IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 22 May 2023, accepted 17 June 2023, date of publication 22 June 2023, date of current version 28 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3288694

RESEARCH ARTICLE

Object Tracking in SWIR Imaging Based on Both Correlation and Robust Kalman Filters

MILOŠ PAVLOVIĆ[1,2], (Member, IEEE), ZORAN BANJAC[2], (Member, IEEE), AND BRANKO KOVAČEVIĆ[1], (Senior Member, IEEE)

[1]School of Electrical Engineering, University of Belgrade, 11120 Belgrade, Serbia
[2]Vlatacom Institute of High Technologies, 11070 Belgrade, Serbia

Corresponding author: Miloš Pavlović (milos.pavlovic@vlatacom.com)

This work was supported by the Vlatacom Institute of High Technologies under Project 157VMSIS3_Advanced.

ABSTRACT Short-wave infrared (SWIR) imaging has significant advantages in challenging propagation conditions where the effectiveness of visible-light and thermal imaging is limited. Object tracking in SWIR imaging is particularly difficult due to lack of color information, but also because of occlusions and maneuvers of the tracked object. This paper proposes a new algorithm for object tracking in SWIR imaging, using a kernelized correlation filter (KCF) as a basic tracker. To overcome occlusions, the paper proposes the use of the Kalman filter as a predictor and a method to expand the object search area. Expanding the object search area helps in better re-detection of the object after occlusion, but also leads to the occasional appearance of errors in measurement data that can lead to object loss. These errors can be treated as outliers. To cope with outliers, Huber's M-robust approach is applied, so this paper proposes robustification of the Kalman filter by introducing a nonlinear Huber's influence function in the Kalman filter estimation step. However, robustness to outliers comes at the cost of reduced estimator efficiency. To make a balance between desired estimator efficiency and resistance to outliers, a new adaptive M-robustified Kalman filter is proposed. This is achieved by adjusting the saturation threshold of the influence function using the detection confidence information from the basic KCF tracker. Experimental results on the created dataset of SWIR video sequences indicate that the proposed algorithm achieves a better performance than state-of-the-art trackers in tracking the maneuvering object in the presence of occlusions.

INDEX TERMS Kalman filter, kernelized correlation filter, object tracking, robust estimation, SWIR imaging.

I. INTRODUCTION

Video tracking presents the process of estimating the location of a moving object using a camera [1], and finds various applications in surveillance imaging systems for outdoor environments. In addition to object tracking in the visible-light domain [2], [3], [4], [5], [6], [7], for successful tracking of objects in low-light conditions or total darkness, thermal cameras are often included in surveillance systems. For this reason, many algorithms have been developed for tracking objects in thermal images [8], [9], [10], [11].

However, in conditions where there is smoke, haze or fog, visible-light cameras tend to produce images with limited detail and information, and thus directly influence the performance of the object tracking algorithms. In these challenging conditions the possibility of effective view can be provided using short-wave infrared (SWIR) cameras, which provide images richer in details [12], [13]. SWIR includes the electromagnetic spectrum range between 1 and 3 microns in wavelength. In this spectral region light is predominantly reflected from the objects, similarly to visible range. Advantages of SWIR cameras over visible and near infrared (NIR) cameras are in situations such as: haze penetration, forest and oil fire penetration, maritime and ground target contrast and long-range identification [13]. Although thermal cameras can

The associate editor coordinating the review of this manuscript and approving it for publication was Gerardo Flores.

also provide view in conditions of haze, fog, or smoke, thermal cameras detect the presence of a warm object against a cool background. On the other hand, SWIR cameras can actually identify what that object is, and provide more information about the object in these challenging weather conditions. The reason is that thermal cameras do not provide the resolution and dynamic range of imaging possible with InGaAs (Indium Gallium Arsenide) technology in which SWIR focal plane arrays are most often made [12]. Moreover, the advantage of the SWIR cameras in comparison with thermal cameras is the ability to capture images through glass. Due to its advantages related to reduced scattering effects and spectral signatures, SWIR has found application in many civilian and military video surveillance systems.

Generally, object tracking methods can be classified into two categories: traditional and deep-learning-based approaches. In visual object tracking, as well as in thermal object tracking, deep-learning-based methods have gained significant attention in recent period. However, the application of deep learning methods in the SWIR domain is mostly focused on object detection [14], [15]. Deep-learning-based algorithms related to object tracking in SWIR video are applied in [16] and [17], where convolutional neural networks are used for multiple targets tracking in a degraded SWIR image with a significant percentage of missing data and bad pixels, called the compressive measurement domain. However, object tracking in the degraded SWIR image is out of the scope in this paper. Observed in the applications in the visible domain, deep learning-based methods for object tracking achieve incredibly good performance, but with the need for a large training dataset and unknown behavior in scenarios for which they have not been trained, that indicates a general drawback of deep learning-based approaches.

Traditional object tracking methods can also be divided into two categories: generative and discriminative algorithms. Generative algorithms search for the image regions which best match the target model, using only the information of the target. The goal of discriminative algorithms is to distinguish between the object and the background, using the information of both the target and its background. Performance of the generative algorithms is limited by the model representation space dimensions, and in more complex scene, they show less discrimination. On the other hand, discriminative algorithms, which utilize the background information of the target, have better ability to cover a wide range of changes in target appearance.

Although SWIR covers part of the infrared spectrum, light in SWIR domain is predominantly reflected from the objects, as in the visible range. Compared to thermal images, SWIR images have different characteristics, as the patterns in thermal images come from differences in the materials the objects are made of, as well as their different temperatures. The change in emitted radiation is a process that is slower than the change in reflected radiation. This means that changes in the thermal image occur more slowly than changes in the SWIR image, which results in different noise characteristics in SWIR and thermal images. Also, in the thermal images there are no shadows or different patterns depending on the colors (visual features) on the scene. So, analyzing the literature on real-time discriminative tracking algorithms primarily in the visible-light domain [2], [3], [4], [5], [6], [7], [18], it was found that the most often employed approaches are based on the correlation filters. Therefore, as a starting point for the development of algorithm for object tracking in SWIR imaging, a correlation filter-based algorithm is used: Kernelized Correlation Filter (KCF) [19].

Even though the KCF algorithm is generally effective in object tracking, it is not robust enough to deal with changes in object size and orientation, and especially with presence of various occlusions. Therefore, measurements of the tracked object's size and especially position are not always accurate. For applications in tracking systems where the camera is mounted on a pan-tilt platform, such as [20], the goal is to maintain the object in the center of the camera's field of view (FOV), and to achieve long-term tracking in a variety of conditions. So, there is a requirement for accurate and efficient prediction and estimation of object motion.

Starting from the limitations of the basic KCF algorithm, this paper firstly proposes improvements of the basic KCF algorithm in object size estimation and occlusion detection. Since occlusions have been identified as the most challenging problem, and especially a maneuver of the tracked object under occlusion, the paper further proposes using the Kalman filter as a predictor and expanding the object search area, in order to enable occlusion overcoming and object re-detection. Expanding the object search area helps in better re-detection of the tracked object after occlusion, but also leads to the appearance of errors in position measurement data that can lead to object loss and tracking termination. These occasionally large errors that lead to tracking termination present bad data or outliers [21]. For that reason, paper aims to investigate the characteristics of outliers in connection with object tracking in SWIR imaging. Statistical error analysis of the combination of the KCF with expanded search area and the Kalman filter can reveal common errors that impact the tracking performance, and errors that lead to the object loss. However, the standard Kalman filter in the presence of outliers in measurement data is not the optimal solution since it is sensitive to them.

