a binary parameter. Receiving new input, the weak classifier is updated using these two measures (11) and (12).

$$\mu_1 \leftarrow \gamma\mu_1 + (1-\gamma)\frac{1}{n}\sum_{i|y_i=1} f_k(x_i) \quad (11)$$

$$\sigma_1 \leftarrow \gamma\sigma_1 + (1-\gamma)\sqrt{\frac{1}{n}\sum_{i|y_i=1}(f_k(x_i) - \mu_1)^2} \quad (12)$$

where $0 < \gamma < 1$ is a learning parameter. The update rules for μ_0 and σ_0 are similarly defined.

The proposed tracking algorithm is described below. In addition, Figure 1 illustrates the framework of the proposed tracker.
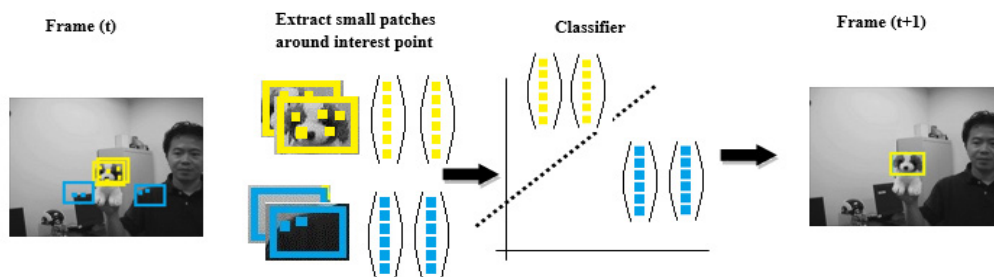
Algorithm 1 : Proposed Tracking Algorithm
1. Input: Initialize tracking window location
2. Extract a set of image samples : $$E^{\tau} = \{k \; \|l(k)-l_{t-1}\| < \tau\}$$ where l_{t-1} is the object location at the frame number (t-1).
3. Get the interest points found within object window using improved selective interest point detector in order to extract the features.
4. Locate the target object which is based on local maximal classifier response obtained from GC classifier to each feature vector f
5. Extract two sets of image samples $$P_\alpha = \{k| \; \|l(k)-l_t\| < \alpha \} \text{ and}$$ $$P_{\zeta,\beta} = \{k| \zeta < \|l(k)-l_t\| < \beta\} \text{ with } \alpha < \zeta < \beta.$$
6. Pick out the features with the obtained two sets of patches and update the classifier criterion according to (11) and (12).

5. Experiments

5.1 Experimental Setup

The proposed approach is implemented using MATLAB. It is evaluated on various publicly available video sequences relative to diverse challenges such as: pose, scale and illumination, occlusion, background clutters and motion blur. Table 1 illustrates the sequences used for evaluation and the challenges of every one.

To extract the positive patches, a search radius is defined with a value $\alpha = 4$. The internal and external radius of the set $X_{\zeta,\beta}$ that produces negative patches are fixed to $\zeta = 8$ and $\beta = 30$. To detect the location of target object, the value of the search radius is fixed at $\tau = 10$. γ is the learning parameter and is set to 0.85 and the projected space dimension is set to n=50. To detect IPs, an improved selective corner detector is used.

Figure 1. Framework of the proposed approach (image sample taken from TB 50 dataset)

Since the number of IPs are normally proportional to the target object size, the IP detector parameters is set with a standard deviation value σ = 0.5 and a threshold th = 1 depending exclusively on the size of the target object.

5.2 Evaluation Criteria

In order to evaluate the proposed approach accurately in opposition to state-of-the-art recent proposed approaches, the success and precision rates are used as quantitative results. In addition, the tracking algorithms robustness is also evaluated.

5.2.1 Precision Plot

The center location error is one of the most used evaluation metrics to quantify the tracking precision. It is computed as the average Euclidean distance between the true position of the center locations and the position estimated by the tracking algorithm. The overall precision is obtained by measuring the average center location error for all the frames of one sequence. However, this error measure may not measure the tracking precision correctly when the tracker loses the target and the output location can be random [7]. To be always significant, the precision plot is measured only for the estimated locations that aren't far from the true center location with a threshold = 20 pixels ([7], [18]).

