SETh: The Method for Long-Term Object Tracking

Karol Jedrasiak[1], Mariusz Andrzejczak[2], and Aleksander Nawrat[1]

[1] Institute of Automatic Control, Silesian University of Technology
Akademicka 16, 44-100 Gliwice, Poland
[2] Ośrodek Badawczo-Rozwojowy Urządzeń Mechanicznych "OBRUM" sp. z o.o.
ul. Toszecka 102, 44-117 Gliwice, Poland
{karol.jedrasiak}@polsl.pl

Abstract. The article presents a novel long-term object tracking method called SETh. It is an adaptive tracking by detection method which allows near real-time tracking within challenging sequences. The algorithm consists of three stages: detection, verification and learning. In order to measure the performance of the method a video data set consisting of more than a hundred videos was created and manually labelled by a human. Quality of the tracking by SETh was compared against five state-of-the-art methods. The presented method achieved results comparable and mostly exceeding the existing methods, which proves its capability for real life applications like e.g. vision-based control of UAVs.

Keywords: object tracking, long-term tracking, adaptive, image processing.

1 Introduction

The modern world is full of cameras placed in supermarkets, banks, or on the streets. Each of these cameras record movies in the form of a compressed image sequence. A huge number of recorded information makes it impossible to be verified by human. Therefore there is a need for an automatic method of analysis of the sequences. One of the main issues associated with this problem is the problem of tracking objects in image sequences. For a human it is considered a simple task, but for machine it is rather a complicated process because of the need for the extraction of the data from the image.

Currently, algorithms for tracking objects of interest between the individual frames have reached a certain level of maturity allowing accurate tracking of the objects under the assumption that the objects does not change its shape and appearance. Such restrictions are not met in real scenarios therefore the existing algorithms for long-term tracking are disappointing. Changes in the appearance of the tracked object requires a certain way of updating detection module to the new conditions. Development of a new method for long-term objects tracking is motivated by the fact that less than one percent of the recorded surveillance video is ever watched [1,2]. The use of automated analysis of recorded material

L.J. Chmielewski et al. (Eds.): ICCVG 2014, LNCS 8671, pp. 302–315, 2014.

is particularly important in the crises situations, such as terrorist attacks. For example, a review of video material of bombings in Dubai in 2010 by a human lasted for several weeks, where the automatic analysis of materials of attack in Boston lasted only three days. The article presents a novel object tracking method which is computationally straightforward and performs in near real-time. It consists of three consecutive phases: detection, verification and learning in a way inspired by semi-supervised methods. The proposed approach and the developed algorithms were verified using a comprehensive set of prepared test sequences consisting of both synthetic and real scenes.

2 Literature Review

Term "method for object tracking" is defined as any method aimed to estimate the trajectory of a moving object being tracked in a sequence of images. The task of the tracker is to assign consistent labels to the tracked object in a sequence of consecutive frames [3]. Object tracking is, however, complex, e.g. due to the following problems [4]: the loss of information caused by projection of the 3D world on a 2D one; noise in the images; complex motion of objects; loose or articulated nature of the shape of objects; partial or complete occlusion of the tracked object; complicated shape of objects; changes in scene illumination; time constraints related to the real-time processing.

Visual tracking is considered one of the fundamental problems in computer vision. It is used in e.g. vision surveillance, human-computer interactions, navigation of unmanned objects, or issues related to the expanded reality [5]. Some tracking applications assume that the tracked object is known in advance, which allows to use the knowledge during the process of designing the tracking method. However, majority of the methods allows to track any object determined during the algorithm work time [4].

Below are presented some of the object tracking methods considered as the state-of-the-art reference methods. One of the most popular algorithms for tracking of the visual features is the algorithm called the Lukas-Kanade Tracker (KLT) [6]. The algorithm allows tracking features between subsequent images of the sequence. KLT can be divided into two main phases: detection of features and tracking. Detection of characteristic points is usually implemented using the autocorrelation method, e.g. Harris corner detector. Localization of feature points is found by identifying for each of the points the translation vector that minimizes the difference between the measure computed within a rectangular window centered around analyzed in pixels.

TLD method [7] (*Tracking-Learning-Detection*) is able to unequivocal state whether the defined in the first frame of a sequence tracked object is within the cam-era view or not. TLD method assumes that the long-term tracking of objects should consist of three phases: tracking, learning and detection. Tracking is realized by the Median-flow-tracker [8]. The task of the detection is independent of the tracking. NCC was used for the purpose. The detector can commit two types of errors: false positive and false negative. The task of the learning element is the observation of the tracker and detector and estimation on the basis error of detection and generation of new training samples in order to reduce the impact of the identified errors in the future.