Robust statistical methods provide tools to cope with outliers. Particularly, the Huber's approximate maximum likelihood (ML) robust approach, called the M-robust approach, is the most frequently used in engineering practice, because it is motivated by the optimal ML estimator that makes it more natural. Thus, the M-robust criterion, as a generator of a class of robust algorithms, has to approximate the optimal ML criterion so to achieve the insensitivity to outliers [22]. In general, the Huber's M-robust estimator requires rather easy computation with good convergence characteristics. In this sense, to avoid the complex methods that combine several different target appearance modeling techniques [23], [24], [25], [26], and to reduce the processing time, a new feasible approach to the SWIR object tracking based on a combination of the KCF tracker and the M-robustified version of the standard Kalman filter has been proposed in this paper. The M-robustified Kalman filter is derived by applying the Huber's M-robust approach to redesign the estimation step in the predictor-corrector structure of the standard Kalman filter. This, in turn, results in an approximate minimum variance nonlinear recursive state estimator, using the Huber's saturation type nonlinear influence function in the Kalman filter estimation step. Moreover, a new adaptive version of the M-robustified Kalman filter is also designed, by using the information from the basic KCF algorithm to adapt the saturation threshold of the influence function, so that the robust tracking algorithm makes a better balance between the tracking of object maneuver and the resistance to occurrence of outliers.

The rest of this paper is organized in the following manner: Section II reviews the basic KCF tracker and the related work about improvements of the KCF. Section III de-scribes the created dataset for SWIR object tracking, proposes the improved tracking method based on the KCF, and provides the statistical analysis of the proposed method results in SWIR object tracking. In Section IV, the M-robustified Kalman filter is introduced, and its adaptive version, based on the fitting of the influence function parameters. In Section V, the object tracking experiments are described, the results are presented, and a detailed discussion is given. The conclusion is presented in Section VI.

II. RELATED WORK IN VIDEO TRACKING BASED ON THE KCF ALGORITHM

The KCF algorithm [19] converts the target tracking problem into solving a ridge regression problem. The algorithm starts by defining a bounding box around the target. An image patch, x, of the size $M \times N$, which is larger in size than the target bounding box, together with the all circular shifts of that patch, x_i, are used in training the classifier. With the goal to find a discrimination function in the form of the linear product $f(z) = w^T z$, the classifier is trained using the squared error loss function over samples, x_i, and their regression targets, y_i:

$$\min_{w} \sum\nolimits_i (f(x_i) - y_i)^2 + \lambda \|w\|^2 \tag{1}$$

where λ is the regularization parameter that prevents overfitting.

Making the partial derivative of the optimization criterion in (1) with respect to w equal to zero, the obtained solution of the minimizer, w, has the closed-form as:

$$w = (X^T X + \lambda I)^{-1} X^T y \tag{2}$$

where X represents the sample matrix with the one sample x_i per row, and y is the vector whose elements present regression targets, y_i, while I represents the identity matrix.

Using the property that all circulant matrices are made diagonal by the Discrete Fourier Transform (DFT), the equation (2) can be converted into:

$$\hat{w} = \frac{\hat{x}^* \odot \hat{y}}{\hat{x}^* \odot \hat{x} + \lambda} \tag{3}$$

where $\hat{w}$, $\hat{x}$, and $\hat{y}$ represent the DFT of w, x, and y, respectively, while $\hat{x}^*$ is the complex-conjugate of $\hat{x}$. The fraction denotes element-wise division, and $\odot$ is the element-wise product.

To obtain more powerful nonlinear filter, a kernel trick is introduced. Input samples, x_i, are mapped to the high dimensional feature space (dual space), $\varphi(x)$, through the kernel function, and weight vector, w, at this point can be expressed as a linear combination of the samples:

$$w = \sum\nolimits_i \alpha_i \varphi(x_i) \tag{4}$$

In this way, the problem of finding the optimal parameter vector, w, is transformed into a problem where α is the alternative representation in dual space and variable under optimization. Therefore, the optimal solution α in dual space can be expressed as:

$$\alpha = (K + \lambda I)^{-1} y \tag{5}$$

where each element of the vector α represents the coefficient α_i, and K is the kernel matrix. Using kernels for which it is possible to make the matrix K circulant, and taking into account the characteristic of the circulant matrix in the Fourier domain, it is obtained:

$$\hat{\alpha} = \frac{\hat{y}}{\hat{k}^{xx} + \lambda} \tag{6}$$

where $\hat{k}^{xx}$ is the DFT of the first row of the kernel matrix K, named kernel correlation. Particularly, the Gaussian kernel, which is used further, is the one for which the matrix K is circulant, and for two arbitrary vectors x and x' is expressed as:

$$k^{xx'} = \exp\left(-\frac{1}{\sigma^2}\left(\|x\|^2 + \|x'\|^2 - 2F^{-1}\left(\hat{x}^* \odot \hat{x}'\right)\right)\right) \tag{7}$$

In the new frame of the video sequence, the image patch, z, of the same size $M \times N$ is extracted from the position of the object in the previous frame, and the regression function or detection response map is obtained as:

$$f(z) = F^{-1}\left(\hat{k}^{xz} \odot \hat{\alpha}\right) \tag{8}$$

where $\hat{k}^{xz}$ is the DFT of the kernel correlation of the tracked object model, x, and new image patch, z, while F^{-1} represents the inverse DFT. The position corresponding to the maximum value of the response map (pick value) presents the location of the tracked object in the new frame.

The basic KCF algorithm has very good performance in object tracking in conditions without occlusion, when the object is at a relatively constant distance from the camera, and has an impressive processing speed for real time operation.

However, when the size, orientation, and appearance of the object change, and especially under occlusion conditions, the performance of the basic KCF drops significantly.

By analyzing the problems in video tracking, as well as the available literature on the improvements of the KCF algorithm, several directions of improvement can be observed. Approaches that use color features in RGB or HSV domain [25], [27], [28], [29], or deep features [26], [30], [31] (obtained from deep neural networks trained on color images) to model the appearance of the object are not of interest in this research because the SWIR image lacks color information and is significantly different from the visible-light image. Other approaches in the recent literature that improve the KCF algorithm can be divided into four categories: scale estimation [26], [27], [32], [24], occlusion detection [24], [25], [28], [33], overcoming occlusions and object re-detection [34], [35] and application of predictors [31], [36], [37]. Application of these approaches to the SWIR object tracking requires upgrades of the basic KCF tracker, that are presented in the sequel.

III. SWIR OBJECT TRACKING ALGORITHM DESIGN

A. DATASET

To conduct a statistical analysis of the KCF algorithm in SWIR object tracking, the appropriate dataset is required. Publicly available datasets for visual object tracking, such as TrackingNet [38], LaSOT [39], VOT2022 [2], as well as datasets for thermal infrared object tracking: VOT-TIR16 [9], LTIR [40], PTB-TIR [10], LSOTB-TIR [11], contain a large number of video sequences recorded in different conditions, with different types of objects for tracking, and different challenges. These datasets provide the possibility of training deep learning models, and also of evaluating trackers in challenging scenarios. However, to the best of our knowledge, there are no publicly available datasets which contain SWIR video sequences specially created for the object tracking. Therefore, for the statistical analysis of the KCF tracker in SWIR imaging, it is necessary to create a new dataset.

This dataset should fulfill several conditions. First, a large number of frames in the dataset is required for proper statistical analysis. Also, it should contain a moving object of interest in various scenarios. These scenarios include tracking error causes such as: changes in object's motion dynamics, changes in orientation, size changes and, the most challenging, the presence of short-term and long-term occlusions. All scenarios should be recorded with the same camera and under the same conditions.

