Real-Time Face Tracking under Long-Term Full Occlusions

Martin Soldić, Darijan Marčetić, Marijo Maračić, Darko Mihalić, Slobodan Ribarić
Faculty of Electrical Engineering and Computing, University of Zagreb
{martin.soldic, darijan.marcetic, marijo.maracic, slobodan.ribaric}@fer.hr

Abstract — **The identified weaknesses of most of state-of-the-art trackers are inability to cope with long-term full occlusions, abrupt motion, detecting and tracking a reappeared target. In this paper, we present a robust real-time single face tracking system with several new key features: semi-automatic target tracking initialization based on a robust face detector, an effective target loss estimation based on a response of a position correlation filter, a candidate image patch selection for re-initialization supported with a short- and long-term memories (STM and LTM). These memories are used for tracking re-initialization during online learning procedure. The STM is used to select an image patch as candidate for re-tracking based on stored position correlation filters (from current frame) in case of short-term full occlusions, while the LTM stores aggregated position correlation filters (online learned) is used to recover the tracker from long-term full occlusions. Validation of the tracking system was performed by evaluation on a subset of videos from Online Tracking Benchmark (OTB) dataset and our own video.**

Keywords — tracking, long-term full occlusion, re-tracking, correlation filter, face.

I. INTRODUCTION

Visual tracking is a process of locating, i.e. estimating the spatial and temporal parameters, of a moving object (or multiple objects) in consecutive video frames, starting from an initial image patch containing the target (object) given in the first frame. Visual tracking of objects in video sequences taken in realistic scenarios is a challenging and hard computer vision task. There are three main desired characteristics of a robust tracking system [1]: robustness, adaptivity, and real-time processing. Robustness implies ability to track an object of interest under conditions such as changing illuminations, specularities, long-term full occlusions, scale variations, richness of colour and texture for both object and background and/or abrupt object motion. Adaptivity is related to a requirement of continuous adaptation to the actual object and/or environment that undergo changes. For near real-time processing, it is necessary to achieve at least 15 frames per second [1].

The online learning based tracking methods are used to cope with appearance variations of a target object. There are two main approaches to online learning based tracking methods [2]: generative and discriminative. The first approach uses templates or statistical models for describing a target appearance and the models are online updated to adapt to appearance changes. Discriminative methods are based on binary classifiers which are trained and updated online to distinguish the object from the background. These methods are referred as tracking-by-detection.

In this paper, we describe a robust real-time single face tracking system capable to handle with appearance variations of a face, as well as, with a long-term full occlusion. It is based on combination of: i) the discriminative scale space tracking (DSST) [3] to achieve robustness on a face position and scale variation, ii) the-state-of-the-art face detector based on normalized differences of pixels (NDP) [4] for detection of the face when a target is lost, and iii) FIFO short- and long-term memories (STM and LTM) for tracking re-initialization after short- and long-term full-face occlusion. A long-term full occlusion is related to the situation when a tracked object (face) is entirely invisible for a number of consecutive frames, while knowing that the object has not left the area of view of the camera.

Tracking methods [5-9] typically do not put much focus on the initialization and long-term full occlusion issues, but instead they are focused on resolving variable visual appearance of tracked objects. Assumptions are that the object of interest is manually annotated in the first frame which is used for initialization of a tracking procedure, and that the object is present in all subsequent frames (or partially occluded in a short sequence of frames) of the analysed video. These two assumptions are not valid in realistic scenarios of tracking of an object, where manual ground truth object annotation is not available and occlusions are very common phenomenon. For example, Fig. 1. illustrates the comparison of the result of the tracking method DSST [3], which obtains the top rank in performance by outperforming 19 state-of-the-art trackers on OTB and 37 state-of-the-art trackers on VOT2014 datasets, and our proposed face tracking system. From Fig. 1 it can be seen that in case of the DSST the occlusion results with permanent target loss, because the target window is "glued" at the position where the target loss occurred (frames #21, and #30 first row).