5.2.2 Success Plot

Another evaluation metric is computed by measuring the overlap score between the estimated tracking bounding box and the ground truth one. Considering Bt as the resulting tracking bounding box and Bg as the ground truth bounding box, the overlap score OS is computed according to equation 13.

$$OS = \frac{\left| B_t \cap B_g \right|}{\left| B_t \cup B_g \right|} \qquad (13)$$

where $\cap$ and $\cup$ are the intersection and union of two bounding boxes respectively, and $|\cdot|$ denotes the number of pixels in each region. OS is then compared to a threshold varying between 0 and 1 and the number of successful frames is computed

when the OS is higher than the threshold. It is unfair to represent the tracker performance correctly, only by using the success rate value. Thus, the Area Under Curve (AUC) is computed for each success plot and then the tracking algorithms can be easily ranked according to AUC values.

5.2.3 Robustness Evaluation

Wu et al. proposed in [26] a new measurement for robustness evaluation. These measures are referred as temporal robustness evaluation (**TRE**) and spatial robustness evaluation (**SRE**) respectively. Previously the tracking robustness was tested according to a one-pass evaluation (**OPE**) with only one initialization position from the ground truth. TRE is obtained by varying the initialization temporally and SRE is obtained by varying the initialization spatially.

5.3 Benchmark Dataset

To appraise the proposed approach, the benchmark dataset TB 50 [29] has been utilized. The evaluation metrics presented in the previous subsection have been used in order to compare the algorithm proposed in this paper to other trackers. This dataset contains 50 video sequences. Each video sequence is annotated with one or more various attributes for better evaluation of the trackers robustness. The proposed tracker has been compared with model update trackers (Frag [1], MIL [6], OAB [16], SemiBoost [17]).

From the benchmark dataset, 8 video sequences presenting different types of challenges has been chosen.

Tiger1: This video contains many challenges such as illumination variation, occlusion and out of plane rotations. The proposed approach obtains the best results as presented in Figure 2 (a).

Coke: The coke sequence presents a brilliant object, which involves some difficulty. In Figure 2 (b), the proposed algorithm tracks correctly the brilliant object compared the other methods.

Crossing: This video includes the following changes: Out-of-Plane Rotation Scale Variation
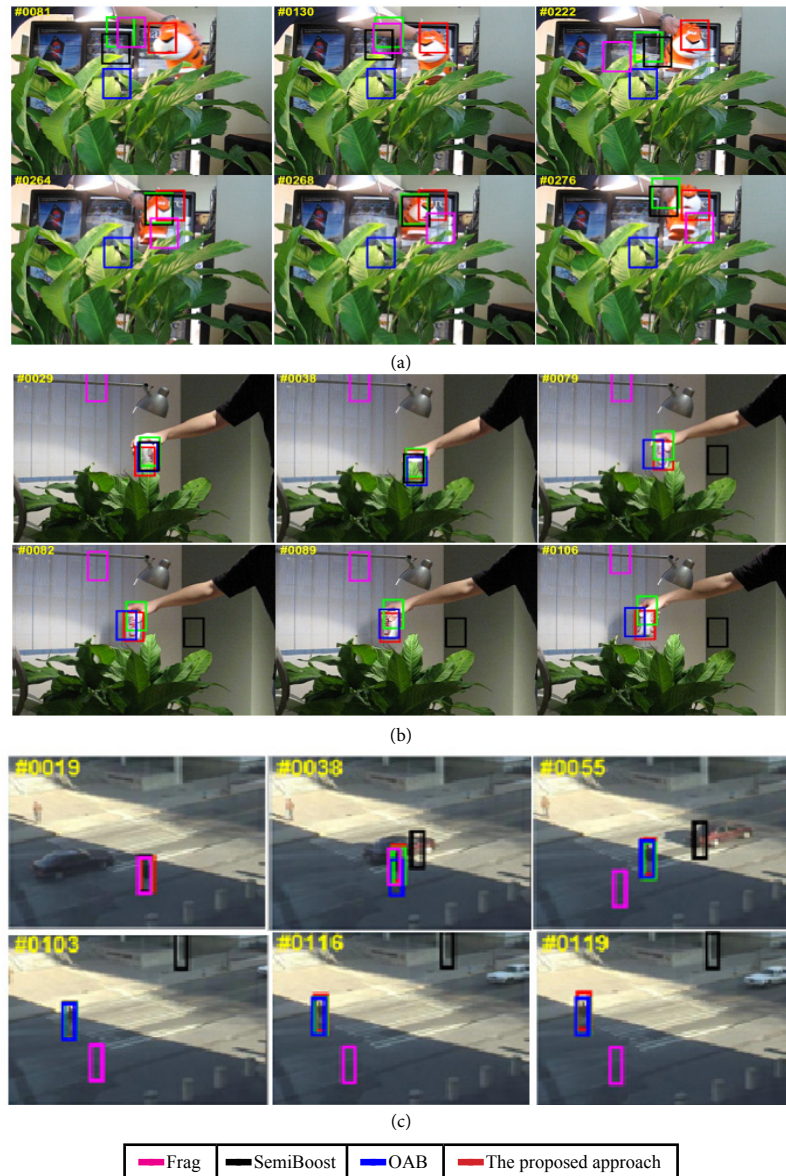
Figure 2. Tracking Object Location: Snapshots of tracking results, using SRE evaluation metric and presenting instances of pose, scale and illumination change, occluding, background clutter and motion blur