Dataset created for these purposes is very challenging in terms of images illustrating realistic scenes. It should provide a good basis for developing the best possible moving object tracking method for various scenarios, which can be found in real-life surveillance applications. As the most common in urban scenes, the chosen objects of interest for tracking and analysis are pedestrians. Also, the pedestrians represent typical objects with relatively slow and fast motion dynamics, which is also important in tracking analysis. A typical frame from database with the object of interest is shown in Fig. 1.

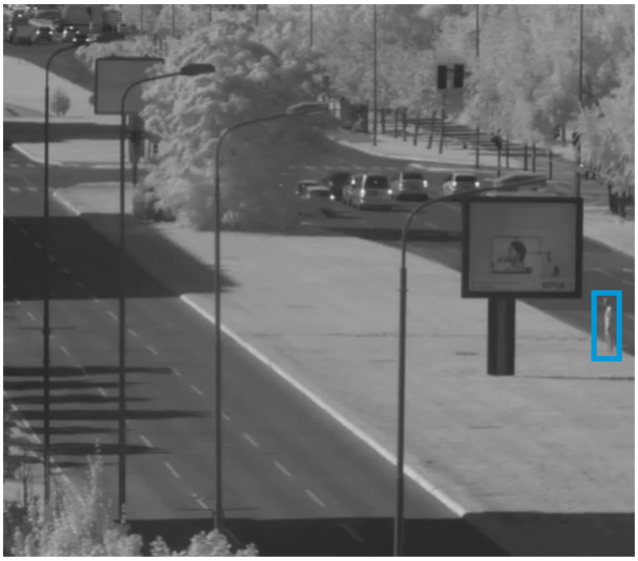


FIGURE 1. A typical frame from the SWIR dataset representing a scene with an object of interest and various types of occlusions. The object of interest is the pedestrian marked by the surrounding bounding box.

For the analysis, 9 video sequences are recorded with a moving person as the object of interest for tracking. Created dataset contains 4400 frames in total, being sufficient for de-tailed statistical analysis. Object of interest on each frame is manually labeled with the corresponding bounding box. It is assumed that the center of the bounding box is the position of the tracked object, and together with the corresponding bounding box width and height, represent the ground-truth data for further experiments. The video sequences are recorded using the SWIR camera, with the resolution of 576 × 504 pixels and 25 frames per second (FPS). The used camera is implemented in an interlaced technology, representing an additional challenge for tracking algorithm to extract the features from the image obtained in that technology. Although the created SWIR dataset has a smaller number of sequences than other infrared tracking datasets [9], [10], [11], it includes many challenging scenarios for detailed evaluation of the tracking algorithm. Table 1 shows the challenges in each dataset sequence. Sequences 1 - 4 were recorded with a fixed camera and sequences 5 - 11 were recorded with a moving camera. The size of objects is from 504 to 3240 pixels.

B. PROPOSED TRACKING METHOD

The goal of long-term tracking is to achieve continuous tracking of a moving object without additional manual corrections in situations of tracking failure. In order to achieve long-term tracking, the camera needs to be mounted on a pan-tilt positioner that will position the system [20], so that the object is constantly centered in the camera's FOV. In that case, it is important that the video tracker, which provides control

TABLE 1. Description of SWIR video sequences.

Sequence	Challenges
1.	short-term and long-term full static occlusions, orientation changes, size changes
2.	short-term partial and full static occlusions, changes in orientation and movement direction
3.	long-term full static occlusion, motion dynamics changes
4.	clutter, short-term occlusion, camera shaking
5.	short-term and long-term, partial and full static occlusions
6.	short-term partial and full static occlusions, orientation changes, motion dynamics changes
7.	one long-term full and several partial static occlusions, size changes
8.	partial and full static occlusions, several partial moving occlusions
9.	clutter, partial and full moving occlusions

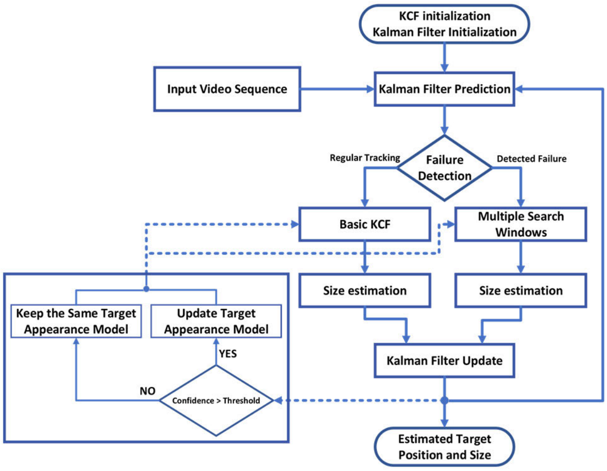


FIGURE 2. Flowchart of the proposed algorithm for SWIR object tracking.

signals to the pan-tilt positioner, successfully tracks the object of interest in the image, and overcomes the related video tracking problems such as occlusions, scale changes, changes in motion dynamics. Also, in addition to robust tracking performance, it is especially important that video tracking algorithm does not have high complexity, and can be executed in real-time.

Since the SWIR image characteristics significantly differ from those of a visible-light camera image, improvements of the basic KCF algorithm are mostly focused on the object motion model. Additional improvements are performed in object size estimation, tracking failure (occlusion) detection, overcoming occlusion and object re-detection. The proposed new method is shown in the flowchart in Fig. 2.

In the proposed method the basic KCF is synchronized with the Kalman filter to improve the object tracking performance. In the initialization step, on the first video sequence frame, the Kalman filter is initialized with the same data (object position and size) as the KFC algorithm. On each subsequent frame, the overall tracking method relies on the Kalman filter object state prediction. The center of the object search window is determined by the Kalman filter's object position prediction, with the width and height of the search window being 2.5 times that of the object. Predicted position by the Kalman filter is evaluated in the failure detection block. The main purpose of this block is to detect the tracking failure caused by tracking drift, object deformation and, the most important, the presence of occlusions. Also, the failure detection block plays an important role in switching between the basic KCF tracker and the multiple search windows. If the tracking failure is not detected, the basic KCF is applied, primarily due to the processing speed. The histogram of oriented gradient (HOG) features are extracted and the corresponding response map (8) is calculated. The position of the maximum value in the response map (pick value) presents the estimated object position by the basic KCF tracker. The basic KCF is followed by a size estimation block for the best possible estimation of the object size.

On the other hand, if the failure is detected, it is technically impossible to keep tracking if the object completely disappears under full occlusion or become out of view. In many practical cases, such as walking pedestrians, objects show constant movements (velocity) for a certain period, assuming that the object reappears after the occlusion. Therefore, the Kalman filter predicts the area for object re-detection, thus overcoming the problem of the basic KCF tracker, which gets stuck at the position of the first appearance of the occlusion. In addition, in some cases, even though the prediction algorithm is used, it may still be impossible to find the object, since the object can maneuver during the occlusion and, after the occlusion, it can be outside the predicted search area. Therefore, a multiple search windows are employed in combination with the Kalman filter. The search is needed only in a certain number of windows around the central one, because the Kalman filter predicts the dynamics of the object's movement, thus reducing the processing time that would be required for the search on the entire image. Multiple search windows block is also followed with the size estimation block to estimate the best possible object size. Although this introduces an additional latency in the algorithm, switching between the basic KCF and the multiple search window block which is applied only when the occlusion is detected, the algorithm still has the real-time execution on average. The detected position and the estimated size of the object, from the basic KCF or multiple search windows branch, are used in the update step of the Kalman filter. The output of the Kalman filter update block is an estimate of the position and the size of the tracked object in the current frame. This guarantees that the tracker will not be stopped at the point of occurrence of the occlusion. Thus, the dynamics of the moving object will be tracked, and the control signals will position the pan-tilt positioner so that the object remains in the camera field of view after occlusion.