FRAGtrack algorithm [9] assumes that the tracked object is represented by multiple image fragments. Each fragment vote regarding the probable position and the scale level of the tracked object by comparing the histogram of their area to the histogram of the tracked object from the first frame. The approach based on voting allows to track during partial occlusion or changes in pose of the tracked object. The authors emphasize that the proposed method is characterized by the constant computational complexity regardless of the size of the object being tracked.

VTD tracking method [4] according to the authors allows to track objects at the same time changing the appearance and character of the movement. The solution assumes the division of tracking tasks in two stages: defining the model of observa-tion and tracking its movement. Sparse PCA is calculated on a set of basic patterns of motion and appearance features. Tracking is also composed of a number of tracking compound elements where each of them realize tracking of different type of object changes. Results returned by tracking elements are further combined into one by usage of IMCMC (*Interactive Markov Chain Monte Carlo*).

The authors in [10] note that the tracking methods based on detection are largely based on a classifier, which task is to distinguish an object from its background. Even small errors in tracking element can cause the erroneous determination of training samples of the classifier and in the result cause a drift of the solution. The authors present the solution where they use the method called MIL Track (*Multiple Instance Learning Tracking*) instead of the typical supervised learning.

3 SETh

Among many groups of different methods of tracking one of the most convenient for the user with simultaneously some of the best tracking results is the group of tracking by detection [11, 12]. Typically, the object of interest is visible in the frame for considerable amount of time. However, there is a high probability that in a non-zero time the object is outside the view of the camera. It is assumed that in the first frame of a sequence a rectangular area of interest for tracked object is selected and the aim of the tracking algorithm is to detect the object of interest in successive frames of a sequence or to specify that the object is not visible in the image. Stream processing is done frame by frame, and the process time can be infinitely long. Thus defined tracking is known as long-term tracking [7]. Long-term tracking is difficult due to, e.g. problem of determining whether an object is within the field of view of the camera. This problem belongs to the complex ones, as the tracked object at that time could change the position, orientation, or appearance, therefore its appearance known from the first frame of the sequence may become obsolete [13]. As another important problem we can identify resistance of the tracking algorithm to changes in a camera position, lighting conditions, partial and total occlusion and moving background and finally, reducing the time of processing. The long-term tracking is widely considered as a combination of two phases: tracking and detection [7].

The proposed algorithm is derived from the family of methods of tracking by de-tection, generalized by updating the model of both tracked object and its closest sur-roundings. The developed algorithm for long-term tracking, SETh, is based on many years of experience and researches of the author [14, 15, 16].

The algorithm is initialized, and then executed sequentially in three successive phases: detection, verification, and learning. SETh algorithm is used to determine the position of the object being tracked or unequivocal statement that the object is not visible in the image. The algorithm allows tracking in a manner inspired by the semi-supervised methods, i.e., the operator in the first frame to track of the sequence indi-cates the area of interest containing the object to be tracked. In subsequent frames of the sequence the task of the algorithm is to track the position of the object without any additional information.

Fig. 1. Algorithm overview schema

During the detection step the goal is to detect features in the image, and then assign them to the appropriate labels: object, background or indeterminate features. The result of this step are sets: Θ – features of the object, Ω – features of the background and Υ – unrecognized features so far.

An important element of the presented algorithm is a method for detection of features. The ideal detector is defined as possibly computationally simple method for finding the areas of the image possible to detect reproducibly regardless of the change in the point of view of the camera and at the same time resistant to all possible types of transformation. Currently, the closest to the prescribed requirements and with shortest computation time is BRISK detector [17]. Therefore, it was decided to use it as an element of the proposed long-term tracking algorithm. BRISK is insensitive to scaling and rotation due to the addition of local maxima search step not only in image space, but also the in the scale space.

Detected features have to be described by a descriptor that allows them to be uniquely compared. Description of the detected features should provide plenty uniqueness of the description, be computed efficiently and allows to timely and accurately compare the descriptor with a large set of data. All these advantages are met by FREAK descriptor described in [18]. FREAK was created based on the inspiration of information processes occurring in the human retina. FREAK descriptor is an efficient way to describe the feature by a cascade of binary string numbers calculated on the basis of differences in brightness in the area similar to the human retina sampling area. The sampling pattern which is used is circular and the points closer to the center have a higher density distribution. The density of occurrence decreases exponentially with the distance from the center of the feature which is described. Binary string of FREAK descriptor F (1) is a one-bit sequence coding differences in Gaussian function (DoG):

$$F = \sum_{0 \leqslant a \leqslant N} 2^a T(P_a), \tag{1}$$

$$T(P_a) = \begin{cases} 1, & (I(P_a^{r_1})) > 0, \\ 0, & otherwise, \end{cases} \tag{2}$$

where P_a is a pair of receptive fields, and N is the desired length of the descriptor, $I(P_a^{r_1})$ is a smooth function of a Gaussian brightness value of the first pair of reception field P_a. Combinations of several tens of pairs of fields result in the thousands of possible pairs of which 512 are selected by the decorrelation.