978-1-5090-4011-7/17/$31.00 © 2017 IEEE

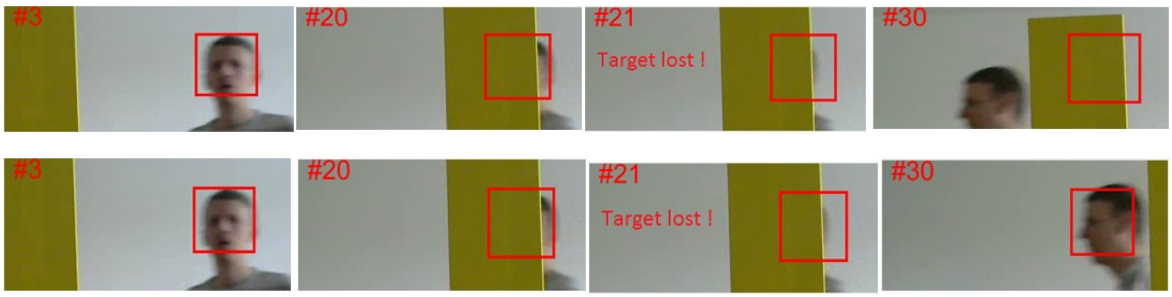


Fig.1. Comparison of the result the tracking method DSST (above) and our proposed face tracking system (bottom).

II. BACKGROUND

Two main components, besides the LTM and STM, of the proposed tracking system are the face NPD detector and the DSST object tracker. In this section, these two components are briefly described.

A. Face NPD detector

State-of-the-art face detector based on Normalized Pixel Difference (NPD) [4] is used because of its high recall and speed. The NPD features are defined as: $f(v_{i,j}, v_{k,l}) = (v_{i,j} - v_{k,l}) / (v_{i,j} + v_{k,l})$, where $v_{i,j}$, $v_{k,l} \geq 0$ are intensity values of two pixels at the positions (i, j) and (k, l). By definition, f(0,0) is 0. The NPD feature has the following properties [4]: i) the NPD is antisymmetric; ii) the sign of NPD is an indicator of the ordinal relationship; iii) the NPD is a scale invariant; iv) the NPD is bounded in [-1, 1].

The properties of features used in the NPD detector are suitable for reducing feature space, encoding the intrinsic structure of an object image, increasing robustness to illumination changes. The bounded property makes an NPD feature amenable for threshold learning in tree-based classifiers [10]. A quadratic splitting strategy for deep tree, where depth was eight is used for learning. For practical reasons, the values of NPD features are quantized into L = 256 bins. To reduce redundant information contained in the NPD features the AdaBoost algorithm is used to select the most discriminative features. Based on these features the strong classifier is constructed. The score $ScoreNPD(I, S_i)$ is obtained based on result of 1226 deep quadratic trees and 46401 NPD features, but the average number of feature evaluations per detection window is only 114.5. The output of the NPD detector is represented by square regions S_i i = 1, 2, ... n, where n is number of detected regions of interest in an image I. For each region S_i in an image I, the score $ScoreNPD(I, S_i)$ is calculated. The regions S_i i = 1, 2, ..., $j \leq n$, with scores $ScoreNPD(I, S_i)$ greater than zero are classified as faces [4]. The detector is trained by using the same dataset and training procedure as described in [4].

B. DSST tracker

The DSST tracker [3] first searches an optimal target position using discriminative position correlation filter, and then applies a scale correlation filter to find an optimal target scale. Thus, it avoids computationally expensive exhaustive search of a position-scale space. The circular correlation score is efficiently obtained in the Fourier domain what enables real time performance.

The position and scale correlation filters are online updated for each consecutive frame. These features make the DSST one of the best state-of -the-art visual tracking algorithms. The steps performed by the DSST tracker, which are relevant for our proposed tacking system, are described in more detail as follows.

1) Initialization/learning of position

The DSST in the first frame uses ground truth annotation of the tracked object (face) to learn the visual appearance and scale of the tracked object. First, two multichannel feature correlation filters are calculated independently for position and the scale.

For an image patch f_1 centered around the target, nine HOG features [11, 12] (nine-bin histogram) for each colour channel are calculated and appended to the grey image of the image patch f_1, thus a (9×3+1)-channel position correlation filter h^P that has dimension $M \times N \times 28$ is obtained, where M and N are dimensions of image patch f_1. The position correlation filter is transformed into Fourier domain.