During the object re-detection process, the target appearance model is not updated. As it is shown in Fig. 2, the confidence value of the final estimation is used in deciding whether to update the target appearance model. A confidence value higher than the defined threshold denotes that the detection of the object on the current frame is reliable, the object is not under occlusion, and the target appearance model can be updated with correct features. In the following sections, object motion model, modules for size estimation, failure detection, and object re-detection will be described in more details.

1) STATE ESTIMATION

In addition to tracking the object position, for successful tracking of the object, the estimation of the object size is also important. In order to approximate the inter-frame displacements in the position of the object, as well as in the size of the object, a linear constant velocity (CV) model is adopted [41]. The state of the object is modelled, as in [42]:

$$X = \begin{bmatrix} x_c \\ y_c \\ s \\ r \\ \dot{x}_c \\ \dot{y}_c \\ \dot{s} \end{bmatrix} \tag{9}$$

where x_c and y_c represent the pixel location of the object center in the image plane, in horizontal and vertical direction, respectively. The object's bounding box area is represented with s, while r is the aspect ratio (ratio of the width and height of the bounding box: w/h). Moreover, the last three components of the state vector in (9) represent the first derivatives, or velocities, of the first three components of the state vector, where a state vector component and its velocity are linked by the CV model. The model defined in this way provides the possibility of both the position estimation and the object size estimation in regular conditions, as well as during movement under occlusions.

2) OBJECT SIZE ESTIMATION

In video tracking, estimating the size change of an object has a significant influence on the tracking performance. A change in the relative size of the object in the image plane is caused by the movement of the object in the scene, being closer or further away from the camera. The KCF is not able to deal with the size changes. In the KCF tracker, the size of the tracked object is constant, and the size of the bounding box is the same as it was on the first frame of the video sequence. With the fixed size of the object bounding box, the extracted features will be incomplete if the size of the object increases. On the other hand, if the object size decreases, variable background features will be introduced in the object appearance model.

In order to handle the size variations, on the current frame of the video sequence, the Kalman filter object size prediction is firstly used. The state vector defined in (9) provides the possibility of object size estimation, as well as the estimation during the movement of the tracked object under occlusion, which is especially important for object re-detection with the correct size after occlusion. As the kernel correlation function needs the data with the same dimensions, a scaling set $S = \{-10\ \%, -5\ \%, +5\ \%, +10\ \%\}$ is defined. Beside extracted patch based on the Kalman filter object size prediction, four more patches are extracted. These patches are centered at the position predicted by the Kalman filter, and with the sizes being relative to the size predicted by the Kalman filter according to the scale percentages in the set, S. Patches are resized to the initial object size (size defined on the first frame), after which the features are extracted. The size of the patch with the highest confidence is used as the current estimation of the object size, and is also applied in the Kalman filter update step. The procedure is shown in Fig. 3.

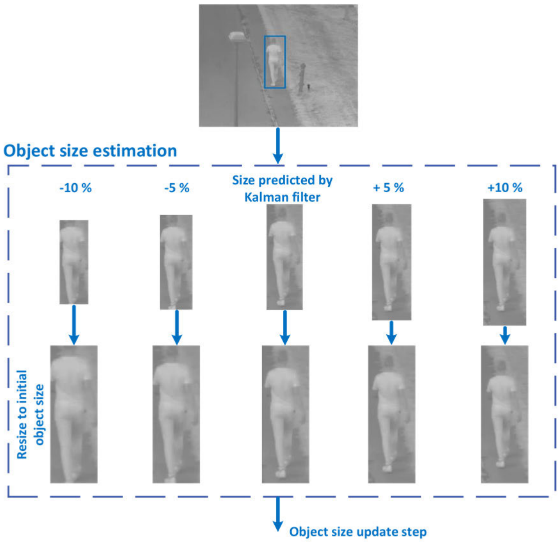


FIGURE 3. Object size estimation scheme.

3) FAILURE DETECTION

The KCF tracker is unable to detect the tracking failure and the situations of the object loss. When the object is lost, the algorithm will still track the background as the object of interest. Even the confidence is incredibly low, the target model will be updated, which results in tracking failure. Estimation of the target position on the current frame of the video sequence is based on the maximum value of the kernel correlation response map between the target and the reference, defined in (8). Obtained response map can be used for detection of the tracking failure.

In this paper, as a parameter for tracking failure detection is used PSR (Peak to Side-lobe Ratio). The PSR is calculated as follows:

$$PSR = \frac{f_{max} - \mu}{\sigma} \tag{10}$$

where f_{max} represents the peak value of the correlation response map, μ is the mean value of the sidelobe and σ is the standard deviation of the sidelobe. Sidelobe is defined as response map outside the 10 × 10 pixels region around the peak value.

The PSR value plays an important role in switching between the basic KCF tracker and the multiple search windows prediction branch when the occlusions occur, as it shown in Fig. 2. When the object is under occlusion, the PSR value will drop rapidly. A PSR value below a defined threshold indicates that the object is under the occlusion and multiple search windows should be activated. With the PSR value above the threshold, tracking can be performed with the basic KCF algorithm.

4) MULTIPLE SEARCH WINDOWS

Upon detecting a tracking failure, object searching in an extended area is activated using multiple search windows. An illustrative example is shown in Fig. 4. Relying on the Kalman filter prediction and a single search window, the object can easily be lost under full occlusion. If the object maneuvers behind the occlusion, with only one search window used, an accurate re-detection of a disappeared object after the occlusion is practically impossible. By searching in an extended zone around a fixed position where the object disappeared, and in a case of long-term occlusion, the re-detection can also be unreliable. With the Kalman filter prediction of the object's movement, and deploying multiple search windows around the predicted object position, this problem can be solved efficiently. The Kalman filter will estimate the dynamics of the object's movement, while the multiple search windows will enable to capture the object's maneuver during occlusion, as well as the object re-detection after the occlusion.

As shown in Fig. 4, the central search window is the one predicted by the Kalman filter. The other 8 search windows are of the same size as the central window and are positioned around the central window. To solve the problem arising when the object is in neighboring windows at the same time, and to improve detection, the windows overlapping is introduced. The overlap between neighboring windows is adopted to be 1/3 of the window area.

The response map and the peak value are generated from each search window. Also, the size estimation is applied for each search window. The detection of the object position on the current frame is based on the maximum peak value of all windows peak values. That position and estimated object size are further used in the Kalman filter update step.