According to the scheme of the algorithm detected and described features have to be assigned to Θ, Ω and Υ sets on the basis of comparisons with the features of the object from the previous frame of the tracked object Θ' and the background Ω'.

The main objective of the verification phase is the selection of the correct position of the object, from the proposed by the stage of detection, and to determine the certainty level m_F. Simplified schema of the verification phase is shown in fig. 2.

Features belonging to a group with confidence level above the thresholds γ_1 and γ_2 are passed on to the stage of learning which update the model for binary classification of features between the object and the background. If there were not enough features detected during the detection phase, an alternative calculations are made in order to face the problem.

The detected features are labeled as object contained in the set Θ may indicate multiple localizations of the object being tracked. The observed scene may contain more than one object identical to the tracked. In addition, the detected characteristic points in the face of noise can be detected incorrectly. For this reason, the detection and matching of characteristic points is insufficient to determine the correct position of the object.

It is therefore required to specify unknown number of areas in a way resistance to noise and outliers. It is assumed that incorrect matches are distant from each other in Euclidean distance sense. For the purpose an unsupervised learning method DBSCAN was used [19]. The following parameter values were chosen: $Min_{pts} = 3$, because a group of three points can be described by the minimal area rectangle; $E_{ps} = 61$, as suggested in a research paper of FLIR company [20], where measured the minimal number of pixels sufficient for recognition of a person from the distance of 45 meters. Discovered N groups g in a set Θ were labeled Θ_{tym}. The tracked object is labeled using Feret box with a center in a point $c(x, y)$ in a first frame of sequence, so in each successive frame the detected