2) Initialization/learning of scale

Patches at 33 scales (from 0.7284 to 1.3728, with step 1.02; regarding the image patch f_1) centered around the target position estimated at the first step are taken from the first frame. The initial assumption is that the scale change of the target from frame to frame is limited from 0.7284 to 1.3728 relative size of the target from the first frame. These 33 patches are resized to the size of the image patch f_1. The HOG features are calculated for each image patch (9×3 channels) and 4 texture channels are appended and arranged as one row vector, thus giving d_s-channel scale correlation filter h^S that has dimension $33 \times d_s$; where 33 is number of scales and d_s is number of HOG and texture features for each patch. The maximum feature descriptor length d_s is limited to 992. The scale correlation filter is transformed into Fourier domain.

3) Target position estimation

For each consecutive frame image patch f_t , $t > 1$, centered around the target position estimated in the previous frame, but with doubled patch size (width and height), is selected as the candidate for the target position search. For this image patch f_t the same form of feature representation is obtained as the form of representation of the 28-channel position correlation filter,

as described for the first frame during the target position learning procedure.

Then the new target position is estimated by performing the circular correlation in the Fourier domain between the 28-channel position correlation filter from the previous frame and features obtained from the image patch f_t in the current frame. The new position of the target in the current frame is determined based on the maximal correlation response. The explanation of doubling the patch size is the assumption that the position change of the target from frame to frame is limited to the half of the value of the target width/height in the horizontal/vertical direction.

4) Target scale estimation

Patches at 33 scales centered around the target position estimated at the previous step are taken from the current frame. For these 33 image patches the same form of feature representation is obtained as the form of representation of the scale correlation filter, as described for the first frame during the target scale learning procedure. Then, a new target scale is estimated by performing the circular correlation in the Fourier domain between the scale correlation filter and features obtained as described above. The new scale of the target in the current frame is determined based on the maximal correlation response which corresponds to one of 33 scales.

5) Filters update/online learning

Position and scale correlation filters are online updated for position and scale target estimation in the next frame. Position and scale estimated for the current frame are used to calculate new position and scale correlation filters. Both filters are updated separately. These new values of filters contribute with the learning rate η and the corresponding values of the filters from the previous frame contribute with the 1- η, where typical value of η is 0.025.

III. FACE TRACKING SYSTEM

The main identified weaknesses of the DSST are: sensitivity to long-term full occlusions what consequently results with the problems of position and scale correlation filter degradation during online learning procedure (i.e. target loss), and inability to track the reappeared target (Fig. 1). In order to solve these weaknesses, in our tracking system the several new key features are introduced to the DSST: face detector, semi-automatic target tracking initialization, target loss estimation, re-initialization supported with the short- and long-term memories (STM and LTM) that improves online learning. Descriptions of these new features are given in the next subsections.

A. Face detection and semi-automatic target tracking initialization

The NPD face detector is used for the automatic DSST initialization and also to determine image patches that are candidates for re-initialization of the tracker when the face is lost. Besides of the well-known reasons for face loss (e.g. illumination changes, occlusion, background clutter, camera motion...), a special attention is devoted to the problem of long-term full occlusion in this work. The NPD face detector is invoked for the first frame during initialization, and for each frame in which tracking failure occurred. Instead of manually tracker initialization, the NPD detector is invoked in the first frame (or consecutive frames until at least one face is detected) for finding image patches that are candidates for tracking. A user is requested to select a single face to be tracked. Before initializing the DSST and starting tracking, detected face window of the NPD detector is extended by 15 % in each direction to be sure that it captures entire face.

B. Tracking failure detection

Tracking failure is detected based on a measurement of peak strength called the Peak to Sidelobe Ratio (*PSR*) obtained from response of the position correlation filter [8]. The position correlation response is divided to two areas (Fig. 2): peak with its surrounding area and the area called sidelobe. The square area surrounding the peak has a square side that is 12% of the size of the position correlation response (which is represented as 2D square matrix). The sidelobe is the remaining area that excludes the area surrounding the peak. The value of the *PSR* is computed by the following formula: $PSR = \frac{(MaxPeak - \mu)}{\sigma}$, where *MaxPeak* is the maximum peak value, the mean value μ and standard deviation σ are calculated from the sidelobe of the position correlation response. The value of the *PSR* for so-called good tracking (i.e. manually annotation) is typically between 20 and 60. In the case of tracking failure the *PSR* is less than 10 [8].