C. STATISTICAL ANALYSIS OF THE PROPOSED ALGORITHM RESULTS IN SWIR OBJECT TRACKING

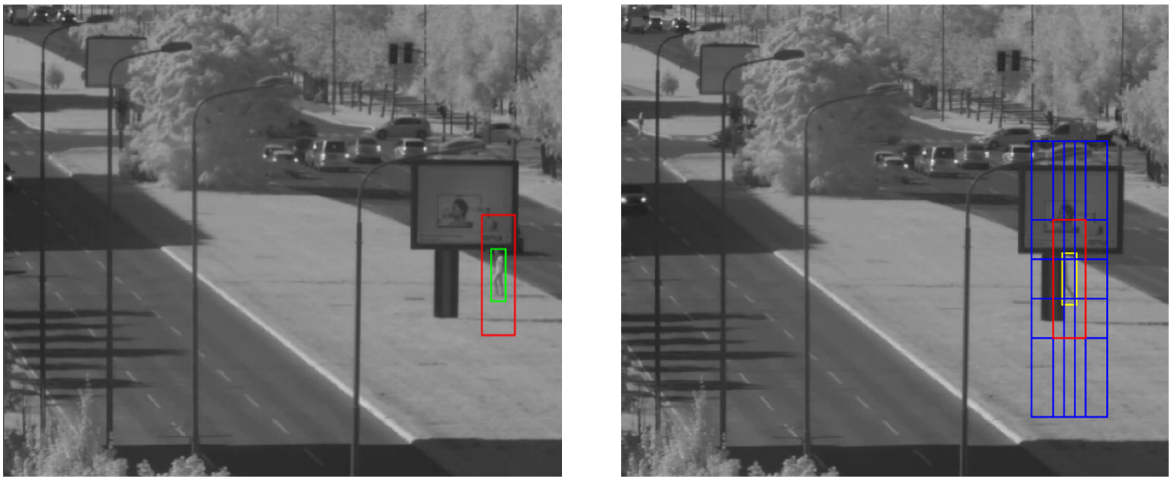
In order to statistically analyze the performance of the proposed method, created dataset of 4400 labeled SWIR frames is used. Algorithm behavior in various tracking scenarios is examined, with a walking pedestrian as object of interest. The ground-truth data are represented by the manually labeled position of the moving object center and the size of the object (height and width). Error in the object position and the object size error are analyzed in the sequel. For each frame, the position error is calculated as a difference between the ground-truth position and that estimated one by the proposed tracking method. The size error is divided into two categories: the height error and the width error, also measured as a difference between the actual height (width) and that estimated by the proposed tracking method. To analyze the error of the proposed algorithm on the entire dataset, both in regular conditions where tracking is successful and in the challenging ones, the algorithm was reinitialized every time after the occurrence of errors leading to the object loss and termination of tracking.

Fig. 5(a) shows the proposed method position errors for the vertical (y coordinate) and the horizontal (x coordinate) direction, together. In Fig. 5(b) are presented object size estimation errors, both the object width errors (horizontal axis) and the object height errors (vertical axis). It can be seen that the most of the size error population, both in width and height, is located within a single cluster. Also, most of the position error population belongs to the single cluster. However, from Fig. 5(a), it can be seen the presence of errors in the position, deviating significantly from the majority of the population in the cluster. This represents bad data or outliers.

In Fig. 6(a) is shown position error histogram. The position error is calculated as the Euclidean distance between the ground-truth object position and object position estimated by the proposed tracking method. To emphasize errors that lead to object loss and tracking termination, the so-called outliers, the histogram is shown in the log scale. Thus, the outliers can be seen distinctly in the histogram as tails with remarkably high error values.

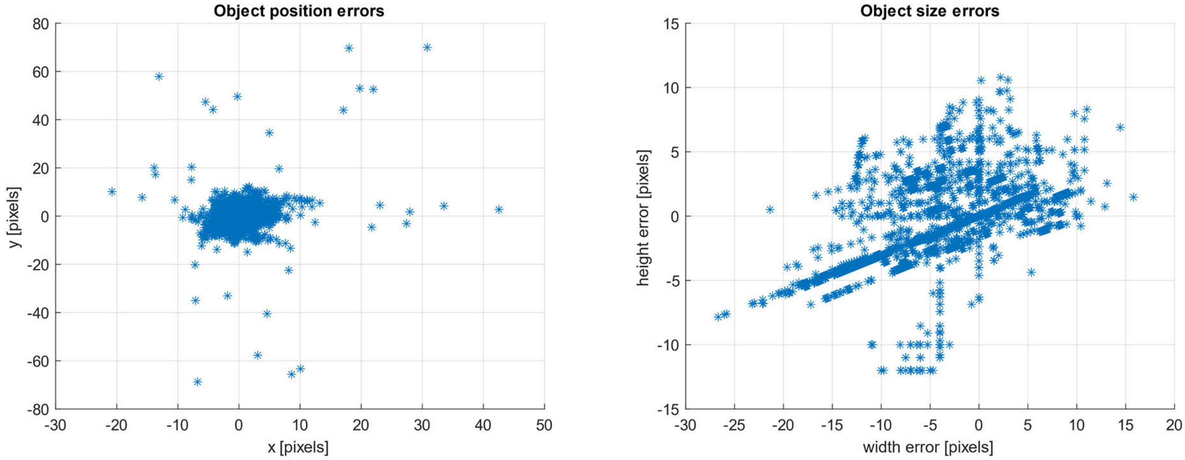
The number of these errors is not too high, but their values are extremely large compared to the rest of the population, which leads to the complete loss of the object and the termination of tracking. In accordance with Fig. 6(a), in the histogram distribution representation, the majority of the population indicating bad measurements that deviate from the ground-truth position, increasing the tracking variance, but not leading to the termination of tracking. The nature of these errors comes from the changes in the appearance of the object during tracking, changes in the dynamics of movement, as well as changes in both the direction of movement and the orientation of object. For the same reasons, an error occurs in the estimation of the object size (width and height), but the distribution of this error is not contaminated with outliers, which can be seen from the histograms in Fig. 6(b) and Fig. 6(c).

Analyzing the dataset and the situations in which position error outliers appear, it can be concluded that outliers are the result of partial or complete short-term and long-term occlusions. The basic KCF tracker, in situations where occlusions occur, will get stuck at the positions of occlusions, and will not be able to track the object further. Adding a predictor, such as the standard Kalman filter, will allow object tracking to continue even after occlusion, but only if the object does not maneuver or change movement dynamics during the occlusion. With the multiple search windows in combination



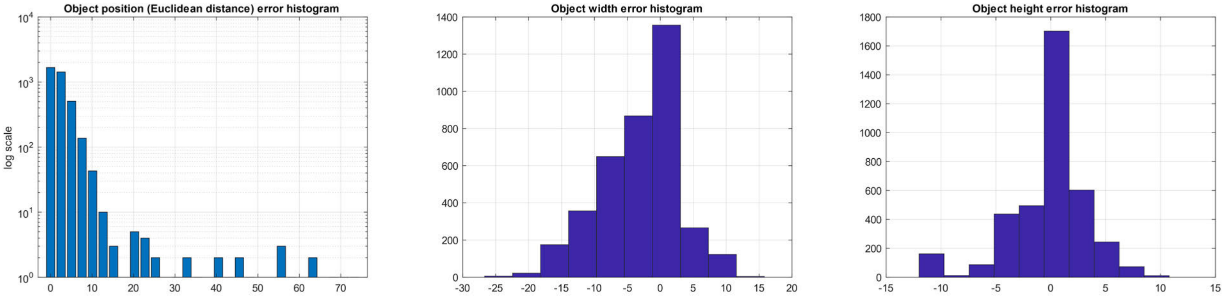
(a) (b)

FIGURE 4. **a) Object tracking in the regular conditions using a single search window (- - - tracked object, - - - search window) b) Multiple search windows are activated after the failure is detected (- - - estimation of tracked object, - - - central search window predicted by the Kalman filter, - - - surrounding 8 overlapped multiple search windows).**



(a) (b)

FIGURE 5. **Proposed method errors: a) position errors (in horizontal – x, and vertical – y direction), b) object size errors (width and height errors).**



(a) (b) (c)

FIGURE 6. **Histogram of the proposed method error: a) position error (Euclidean distance) in log scale, b) object width error, c) object height error.**

with the standard Kalman filter predictor, it is possible to overcome occlusions and capture potential object maneuvers. However, the multiple search windows provide a larger area where the object can be found, but also occasionally lead to the appearance of large position errors (outliers). In the event of a large error in position, in real-life applications the system that is on the pan-tilt positioner will bring the center of its field of view to the wrong position and thus permanently lose the object, as well as the area in which the object could potentially be found.