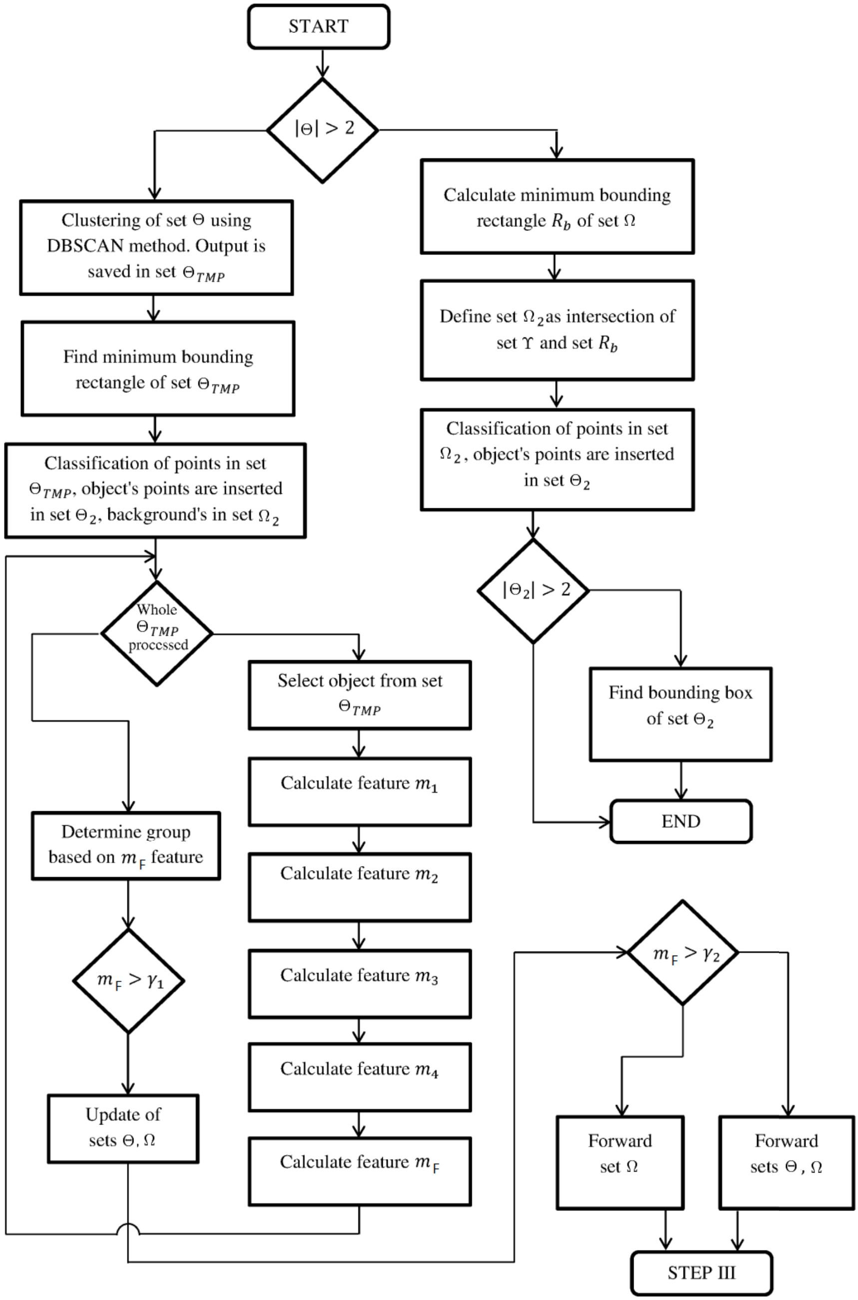

Fig. 2. Simplified schema of the verification stage of SETh long-term tracking method

area is also bounded by a rectangle. For each feature in each group g in the set Θ it is required to describe the minimal area rectangle r_i with a center in point $c_i(x, y)$. The problem of finding the minimal area rectangle bound given set of points has been solved using the SGPRC [21] method.

Analyzed during the verification phase of the algorithm areas potentially containing the tracked object P_{o_i} are defined on the basis of predefined groups g_i, describing them rectangles r_i, r_{ib}, detected and labeled background $P_{o_i}^{\Omega}$ and object features $P_{o_i}^{\Theta}$, as data structures consisting of the following elements: $P_{o_i} = \left\{r_i, r_{ib}, P_{o_i}^{\Theta}, P_{o_i}^{\Omega}, P_{o_i}^{\Theta_2}, P_{o_i}^{\Omega_2}\right\}$, where $P_{o_i}^{\Theta_2}$ and $P_{o_i}^{\Omega_2}$ are initially blank sets in which the classified features from the Υ set would be placed.

In order to attach the labels to the features from the Υ set a non-linear SVM classifier with RBF kernel function and $\gamma = 3$ is used. The classifier is trained in a supervised manner during the initialization phase. Features from the Υ are classified and distributed between features set $P_{o_i}^{\Theta_2}$ and background set $P_{o_i}^{\Omega_2}$. For each potential position of the tracked object a measure of area significance m_F is computed. The measure is a weighted average of four partial measures introduced in order to face the typical long-term tracking. Measure m_1 allows to determine the ratio of background matching for the evaluated P_{o_i}. The measure is calculated according to the formula (3). Measure m_2 determines the degree of the evaluated P_{o_i} similarity to the current model of the object being tracked (4). Measure m_3 allows the correct detection of the object being tracked in case of detection of multiple identical objects. It is based on the assumption of a certain continuity of motion and is expressed as the Euclidean distance (5). Measure m_4 determines the ratio of object features matching for the evaluated P_{o_i} (6).

$$m_1 = \frac{|P_{o_i}^{\Omega}| + ||P_{o_i}^{\Omega_2}|}{|\Omega| + T_{B2}}, \tag{3}$$

$$m_2 = \frac{|P_{o_i}^{\Theta_2}|}{T_{T2}}, \tag{4}$$

$$m_3 = 1 - \frac{\sqrt{(c_{ix} - c_x)^2 + (c_{iy} - c_y)^2}}{L}, \tag{5}$$

$$m_4 = \frac{|P_{o_i}^{\Theta}|}{|\Theta|}, \tag{6}$$

$$T_{B2} = \sum_{i=1}^{N} |P_{o_i}^{\Omega_2}|, \tag{7}$$

$$T_{T2} = \sum_{i=1}^{N} |P_{o_i}^{\Theta_2}|, \tag{8}$$

$$L = \sum_{i=1}^{N} \sqrt{(c_{ix} - x_x)^2 + (x_{iy} - c_y)^2}. \tag{9}$$

The values of all measurements were normalized to a closed interval from zero to one. For this purpose the following auxiliary variables (7, 8, 9) were introduced. The final measure m_F is calculated in accordance to the following formula: $m_F = \alpha m_1 + \beta m_2 + \gamma m_3 + \delta m_4$, where $\alpha, \beta, \gamma, \delta$ are the weight coefficients of each partial measure. In the study the following coefficient values were used: $\alpha = 0.3, \beta = 0.1, \gamma = 0.1, \delta = 0.5$. The values α and δ were as greater in order to emphasize the significance of the feature descriptors matching with respect to additional heuristics.