and Fast Motion. Adapting the SRE metric, the proposed algorithm shows better results as illustrated in Figure 2 (c) than other model update trackers.

MotorRolling: It includes some challenges such as illumination, scale, background and motion effects (Figure 3 (a)). The proposed approach outperforms the others*.*

Deer: This video shows the lack of Frag (Figure 3 (b)), OAB and MILtrack e.g frame #12, frame #26 and frame #34 and the robustness of the proposed approach to Background Clutters.

Matrix: In this sequence (Figure 3 (c)), the purpose is to track an object that varies in

Illumination and Scale and presents occlusion and background clutters.

Jogging: In this video, the target object is fully occluded. In frame #86 in Figure 3 (d) the proposed tracker fails because the object of interest is covered.

Dog1: In this video, the challenging aspect is represented by the scale variation, in plane rotation and out of plane rotation. As presented in Figure 3 (e), the proposed tracker tracks the target object well.

Table 1 illustrates the details of the challenges encountered in each sequence.
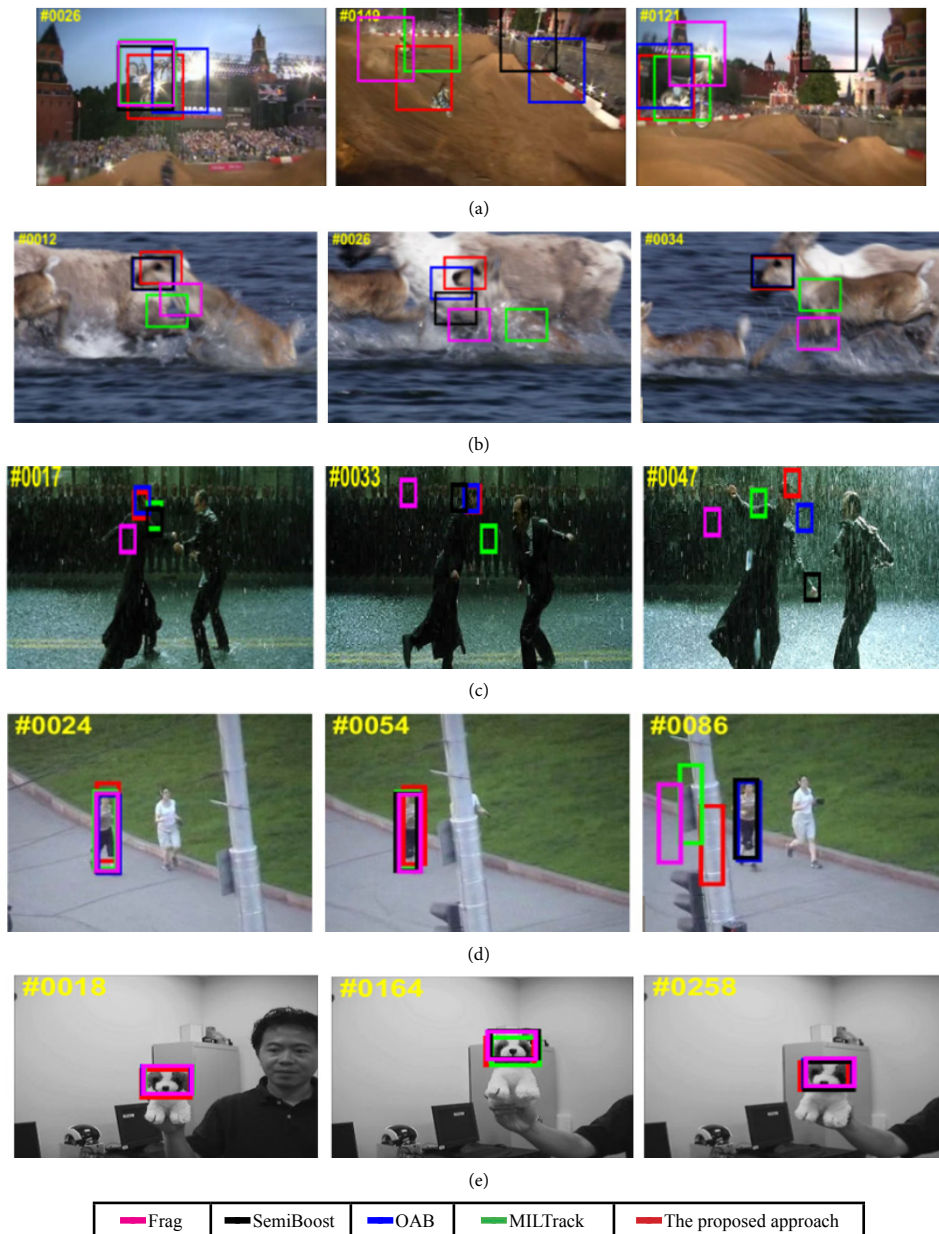
(a)

(b)

(c)

(d)

(e)

| ▬ Frag | ▬ SemiBoost | ▬ OAB | ▬ MILTrack | ▬ The proposed approach |

Figure 3. Tracking Object Location: Snapshots of tracking results, using TRE evaluation metric and presenting instances of out-of-plane, rotation, occluding clutter, scale and illumination change

Table 1. Challenging aspects In The Experimented Clip Sequences

Video sequences	Number of frames	Pose Change	Scale Change	Illumination Change	Occlusion	Background Clutters	Motion Blur
Tiger1	354	•		•	•		•
Coke	291			•	•	•	
Crossing	120	•	•			•	
MotorRolling	164		•	•		•	•
Deer	71					•	
Matrix	100		•	•	•	•	
Jogging	108	•			•		
Dog1	1350		•				

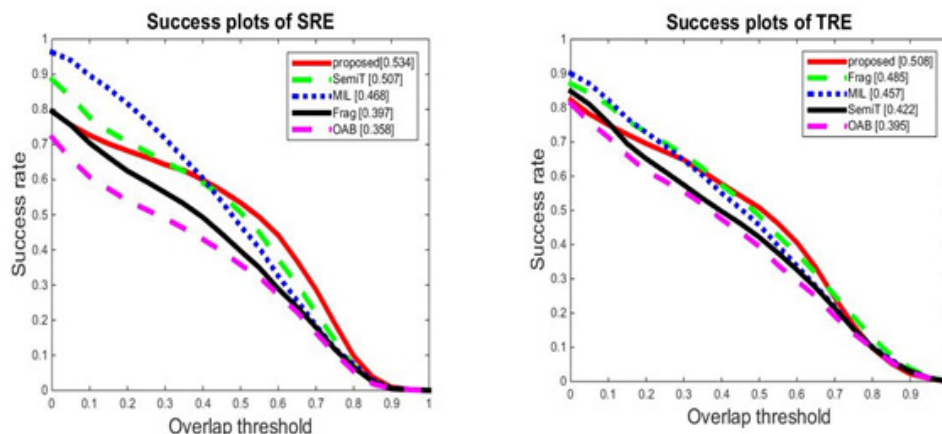The corresponding tracking challenge is included - •