Then, the main question is how to distinguish between errors caused by changes in the object's appearance and the object dynamics, due to maneuvers, and those caused by the incorrect measurements, due to outliers, and make a balance between efficiency and robustness of the designed predictor.

As discussed in Section II, the standard Kalman filter is the most commonly used predictor. However, the standard Kalman filter is an optimal solution only under certain assumptions. One of these assumptions is that the measurement noise follows the Gaussian distribution. However, in real-life tracking scenarios, due to the presence of outliers, measurement noise may be confined to a non-Gaussian distribution. Therefore, one of the basic assumptions of the standard Kalman filter is not satisfied. Taking into account that the real measurements are contaminated by outliers, and that the standard Kalman filter is sensitive to data whose distribution is not Gaussian, a robust Kalman filter was designed and applied in the proposed tracking method.

IV. ROBUST TRACKING SYSTEM DESIGN

One of the most significant contributions to estimation theory is the Kalman filter. Recursive predictor-corrector structure of the Kalman filter, and its simplicity, makes it very attractive for various real-time tracking applications. Standard Kalman filter obtains the optimal performance when an adequate description of system state dynamics is provided, and the distribution of noise in observed data is the Gaussian one [43]. Since the probability density function (PDF) of the measurement noise frequently deviates from the Gaussian one in real applications, the standard Kalman filter is not a robust solution for object tracking in these conditions [44]. Particularly, a real observation noise PDF in many various applications can be represented as a heavy-tailed Gaussian PDF, being a zero-mean Gaussian PDF in the middle, but with heavier tails than the Gaussian one corresponding, for example, to the Laplace PDF [22]. This, in turn, generates a small percentage of outliers contaminating the mainly Gaussian observations. Such PDF is called contaminated Gaussian one, where the contaminating PDF is a zero-mean symmetric with greater variance than the basic Gaussian one. Observing statistical analysis of numerous measurements, the contamination degree is as a rule from 0.05 to 0.1, corresponding from 5 to 10 % of outliers [21]. Even a single outlier in measurement data can have a huge impact on the standard Kalman filter performance. Therefore, there is an additional practical interest in designing a robust filtering technique. Using statistical robust estimation theory, the effect of outliers in the mainly Gaussian observations can be minimized. Thus, the robust Kalman filter has to give approximately the same results as the standard method, if data do not contain outliers. On the other hand, in situations with a small or moderate percentage of outliers, the robust method has to reach significantly better performance. The first property is known as the efficiency robustness, while the second one is called the resistant robustness [22]. In this sense, a robust version of the Kalman filter has to satisfy both the efficiency and the resistant robustness, making the practical robustness goals.

A. ROBUST KALMAN FILTER

The system of discrete control processes is introduced and given by the state-space model:

$$x_{k+1} = F_k x_k + G_k w_k \tag{11}$$

$$y_k = H_k x_k + v_k \tag{12}$$

Here, x_k is the random n-dimensional state vector, y_k is the observation or measurement m-dimensional vector, w_k is the state noise l-dimensional vector, and v_k is the additive measurement noise m-dimensional vector, at the discrete time step indexed by k. Moreover, F_k represents the $n \times n$ dimensional state-transition matrix, G_k is the state-noise or disturbance $n \times l$ dimensional matrix, and H_k is the $m \times n$ dimensional measurement or observation matrix.

Furthermore, the noise sequences, w_k and v_k are zero-mean and assumed to be uncorrelated by itself and mutually, yielding:

$$E\left\{\begin{bmatrix} w_k \\ v_k \end{bmatrix}\begin{bmatrix} w_j \\ v_j \end{bmatrix}^T\right\} = diag\left\{Q_k\delta_{kj}, R_k\delta_{kj}\right\} \tag{13}$$

where $E\{\cdot\}$ is the mathematical expectation, $diag\{\cdot\}$ represents the block-diagonal matrix, and δ_{kj} denotes the Kronecker's delta symbol ($\delta_{kj} = 0$ if $k \neq j$, and $\delta_{kk} = 1$). Also, Q_k and R_k present the given positive semidefinite covariance matrices of the state noise, w_k, and the observation noise, v_k, respectively.

If the $\hat{x}_{k|k-1}$ is the linear one step optimal prediction of the present state, x_k, in the minimal mean-square sense (MMSE), while $P_{k|k-1}$ denotes the corresponding prediction error covariance matrix, then the standard Kalman filter equations are the following [43], [45]:

1. Prediction step (time update)

$$\hat{x}_{k|k-1} = F_{k-1}\hat{x}_{k-1|k-1} \tag{14}$$

$$P_{k|k-1} = F_{k-1}P_{k-1|k-1}F_{k-1}^T + G_{k-1}Q_{k-1}G_{k-1}^T \tag{15}$$

2. Estimation step (measurement update)

$$\varepsilon_k = y_k - \hat{x}_{k|k-1} \tag{16}$$

$$K_k = P_{k|k-1}H_k^T\left[H_k P_{k|k-1}H_k^T + R_k\right]^{-1} \tag{17}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k\varepsilon_k \tag{18}$$

$$P_{k|k} = [I - K_k H_k]P_{k|k-1} \tag{19}$$

In (16) – (18), ε_k is the measurement residual or innovation, K is the Kalman gain, $P_{k|k}$ is the present estimation error covariance matrix, and I is the identity matrix.

It is assumed that the initial state, x_0, is the random vector uncorrelated with the future noises w_k and v_k, with zero-mean and the corresponding covariance matrix P_0. Thus, filter can be initialized with $\hat{x}_{0|0} = E\{x_0\} = m_0 = 0$, and $P_{0|0} = P_0$. Also, measurement residual ε_k represents the zero-mean uncorrelated random sequence with the covariance matrix S_k:

$$E\left\{\varepsilon_k \varepsilon_j^T\right\} = S_k \delta_{kj};\, S_k = H_k P_{k|k-1} H_k^T + R_k \tag{20}$$

Robustification of the Kalman filter can be performed by modifying the estimation step, using the Huber's M-robust approach [22].

The Huber's M-robust approach is defined as the minimization of the empirical average loss, by using the regression parametric model. To apply this approach to dynamic system state estimation, using the system state space representation (11) and (12), one has to replace the Huber's M-robust performance measure, being the empirical average loss, with the time varying functional:

$$J\left(\hat{x}_{k|k-1}\right) = E\left\{\rho\left(\frac{\varepsilon\left(\hat{x}_{k|k-1}\right)}{d_k}\right) \middle| \hat{x}_{k|k-1}, k\right\} \tag{21}$$

where $E\{\cdot|\cdot\}$ is the conditional expectation given the one step MMSE optimal prediction, $\hat{x}_{k|k-1}$, of the current system state vector, x_k, and the system output observations up to the present discrete time, k. Here, the M-robust score, or loss, function, ρ, has to be chosen so to cut off the outliers. Starting from the heavy-tailed Gaussian observation noise distribution, Huber has proposed the ρ-function to be a quadratic in the middle, and to increase more slowly than the quadratic one for the larger absolute values of the argument, such as the linear function, yielding:

$$\rho(x) = \begin{cases} \frac{x^2}{2} & ; |x| \leq \Delta \\ \Delta|x| - \frac{\Delta^2}{2} & ; |x| > \Delta \end{cases} \tag{22}$$

here, Δ is the tuning parameter that has to provide for the desired efficiency at the basic Gaussian noise model. It should be noted that the proposed ρ-function is the optimal ML function, being the negative natural logarithm of the heavy-tailed Gaussian PDF with the tails belonging to the Laplace PDF [22]. Additionally, such PDF is the worst case PDF, in the sense of minimal Fisher information, within a class of the contaminated Gaussian PDF's, where a zero-mean symmetric PDF with huge variance is the contaminating PDF. Moreover, the quantity, d_k, is a scaling factor, providing a scale-invariant state estimation, and analogously to (16), the random variable, ε_k, is the measurement residual or innovation.