The tracked object is considered to be the potential object P_{o_i} characterized by the highest value of the final measure m_F. If the value of the final measurement is above the cut-off γ_1 we assume that one can update the sets of object Θ' and background Ω' features. Experimentally determined value $\gamma_1 = 0.3$. In addition, it is recognized that above this cut-off background features are further passed to the learning phase of the algorithm. The significance level m_F above the cut-off γ_2 is considered reliably and above it all the object and background features are further passed to the learning phase of the algorithm. Experimentally determined value of γ_2 is 0.7.

Alternatively, if there are no more than two features in the set Θ we suggest to conduct countermeasure based on the background features matching. The procedure is based on the observation that a moving object on the stage is not fully independent. Its presence affects the other elements of the scene by changing the reflection of light, generating shadows, shielding some area from rain, etc. Classically this phenomenon is seen as negative, hindering the process of object tracking. Here however, we assume that there is the tracked object visible in the image, however due to dynamic change of appearance the matching process didnt succeed but there is still a possibility that the object is within the best matched background area of interest. In order to point out the localization, we classify the features within the matched area and treat the result as a temporary good hint.

The long-term tracking requires updating a representation of the object being tracked in order to ensure its quality in the face of changes occurring in appearance of the object. Goal of the learning phase is to iteratively build a model representation of the tracked object based on consecutive frames of the video stream. Features within Θ' and Ω' sets selected during verification phase are used to teach nonlinear kernel SVM classifier with RBF kernel with γ selected as three.

4 Tests

An important problem from the point of view of the effectiveness of tracking algorithms is the method for evaluating the acquired results. There are applications of both the qualitative and quantitative analysis. The following measures were inspired by the known from the literature measures [7, 22, 23]. Used measures are based on the sequence length L, determined by the human expert reference ground truth labeled R_G, area of interest computed by the tracking method labeled R_W, the number of correct indications τ_P, such frames as the value $\epsilon_{OR} > 50\%$, the total number of indications labeled τ. There is used a measure of compliance called overlap ratio defined as the weighted ratio between the R_W and R_G areas called ε_{OR} (10), measure of the ratio of the number of correct indications to the length of the sequence is denoted by ε_{SR} (11), measure of the location error calculated as the Euclidean distance between the center of the R_W and R_G denoted by ε_{LO} (12), average error value computed as an arithmetic mean of localization error, labeled as ε_{ALO} (13).

$$\varepsilon_{OR} = 2\frac{|R_G \cap R_W|}{|R_G| + |R_W|}, \tag{10}$$

$$\varepsilon_{SR} = \frac{\tau_P}{L}, \tag{11}$$

$$\varepsilon_{LO} = sqrt(R_{Gx} - R_{Wx})^2 + (R_{Gy} - R_{Wy})^2, \tag{12}$$

$$\varepsilon_{ALO} = \frac{1}{L}\sum_{i=1}^{L} \varepsilon_{LO_i}. \tag{13}$$

Acquired results by the proposed long-term tracking method SETh were compared against reference methods. For comparisons the publicly available for research purposes implementations of the selected methods were used. There were five state-of-the-art reference methods selected for comparison: Lukas Kanade Tracker, labeled in results as KLT; Tracking-Learning-Detection, labeled in results as TLD; FRAGTrack, labeled in results as FRAG; Visual Tracking Decomposition, labeled in results as VTD; Multiple Instance Learning Tracking, labeled in results as MIL Track.

Due to the limitations of length of the article only selected representative test re-sults are presented from a set of total 102 test sequences. There have been hand annotated over $25'500$ frames. All sequences are documented and the data set has been made available to the public [24].

The calculations were performed on a personal computer with an Intel Core i5 2.4 GHz, 8GB RAM, graphics card NVIDIA GeForce GT 525 M and 64-bit operating system Windows 7 Professional.

The acquired results of comparison of tracking quality of SETh and reference state-of-the-art method are presented in fig. 3. It can be observed that presented SETh method was able to track the object during the whole sequence. It is worth to mention that other methods lost their target and had problems with tracking reinitialization. Computed measures (fig. 4) present that both the highest localization precision and the lowest average error were scored by presented SETh method. Second result is taken by VTD and surprisingly the last place was reserved for TLD method.

In the fig. 5. visual results of object tracking during car driving are presented. The camera was attached to the head of the driver (*glasses*). It can be observed that all tracking methods except the proposed SETh and TLD methods failed.