Figure 2. illustrates position correlation responses when the target is visible and a position error is low. Figure 3. illustrates position correlation responses when the target is under a large occlusion and the PSR is below 10 (the target is lost). Figure 4. illustrates position correlation responses when the target is lost.

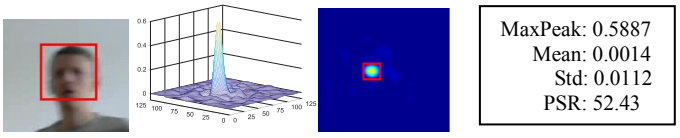


Fig 2. Strong peak response when the target (face) is visible: a) target window (position and scale); b) 3D representation of position correlation responses; c) 2D representation of position correlation responses with denoted peak with its surrounding area and sidelobe, d) PSR value.

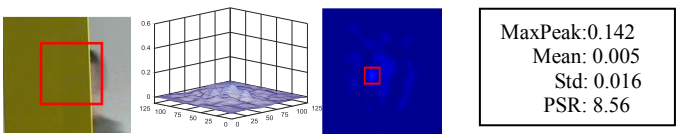
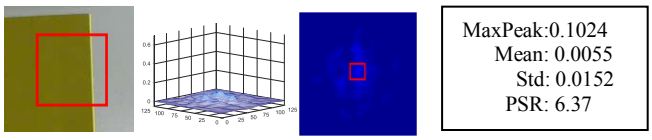


Fig 3. Low peak response when the target (face) is under a large occlusion: a) target window (position and scale); b) 3D representation of position correlation responses; c) 2D representation of position correlation responses, d) PSR value.



a) b) c) d)

Fig 4. Low peak response when the target (face) is lost: a) target window (position and scale); b) 3D representation of position correlation responses; c) 2D representation of position correlation responses, d) PSR value.

C. *Short and long-term memory for improved online learning*

Original DSST tracker assumes that the target is always visible in the frame or partially occluded in only few consecutive frames, and that there are no abrupt changes of visual appearance and/or the location of the tracked object, what is not true for in the wild tracking. The DSST tracking procedure updates the position correlation filter by combining the information from the current and aggregation of all previous frames as described in Section II.B.5. This process of updating a position correlation filter is inappropriate for handling the following problems: tracking failure caused by long-term full occlusion, incorrect current estimation of the location of the tracked object due to abrupt change of visual appearance. Note that each frame with incorrect/poor estimation (due to drift phenomenon) of the position, corrupts the position and scale correlation filters what can lead to permanent loss of a target. To resolve these issues related to the DSST, the long (LTM) and short-term memories (STM) are used. Both are implemented as a FIFO memory with depths d_{LTM} and d_{STM}, respectively. These memories are used for tracking re-initialization to prevent position correlation filter degradation during online learning procedure. The LTM stores one online-learned position correlation filter (Section II.B.5.) every c_{LTM} >1 frames, while the STM stores one current position correlation filter for every frame (Section II.B.3.), but only if the value *PSR* is above 10 for both cases. Note that content of the LTM and STM are unaltered for frames in which tracking failure is detected, thus preventing position correlation filter degradation during online learning procedure. With this approach, it is possible to re-initialize tracker and continue tracking after unlimited number of frames with target loss.