So modified the Huber's M-robust performance index in (21) can be used as the generator for a class of stochastic gradient algorithms:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} - \Gamma_k g\left(\hat{x}_{k|k-1}\right) \tag{23}$$

$$g\left(\hat{x}_{k|k-1}\right) = -d_k^{-1} \psi\left(\frac{\varepsilon\left(\hat{x}_{k|k-1}\right)}{d_k}\right) H_k^T \tag{24}$$

where $g(\cdot)$ is the stochastic gradient of the adopted scalar deterministic M-robust criterion, J, and Γ is the weighting matrix, that influence the speed of convergence [44]. The stochastic gradient, g, represents a feasible approximation of the column gradient vector:

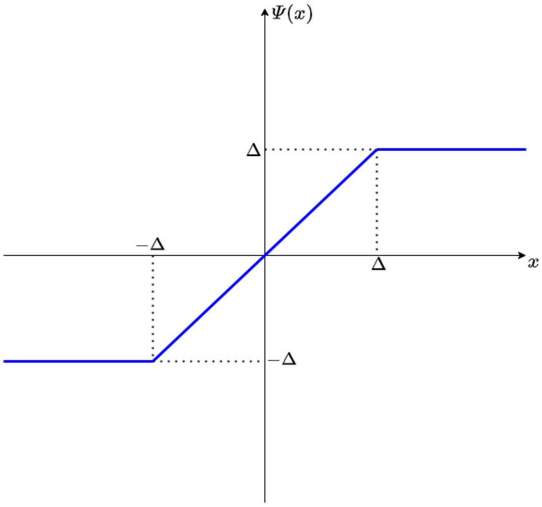


FIGURE 7. Huber's robust influence function.

$$\nabla J\left(\hat{x}_{k|k-1}\right) = \frac{\partial J\left(\hat{x}_{k|k-1}\right)}{\partial \hat{x}_{k|k-1}} \tag{25}$$

with $\partial(\cdot)/\partial(\cdot)$ being the partial derivative operator, obtained by replacing the unknown conditional expectation, $E\{\cdot|\cdot\}$, by the current sample. Moreover, the ψ-function, named the influence function, represents the first derivative of the score function, ρ. In general, it has to be bounded and continuous to suppress the influence of both a single outlier and a group of outliers [22]. Particularly, the Huber's M-robust influence function, ψ, is given by (26), and depicted in Fig. 7.

$$\psi(x) = \rho'(x) = \begin{cases} x & ; |x| \leq \Delta \\ \Delta sgn(x) & ; |x| > \Delta \end{cases} \tag{26}$$

Starting from the requirements for fast tracking performance, the weighting matrix, Γ, can be calculated at each step, k, by minimizing an additional criterion of approximate minimum variance type:

$$\min_{\Gamma_k} \; TraceP_{k|k} \tag{27}$$

$$P_{k|k} = E\left\{(x_k - \hat{x}_{k|k})(x_k - \hat{x}_{k|k})^T\right\} \tag{28}$$

The posed optimization problem is nonlinear, due to a nonlinear form of the robust estimate, $\hat{x}_{k|k}$, and an approximate optimal solution can be obtained by convenient simplifications [44]:

$$\Gamma_k = P_{k|k-1} \tag{29}$$

$$P_{k|k-1} = E\left\{\left(x_k - \hat{x}_{k|k-1}\right)\left(x_k - \hat{x}_{k|k-1}\right)^T\right\} \tag{30}$$

The proposed approach assumes that the components of the observation vector in (12) can be processed sequentially, one-by-one, as uncorrelated scalar observations. If this is not a case, such goal can be achieved by making the observation noise covariance matrix, R, in (13) diagonal, using the

Cholesky decomposition [43]. Taking into account that the prediction and the estimation steps in the standard Kalman filter are mutually independent, the roust Kalman filter prediction step remains unchanged, as is defined by (14) – (15), while the estimation step defined by the (16) – (20), has to be redesigned by using the two-step optimization procedure defined by (21) – (30).

This can be achieved by utilizing the influence function (26) on the scaled measurement residual. The scale factor, d_k, can be generated by using the calculation of covariance residual matrix, S_k, as:

$$d_k = S_k^{1/2} = \left[H_k P_{k|k-1} H_k^T + R_k\right]^{1/2} \tag{31}$$

At the end, the robust normalizing penalty factor is introduced:

$$\omega_k = \begin{cases} \frac{\psi(\varepsilon_k/d_k)}{\varepsilon_k/d_k} & ; \varepsilon_k \neq 0 \; and \; d_k \neq 0 \\ 1 & ; \varepsilon_k = 0 \; and/or \; d_k = 0 \end{cases} \tag{32}$$

presenting the slope of the influence function (26).

As the result of the second step of the optimization procedure (27) – (30), the gain equation of the robustified Kalman filter, can be expressed as:

$$K_R = \omega_k P_{k|k-1} H_k^T d_k^{-2} \tag{33}$$

while, a new robust state estimate and the corresponding estimation error covariance matrix are then given by:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_R \varepsilon_k \tag{34}$$

$$P_{k|k} = [I - K_R H_k] P_{k|k-1} \tag{35}$$

As the percentage of outliers in measurement data is rather small (as a rule, 5 to 10 %), the most observations correspond to the nominal Gaussian distribution and will belong to the linear part of the influence function, ψ, robust normalizing penalty factor ω_k is equal to 1. Thus, the robust Kalman gain K_R is equal to the optimal Kalman gain. On the other hand, when the measurement data contain the outliers, which correspond to the saturation part of the influence function, ω_k tends to zero, yielding a small value of K_R. Small value of K_R produces small changes in the state vector and thus reduces the influence of outliers.

Finally, the parameter Δ, defines quantitatively the Huber's ψ-influence function and depends on the degree of contamination by outliers. Unfortunately, the contamination degree, is not exactly known in practice, and it also cannot be estimated accurately by the measurement residual sequence [22]. In many industrial applications, the choice $\Delta = 1.5$ produces satisfactory results, and this is known as the 1.5-Huber's M-robust estimator. However, such fixed Δ-value is not suitable for the SWIR object tracking problem concerned, since observations during target maneuver may be declared to be outliers thus deteriorating the tracking performance. Therefore, to adequately illustrate a situation on the scene, the value of the parameter Δ has to be determined adaptively. Such tuning of the Δ-parameter results in an adaptive M-robust Kalman filtering technique as it is presented in the sequel.