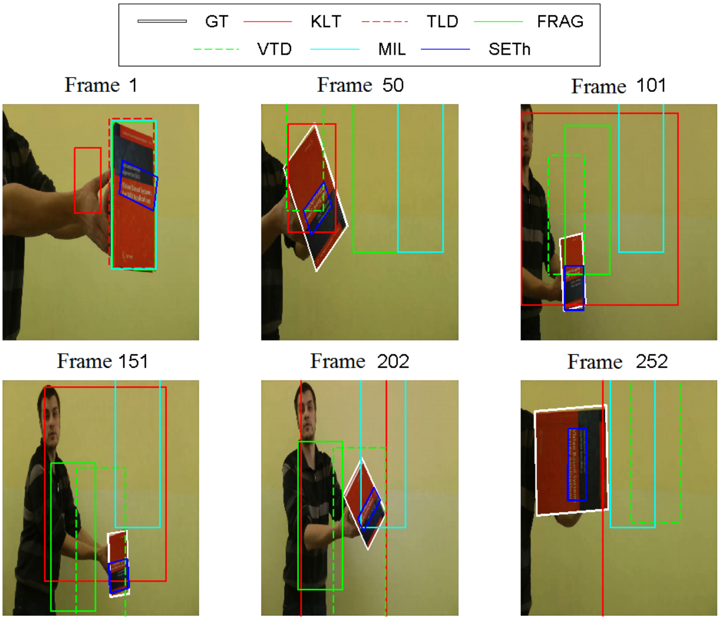

Fig. 3. Visual comparison of tracking results for rts sequence

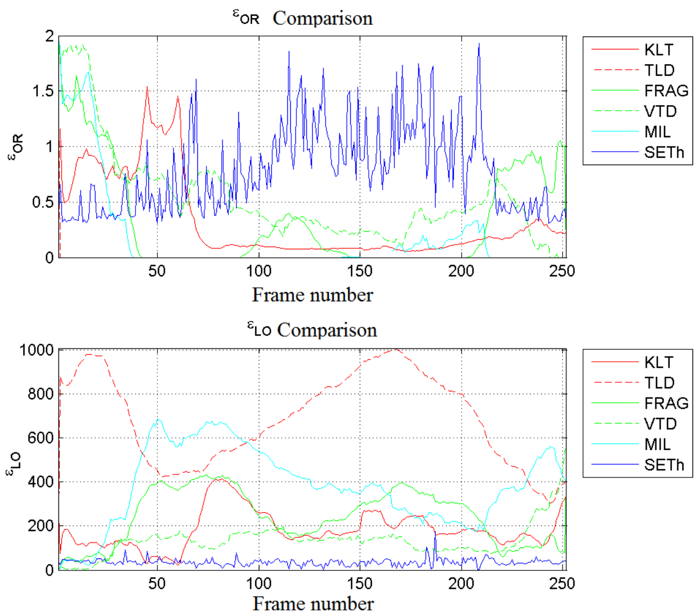

Fig. 4. Comparison of measures test sequences for rts sequence

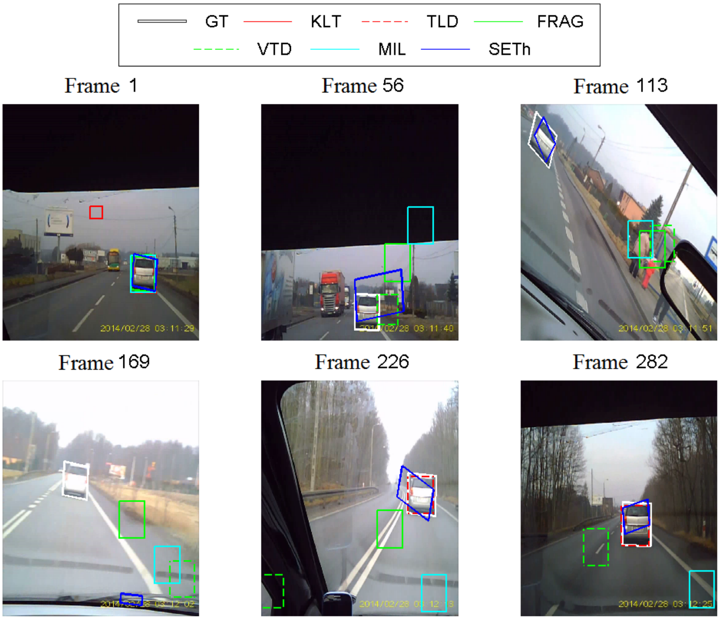

Fig. 5. Visual comparison of tracking results for car sequence

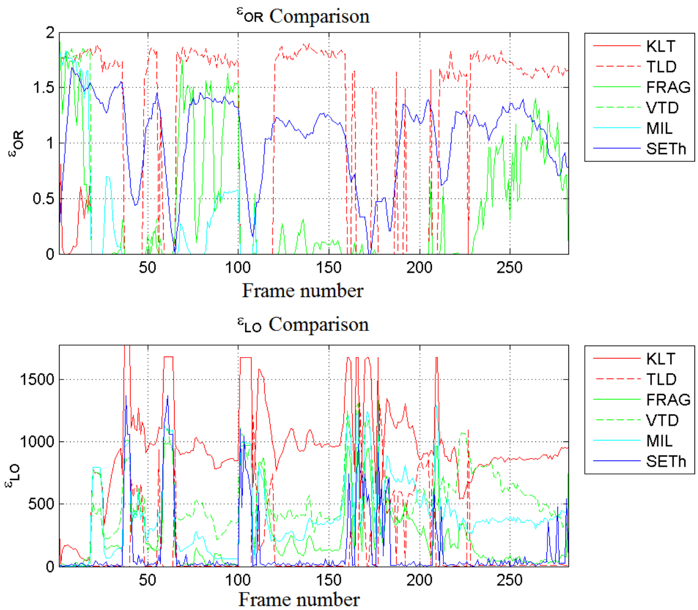

Fig. 6. Comparison of measures test sequences for car sequence.

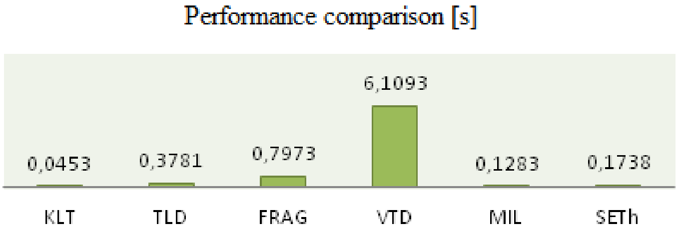

Fig. 7. Column performance comparison of the selected state-of-the-art tracking methods

It is worth to notice that the presented SETh method is invariant to rotation and scale. However, it is vulnerable to low contrast and visible in the frame 169. It can be observed that regardless the short error in tracking the proposed method was able to correctly reinitialize tracking.

Computed measures for the car sequence are presented in the fig. 6. It can be no-ticed that the best results as supposed were acquired by SETh and TLD methods, which are able to reinitialize tracking after failure.

The proposed SETh method was compared in terms of computation time against the five reference state-of-the-art methods. The computed average time of processing is presented in the fig. 7. KLT method was able to compute its output in real-time without any further optimizations. The rest of the methods achieved similar results in the range starting from 100 ms to 1000 ms. Indisputably the worst time of computa-tion, which was over 6000 ms was scored by VTD method. It is worth to mention that video sequences used for measurements were 1280x720 pixels.

5 Conclusions

The article presents a novel method for long-term object tracking named SETh. The solution performs in a near real-time consecutive phases: detection, verification and learning. The proposed tracking method begins with the detection of visual fea-tures using BRISK detector and describing them using