D. *Selection one of candidates for tracking re-initialization*

The NPD is invoked when the tracking failure is detected in the current frame. The face NPD detector can detect image patches p_i; i = 1, 2,..n, where each p_i is face or non-face (false positive detections). All these detected image patches are candidates for tracking re-initialization. Only at most one of these image patches is used for tracking re-initialization (single target tracking). The procedure of selection of the candidate for tracking is as follows: For every image patch p_i; i = 1, 2 ,..n; feature representation is obtained and correlated with all stored position correlation filters pcf_j; $j = 1, 2, \ldots (d_{LTM} + d_{STM})$ in the LTM and STM and PSR_i^j values are calculated. Note that first d_{LTM} indexes $j = 1, 2, \ldots, d_{LTM}$ correspond to the LTM and the last d_{STM}, $j = d_{LTM}+1, \ldots, (d_{LTM} + d_{STM})$ correspond to the STM. The maximal $PSR_{i_{max}}^{j_{max}}$ value from $n \times (d_{LTM} + d_{STM})$ *PSR* values is selected. The i_{max} is an index of the image patch used for re-initialization. The position and scale of this image patch is used for tracking re-initialization. The j_{max} is an index of the position correlation filter taken from either LTM or STM. This position correlation filter is used for online learning procedure which updates its value with the position correlation filter estimated from the selected candidate image patch as described in Section II.B.1. The current scale correlation filter is calculated from the selected image patch with index i_{max} (Section II.B.2.). These position and scale correlation filters are used in the next frame for target tracking. An illustration of selection of one of candidates for tracking re-initialization is depicted in Fig. 5. In the 36th frame the target is lost and then invoked NPD returned three image patches as candidates for re-initialization. The image patch p_1 which has maximal $PSR_1^{14} = 28.31$ is used for tracker re-initialization and corresponding position correlation filter from the fourth location of the STM is taken for online learning procedure.

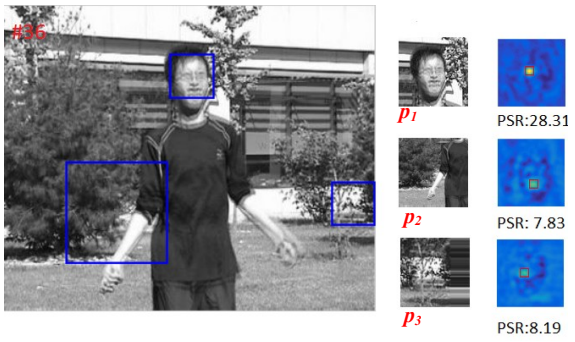


Fig 5. NPD detections with corresponding patches and best correlation scores between stored position correlation filters and features obtained from the patches.

E. *Proposed tracking procedure*

The procedure of the tracking system can be described as follows:

STEP 1: The NPD detector is invoked in the first frame (or consecutive frames until at least one face is detected) for finding image patches that are candidates for tracking and then initialize the tracker (Section II.B.1.).

STEP 2: Initialize the FIFO LTM and STM with depth d_{LTM} and d_{STM}, respectively (Section III.C.). Initially, only current position correlation filter obtained in the STEP 1 is stored in both LTM and STM.

STEP 3: Read a new frame and determine position and scale by the DSST (Sections II.B.1-2.).

Step 4: IF the $PSR \geq 10$ **THEN** store current position correlation filter and online learned position correlation filter in the STM and the LTM (Section II.B.3.), respecivelly. Note that the LTM is periodically updated every c_{LTM} frames, and the STM is updated for every frame. **GOTO STEP 3, ELSE** ($PSR < 10$) **GOTO STEP 5**.

Tracking failure:

STEP 5: Invoke the NPD detector for target redetection in the current frame.

IF no candidate image patches have been found **THEN** read new frame and **GOTO STEP 5, ELSE GOTO STEP 6**

STEP 6: Perform selection of one of the candidates for tracking re-initialization (Section III.D.)

STEP 7: Perform tracking re-initialization (Section III.D.) **IF** the last frame in the video sequence is reached **GOTO END, ELSE GOTO STEP 3**.

END.

IV. EXPERIMENTAL RESULTS

Validation of our approach was performed by evaluation on a subset of videos from the Online Tracking Benchmark (OTB) [13] dataset. Characteristic of selected videos are depicted in Table 1. Note that among 100 available videos from the OTB we selected 9 videos with faces having variety of characteristics (Table 1.). One video (LTFullOcc) with long-term full face occlusion was additionally created and used for evaluation, because such videos are not included in available benchmarks. The last frames with tracking results from used videos are given in Fig. 6. All working parameters of the DSST algorithm are set as described in [3] (learning parameter $\eta = 0.025$, number of scales 33, from 0.7284 to 1.3728, with step 1.02 etc.). Program implementation of the DSST available at [3] is used. Three parameters that are specific to our proposed tracking system are: LTM depth d_{LTM}, STM depth d_{STM} and LTM capture rate c_{LTM}. We have heuristically determined following values of these parameters: $d_{LTM} = 10$, $d_{STM} = 15$ and $c_{LTM} = 25$. The values of the d_{LTM} and d_{STM} should be proportional, and the value of the c_{LTM} should be inversely-proportional to the degree of variability of a visual appearance of a target. Refreshing of both STM and LTM are blocked for frames when the target is under full occlusion.