Figure 4. Success plot of SRE and TRE for different trackers facing the overlap threshold

5.4 Comparison to the State of the Art

As it was previously described, the algorithm proposed in this paper is distinct from other trackers. First, it is different from the IP tracker [27] that uses generative approach and encodes target location by sparse representation. Second, the proposed approach uses a linear combination of generalized Haar-like features extracted around interest points but these trackers ([7], [11]) adopt the full templates for sparse representation and [28] use random measurement matrix which are more sensitive to occlusion.

In this approach, computation of features around each interest point makes the proposed tracker more robust to handle with deformations like rotation, scaling, illumination and partial occlusion.

Analogous representations, e.g. local binary patterns [29] and generalized Haar-like features [6], have proved to be more effective in handling occlusion.

Success plots of SRE and TRE in Figure 4 show that the proposed method outperforms the state of

the art methods (Frag, SemiBoost, OAB and MIL Track) when varying the overlap threshold. The AUC values have been measured and the proposed approach has reached 0,534 for SRE and 0,508 for TRE.

6. Conclusion

This work proposed a novel robust interest point-based tracker using an adaptive appearance model of a tracking system. The combination of both interest point and appearance model leads to a robust tracker. The performance of the proposed approach has been benchmarked with diverse publicly available video clips for evaluating its effectiveness when confronted with various challenges including pose, scale and illumination variation, occlusions, background clutter and motion blur. In a further work, the extension of this robust tracker will be sought in order to make it able to handle other tracking tasks such as tracking multiple objects and tracking contours. Also, an attempt will be made in order to apply it for detecting and tracking video salient objects.

REFERENCES

1. Adam, A., Rivlin, E. & Shimshoni, I. (2006). Robust fragments-based Tracking using the Integral Histogram. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition – volume 1,* New York, USA (pp. 798 - 805).

2. Alahi, A., Ramanathan, V., Goel, K., Robicquet, A., Sadeghian, A., Fei-Fei, L. & Savarese, S. (2017). Learning to Predict Human Behavior in Crowded Scenes, *Group and Crowd Behavior for Computer Vision,* 183-207. Elsevier.

3. Ak, A., Topuz, V. & Ersan, E. (2018). Visual Servoing Application for Inverse Kinematics of Robotic Arm Using Artificial Neural Networks, *Studies in Informatics and Control*, *27*(2), 183-190. DOI: 10.24846/ v27i2y201806

4. Avidan, S. (2004). Support Vector Tracking, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *26*(8), 1064-1072.

5. Avidan, S. (2005). Ensemble Tracking. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, San Diego, CA, USA (pp. 494 - 501).

6. Babenko, B., Yang, M. H. & Belongie, S. (2009). Visual Tracking with Online Multiple Instance Learning. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition,* Miami, Florida, USA (pp. 983-990).

7. Babenko, B., Yang, M.-H. & Belongie, S. (2011). Robust Object Tracking with Online Multiple Instance Learning, *IEEE Transactions on Pattern Analysis and Machine Intelligence - PAMI*, *33*(8), 1619-1632.

8. Babu, R. V. & Parate, P. (2013). Interest Points Based Object Tracking via Sparse Representation. In *Proceedings of 20th IEEE International Conference on Image Processing, ICIP-2013*, Melbourne, Australia (pp. 2963-2967).

9. Black, M. J.& Jepson, A. D. (1998). Eigen Tracking: Robust Matching and Tracking of Articulated Objects using a View-based Representation, *International Journal of Computer Vision, 26*(1), 63–84.

10. Bradski, G. R. (1998). Computer vision face tracking for use in a perceptual user interface, *Intel Technology Journal*, 2nd Quarter.

11. Chakrabort, B., Holteb, M. B., Moeslund, T. B. & Gonzàleza, J. (2012). Selective Spatio-temporal Interest Points, *Computer Vision and Image Understanding, 116*(3), 396-410.

12. Collins, R. T. & Liu, Y. (2003). On-line Selection of Discriminative Tracking Features. In *Proceedings of the International Conference in Computer Vision*, Nice, France (pp. 346-352).