FREAK descriptor. The described features are compared with a set of known features of the object of interest and then clustered by an unsupervised method in order to determine the amount and areas potentially containing the object. Groups of features labeled as object are used to determine the region of interest - the potential area of object which surroundings are compared against the features labeled as background. Utilization during tracking not only objects features but background features as well, allows to increase the quality of tracking of objects characterized by changeable appearance. The object or background labels are attached to unspecified labels within the potential object′s area using binary nonlinear SVM. For each potential object final measure consisting of four partial measures is computed. Final measurement value is double thresholded. The tracked object is assumed to be a potential object characterized by a highest value of final measure. The feature points within selected tracked object′s area and neighborhood is used to update the SVM classifier.

The proposed tracking method SETh was verified on a prepared comprehensive set consisting of both synthetic and real sequences. Reference value for tracking quality evaluation was annotated manually by human. Manually annotated were more than 25′500 frames. Prepared data set has been made available to the public. SETh tracking method was compared with five state-of-the-art methods and achieved comparable or superior results, suggesting that it is possible to apply it in real-life applications e.g. visual-based control of UAVs [25] or tracking pedestrians [26].

Acknowledgement. This work has been supported by Applied Research Programme of the National Centre for Research and Development as a project ID 178438 path A – Costume for acquisition of human movement based on IMU sensors with collection, visualization and data analysis software.