Table 1. Characteristics of videos used for evaluation

Features / Videos	IV	SV	IPR		OPR	BC	OCC	DEF	MB	FM	OV	LTO
trellis	+	+	+		+	+						
david	+	+	+		+		+	+	+			
david2			+		+							
dudek		+	+		+	+	+	+		+	+	
freeman1		+	+		+							
fleetface		+	+		+			+	+	+		
boy		+	+		+				+	+		
jumping									+	+		
girl		+	+		+		+					
LTFullOcc		+	+		+		+		+	+	+	+

Features: IV - Illumination Variation, SV - Scale Variation, IPR - In-Plane Rotation, OPR - Out-of-Plane Rotation, BC - Background Clutters, OCC - Occlusion, DEF - Deformation, MB - Motion Blur, FM - Fast Motion, OV - Out-of-View, LTO - Long-Term Occlusion

Evaluation metrics of the tracking results on the OTB dataset plus LTFullOcc video are reported using central position error (CPE) and tracking speed in frames per second (FPS) in Fig. 6. and Table. 2, respectively. The CPE indicates a distance between an estimated and a ground truth position of a target. Note that videos from OTB have no long-term full occlusion. The results from Fig. 7. validate that the proposed system can successfully semi-automatically initialize tracking and perform re-tracking in the case when the target loss occurs due to long-term occlusion or other reasons.

Time performance comparison between the DSST and the proposed system is given in Table 2. Time performance depends on the number of re-initializations performed due to target loss. Time required for semi-automatic faces selection on the first frame is not included in time performance results.

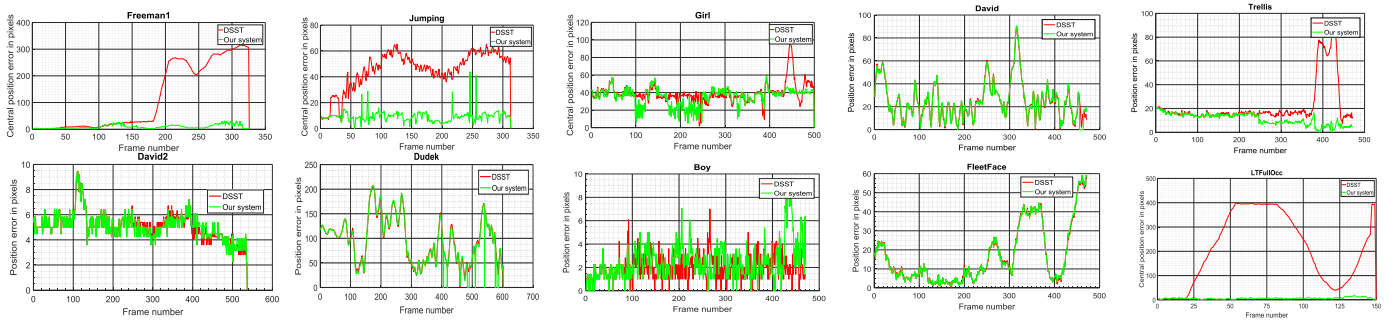


Fig 6. Central position errors in pixels for each frame of videos used for evaluation.

Fig 7. Tracking results obtained in the last frame for videos used for validation. The ground truth annotations are depicted with blue, DSST with red, and for the proposed tracking system with green rectangle.

Table 2. Time performance comparison between the DSST and the proposed system

Video	Resolution	Num. frames	DSST		Our system		Diff. (%)
			Time (s)	FPS	Time (s)	FPS	
Freeman1	360x240	326	3.35	**97.3**	4.82	**67.63**	**69.51**
jumping	352x288	313	4.05	**77.3**	18.23	**17.17**	**22.21**
girl	128x96	500	6.10	**81.9**	6.33	**78.9**	**96.34**
David	320x240	770	32.85	**49.35**	49.40	**15.59**	**31.59**
Trellis	320x240	569	30.8	**18.47**	34.50	**16.48**	**89.23**
David2	320x240	537	6.58	**76.72**	9.34	**57.5**	**74.95**
Dudek	720x480	1145	162.91	**7.02**	171.13	**6.69**	**95.30**
FleetFace	720x480	707	48.06	**14.71**	53.6	**13.19**	**89.67**
Boy	640x480	602	11.4	**52.81**	15.4	**39.1**	**74.04**
LTFullOcc	640x480	150	5.23	**28.68**	7.32	**20.5**	**71.48**