13. Comaniciu, D., Ramesh, V. & Meer, P. (2003). Kernel-Based Object Tracking, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *25*(5), 564-577.

14. Dietterich, T. G., Lathrop, R. H. & Perez, L. T. (1997). Solving the Multiple-Instance Problem with Axis Parallel Rectangles, *Artificial Intelligence*, *89*(1-2), 31-71.

15. Elabbessi, S., Abdellaoui, M. & Douik, A. (2016). An improved Selective Spatio-temporal Interest Point Detector. In the *2nd International Conference on Advanced Technologies for Signal and Image Processing, ATSIP'2016*, Monastir, Tunisia (pp. 147-150).

16. Grabner, H., Grabner, M. & Bischof, H. (2006). Real-Time Tracking via On-line Boosting. In *British Machine Vision Conference*, *1*(5), Edinburgh, UK (p. 6).

17. Grabner, H., Leistner, C. & Bischof, H. (2008). Semi-Supervised On-line Boosting for Robust Tracking. In *Proceedings of 10th European Conference on Computer Vision, ECCV 2008*, Marseille, France (pp. 234-247).

18. Henriques, J. F., Caseiro, R., Martins, P. & Batista, J. (2012). Exploiting the Circulant Structure of Tracking-by-Detection with Kernels. In *Proceedings of the European Conference on Computer Vision, ECCV'12*, Firenze, Italy (pp. 702-715).

19. Kwon, J. & Lee, K. M. (2010). Visual Tracking Decomposition. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, USA (pp. 1269-1276).

20. Lee, D., Kim, G., Kim, D., Myung H. & Choi, H. T. (2012). Vision-based Object detection and Tracking for Autonomous Nnavigation of Underwater Robots, *Ocean Engineering*, *48*, 59-68.

21. Lin, R., Ross, D., Lim, J. & Yang, M. H. (2004). Adaptive Discriminative Generative Model and Its Applications. In *Proceedings of Advances in Neural Information Processing Systems*, Vancouver, Canada (pp. 801-808).

22. Liu, X. & Yu, T. (2007) Gradient Feature Selection for Online Boosting. In *Proceedings of IEEE International Conference on Computer Vision,* Rio de Janeiro, Brazil (pp. 1-8).

23. Liu, Z., Zavesky, E., Shahraray, B., Gibbon, D. & Basso, A. (2008) Brief and high-interest Video Summary Generation: Evaluating the AT&T labs rushes summarizations. In *Proceedings of 2nd ACM TRECVid Video Summarization Workshop*, Vancouver, Canada (pp. 21-25).

24. Ojala, T., Pietikäinen, M. & Mäenpää, T. (2002). Multiresolution Gray-scale and Rotation Invariant Texture Classification With Local Binary Patterns, *IEEE Transactions on Pattern Analysis and Machine Intelligence - PAMI, 24*(7), 971–987.

25. Ross, D., Lim, J., Lin, R.-S. & Yang, M. H. (2008). Incremental Learning for Robust Visual Tracking, *International Journal of Computer Vision IJCV, 77*(1), 125-141.

26. Viola, P. & Jones, M. (2001) Rapid Object Detection Using a Boosted Cascade of Simple Features. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, vol. 1,* Kauai, Hawaii, USA (511-518).

27. Vrejoiu, M. H. (2017). A Pattern Matching Method and Algorithm for Face Detection, *Studies in Informatics and Control, 26*(1), 75-86. DOI: 10.24846/v26i1y201709

28. Wang, J., Chen, X. & Gao, W. (2005). Online Selecting Discriminative Tracking Features Using Particle Filter. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, vol. 2,* San Diego, CA, USA (1037-1042).

29. Wu, Y., Lim, J. & Yang, M. (2013). Online Object Tracking: A Benchmark. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR'13,* Portland, Oregon, USA (pp. 2411-2418).

30. Zhang, K., Zhang, L. & Yang, M. H. (2012). Real-time Compressive Tracking. In *Proceedings of the European Conference on Computer Vision, ECCV'12,* Firenze, Italy (pp. 864-877).