References

1. Pritch, A.Y., Ratovitch, S., Hendel, A., Peleg, S.: Clustered Synopsis of Surveillance Video. In: Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance. IEEE (2009)
2. Pritch, Y., Rav-Acha, A., Peleg, S.: Video Synopsis. The Hebrew University of Jerusalem
3. Yilmaz, A., Li, X., Shah, M.: Contour based object tracking with occlusion handling in video acquired using mobile cameras. IEEE Trans. Patt. Analy. Mach. Intell. 26(11), 1531–1536 (2004)
4. Kwon, J., Lee, K.M.: Visual tracking decomposition. In: Computer Vision and Pattern Recognition (CVPR) (2010)
5. Wu, A.: Towards Linear-Time Incremental Structure from Motion. In: IEEE International Conference on 3D Vision, pp. 127–134 (2013)
6. Tomasi, C., Kanade, T.: Detection and Tracking of Point Features. Carnegie Mellon University Technical Report CMU-CS-91-132 (1991)
7. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-learning-detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 34, 1409–1422 (2012)
8. Kalal, Z., MikolajczykK., M.J.: Forward-Backward Error: Automatic Detection of Tracking Failures. In: International Conference on Pattern Recognition, pp. 23–26 (2010)
9. Adam, A., Rivlin, E., Shimshoni, I.: Robust fragments-based tracking using the integral histogram. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1 (2006)
10. Babenko, B., Yang, M.H., Belongie, S.: Robust object tracking with online multiple instance learning. IEEE Transactions on Pattern Analysis and Machine Intelligence 33(8), 1619–1632 (2011)
11. Breitenstein, M.D., Reichlin, F., Leibe, B., Koller-Meier, E., Gool, L.V.: Robust Tracking-by-Detection Using a Detector Confidence Particle Filter. In: Proc. IEEE 12th Intl Conf. Computer Vision (2009)
12. Grabner, H., Bischof, H.: On-line boosting and vision. In: CVPR (2006)

13. Grabner, H., Leistner, C., Bischof, H.: Semi-Supervised On-Line Boosting for Robust Tracking. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 234–247. Springer, Heidelberg (2008)
14. Jedrasiak, K., Nawrat, A.: Fast color recognition algorithm for robotics. Problemy Eksploatacji, Maintenance Problems (Quarterly), 69–76 (2008)
15. Nawrat, A., Jedrasiak, K.: SETh system spatio-temporal object tracking using combined color and motion feature. Advanced robotics, control and advanced manufacturing systems. In: Proceedings of the 9th WSEAS International Conference on Robotics, Control and Manufacturing Technology (ROCOM 2009), pp. 67–72 (2009)
16. Jędrasiak, K., Nawrat, A.: Image recognition technique for unmanned aerial vehicles. In: Bolc, L., Kulikowski, J.L., Wojciechowski, K. (eds.) ICCVG 2008. LNCS, vol. 5337, pp. 391–399. Springer, Heidelberg (2009)
17. Leutenegger, S., Chli, M., Siegwart, R.: Brisk: Binary robust invariant scalable keypoints. In: ICCV 2011, pp. 2548–2555 (2010)
18. Alahi, A., Ortiz, R., Vandergheynst, P.: FREAK: Fast Retina Keypoint. In: IEEE Conference on Computer Vision and Pattern Recognition (2012)
19. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD 1996), pp. 226–231 (1996)
20. FLIR, `http://www.flir.com/PL` (access on March 5, 2014)
21. Toussaint, G.T.: Solving geometric problems with the rotating calipers. Proc. IEEE Melecon 83, A10 (1983)
22. Oron, S., Bar-Hillel, A., Levi, D., Avidan, S.: Locally Orderless Tracking. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2012)
23. Morzy, T.: Eksploracja Danych Metody i Algorytmy. Wydawnictwo Naukowe PWN, Warszawa (2013)
24. Results, `https://www.dropbox.com/sh/v705ac32e8006z7/s1ezUqplcE`
25. Iwaneczko, P., Jędrasiak, K., Daniec, K., Nawrat, A.: A Prototype of Unmanned Aerial Vehicle for Image Acquisition. In: Bolc, L., Tadeusiewicz, R., Chmielewski, L.J., Wojciechowski, K. (eds.) ICCVG 2012. LNCS, vol. 7594, pp. 87–94. Springer, Heidelberg (2012)
26. Jędrasiak, K., Nawrat, A., Daniec, K., Koteras, R., Mikulski, M., Grzejszczak, T.: A Prototype Device for Concealed Weapon Detection Using IR and CMOS Cameras Fast Image Fusion. In: Bolc, L., Tadeusiewicz, R., Chmielewski, L.J., Wojciechowski, K. (eds.) ICCVG 2012. LNCS, vol. 7594, pp. 423–432. Springer, Heidelberg (2012)