V. CONCLUSIONS

In this paper, we presented the robust real-time single face tracking system. To avoid identified weaknesses of the tracking systems, in our tracking system the several key features are introduced: (semi)automatic target tracking initialization supported by face detection, target loss estimation, re-initialization supported with a short- and long-term memories (STM and LTM). The system combines state-of-the-art Normalized-pixel differences (NPD) face detector and Discriminative Scale Space Tracker (DSST) and FIFO STM and LTM. These memories are used for tracking re-initialization to prevent position correlation filter degradation during online learning procedure. The reasons for two different types of memories are that assumptions that the STM is used to recover from short-term full occlusions (based on sequence of stored current position correlation filters) and the LTM is used to recover from long-term full occlusions (based on sequence of stored aggregated position correlation filters). With this approach, it is possible to re-initialize tracker and continue tracking after unlimited number of frames with target loss. This feature enables candidate image patch selection and improves online learning. Preliminary qualitative evaluation based on central position error metrics showed promising results in comparison with the DSST tracker, especially for long-term full-face occlusion. In the future work we plan to adopt the system for multi-face tracking, improve the speed of the tracker by reducing the time needed for re-tracking and supplement it with face de-identification capabilities.

ACKNOWLEDGMENTS

This work has been supported by the Croatian Science Foundation under project 6733 De-identification for Privacy Protection in Surveillance Systems (DePPSS).

REFERENCES

[1] H. Yang, L. Shao, F. Zheng, L. Wang and Z. Song: "Recent advances and trends in visual tracking: A review", Neurocomputing, vol. 74, no. 18, Nov. 2011, pp. 3823–3831.

[2] A. Smeulders, D. Chu, R. Cucchiara, S. Calderara, A. Dehghan and M. Shah: "Visual tracking: an experimental survey", IEEE TPAMI, vol.36, no.7, July 2014, pp.1442-1468.

[3] M. Danelljan, G. Häger, F. S. Khan and M.Felsberg: "Accurate Scale Estimation for Robust Visual Tracking", Proceedings of the British Machine Vision Conference (BMVC), 2014.

[4] S. Liao, A.K. Jain and S. Z. Li: "A Fast and Accurate Unconstrained Face Detector", IEEE TPAMI, vol. 38, Issue:2, 2016, pp. 211-223.

[5] J. Henriques, R. Caseiro, P. Martins and J. Batista:" Exploiting the circulant structure of tracking-by-detection with kernels", IEEE ECCV, 2012, pp.702-715.

[6] Z. Kalal, J. Matas and K. Mikolajczyk: "P-n learning: Bootstrapping binary classifiers by structural constraints", IEEE CVPR, 2010, pp. 49-56.

[7] A. Adam, E. Revlin and I. Shimshoni: "Robust fragments-based tracking using the integral histogram", IEEE CVPR 2006, pp.798-805.

[8] D. S. Bolme, J. R. Beveridge, B. A. Draper and Y. M. Lui:" Visual object tracking using adaptive correlation filters", IEEE CVPR, 2010, pp.2544-2550.

[9] J. Kwon and K. M. Lee: "Tracking by sampling trackers", IEEE ICCV 2011, pp.1195-1202.

[10] Breiman, L., Friedman, J. H., Olshen, R., & Stone, C. J: "Classification and Regression Trees", Chapman & Hall/CRC, New York, 1984.

[11] P. Dollar, "Piotr's Image and Video Matlab Toolbox", https://pdollar.github.io/toolbox/

[12] P. F. Felzenszwalb, R. B. Girschick, D. A. McAllester and D. Ramanan:" Object Detection with discriminatively trained part-based models", IEEE TPAMI, vol.32, no.9, pp.1627-1645, 2010.

[13] Y. Wu, J.Lim and M.-H. Yang: "Online object tracking: A benchmark", IEEE CVPR 2013, pp.2411-2418.