B. PARAMETERS TUNING

When designing the Kalman filter, the first step is to set up the state-transition matrix and the observation matrix. Defined object state model (9) is used in the Kalman filter, so the state-transition matrix, F, and the observation matrix, H, are defined as:

$$F = \begin{bmatrix} 1 & 0 & 0 & 0 & T & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & T & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & T \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{36}$$

The adopted state noise covariance matrix Q, and the observation noise covariance matrix R, are:

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.01 \end{bmatrix} \quad R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix} \tag{37}$$

The matrix G is the identity 7 × 7 dimensional matrix. The time sampling, T, is adopted to be 1 frame (1/25 s), since tracking is applied frame by frame. In defining matrices F and H, the CV model with only position measurement is used to link a position state vector component with its velocity.

In analysis of the position errors and the size errors of the tracked object in Section III-C, it was shown that the outliers occur only in the position measurements. As size of the object (height and width of the bounding box) can change by 5 % or 10 % between successive frames, large intensity errors (outliers) cannot occur in the measurements of the object size. This information is also used when designing the robust Kalman filter. Therefore, the robust estimation step of the Kalman filter is applied only to the first two state vector components in (9), representing the position of the object in the horizontal and vertical directions.

Before proceeding with design of the robust estimation step of the position states, it is important to note that a strong robustness to outliers can decrease the estimator's efficiency under regular conditions. Therefore, a balance between these two requirements needs to be achieved. So, the position errors caused by the maneuvering of the tracked object are not outliers, and must not be cut off. On the other hand, the errors caused by short-term or long-term occlusions, when the multiple search windows are activated, present the outliers and need to be removed. To tolerate position errors caused by the object maneuvering, while removing outliers, the robust Kalman filter is optimized using adaptive saturation threshold, Δ, in Huber's influence function. Value of the parameter, Δ, should be as high as possible in the situations without outliers in measurement data (error distribution is Gaussian). However, when measurement data contain outliers (error distribution is heavy-tailed), value of the parameter, Δ, should be the lowest possible. Therefore, the parameter, Δ, should reflect the presence of outliers in the measurement data.

The value of the parameter Δ is determined depending on data contamination level, and Huber suggested the values of parameter Δ for different efficiency percentages of the estimator [22]. However, contamination degree of data with outliers is unknown in advance since it depends on the situation on the scene. The contamination level can be estimated, and Huber defined the correlation between that level and the value of the parameter Δ. In recent literature [46], weight factors calculated using the Huber's influence function are employed for estimation of the contamination level. Parameter Δ optimized with estimated contamination level in that way, was used in the robust Kalman filter with the Huber's influence function for object tracking in thermal image based on speeded up robust features descriptor. However, the quality of the contamination level estimation depends on the number of samples used for estimation [46]. A larger number of samples provides for more accurate estimate, but introduces additional delay into the algorithm.

To avoid additional delay in the algorithm while optimizing the Δ parameter to illustrate the situation on the scene, we propose an alternative new approach using the response map peak value (PV) information from the basic KCF tracker for Δ optimization. A higher response map PV of the detected object indicates more reliable detection. In that case, the value of the parameter Δ should be higher, which allows for a greater deviation from the position predicted by the Kalman filter and enables tracking of the object's maneuver. When the response map PV of the detected object is low, the value of the parameter Δ should also be smaller, because this indicates the unreliability of the detection, and that the detection may not actually represent the response of the tracked object. When such an unreliable detection is at a position that is significantly far from the predicted one, it represents an outlier, which can lead to tracking termination. Therefore, based on the experimental analysis, we propose the parameter Δ dependence on the response map PV to the following relation (38), as is graphically shown in Fig. 8.

$$\Delta = \begin{cases} 0.1 & PV < 0.25 \\ 0.1 + 14 \cdot (PV - 0.25) & 0.25 \leq PV < 0.35 \\ 1.5 & 0.35 \leq PV < 0.7 \\ 1.5 + 10 \cdot (PV - 0.7) & 0.7 \leq PV < 0.8 \\ 2.5 & PV \geq 0.8 \end{cases} \tag{38}$$

V. EXPERIMENTAL WORK AND RESULTS

In order to examine the performance of the proposed tracking algorithm, we selected two sequences from the created SWIR video database. The first evaluation sequence is a sequence with static occlusions, one long-term full occlusion and four short-term full occlusions. The scenario in the second evaluation sequence includes full occlusions caused by static objects, and partial occlusions caused by moving objects. In this sense, we compared the results of the basic KCF algorithm and the proposed method with a standard, robust, and robust adaptive Kalman filter. For a fair comparison, all parameters of the KCF algorithm used as part of the proposed tracking method are the same as those in the basic KCF [19]. The PSR threshold for failure detection was experimentally set to the value 7. The confidence threshold used in deciding whether to update the target appearance model was set to the value 0.5. The value of parameter Δ in the nonlinear influence function for the robust Kalman filter is fixed at the value 1.5, whereas for the robust adaptive Kalman filter, it changes according to (38). Since the analysis in Section III-C showed that outliers occur only in position measurements, the graphics in Fig. 9 and Fig. 10 show the position state of the algorithm's output and the ground-truth object center, without the size of the tracked object. This metric is valuable for tracking applications in systems in which the camera is mounted on a pan-tilt positioner. Regardless of the correct size estimation, if the object position output is close to the ground-truth position, the system on the pan-tilt can still track the object of interest.

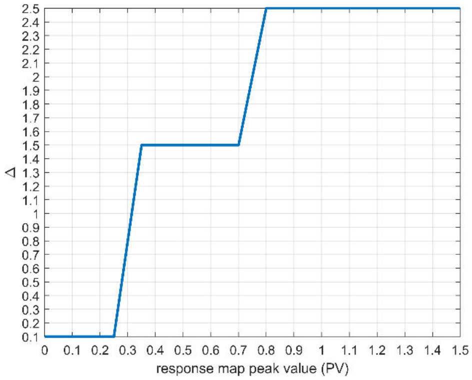
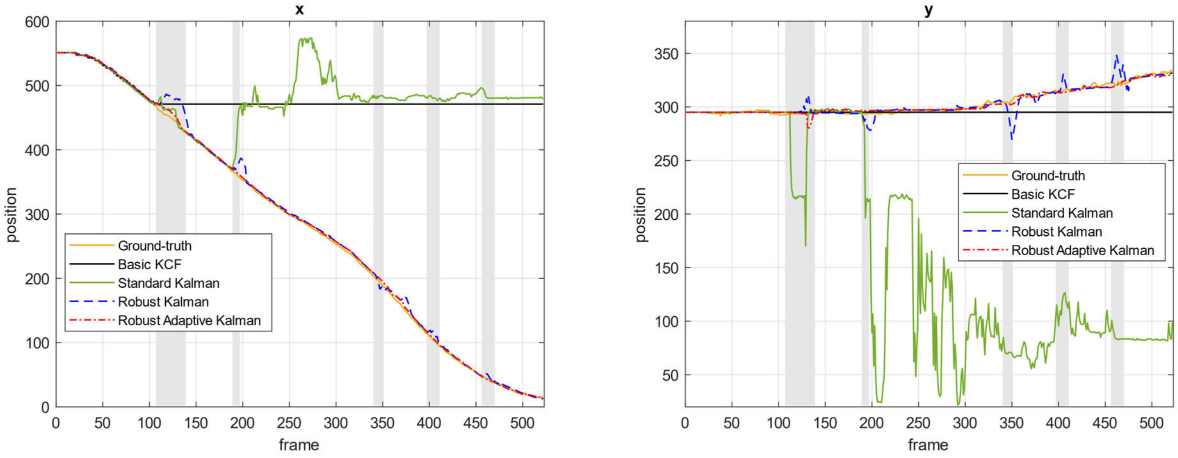


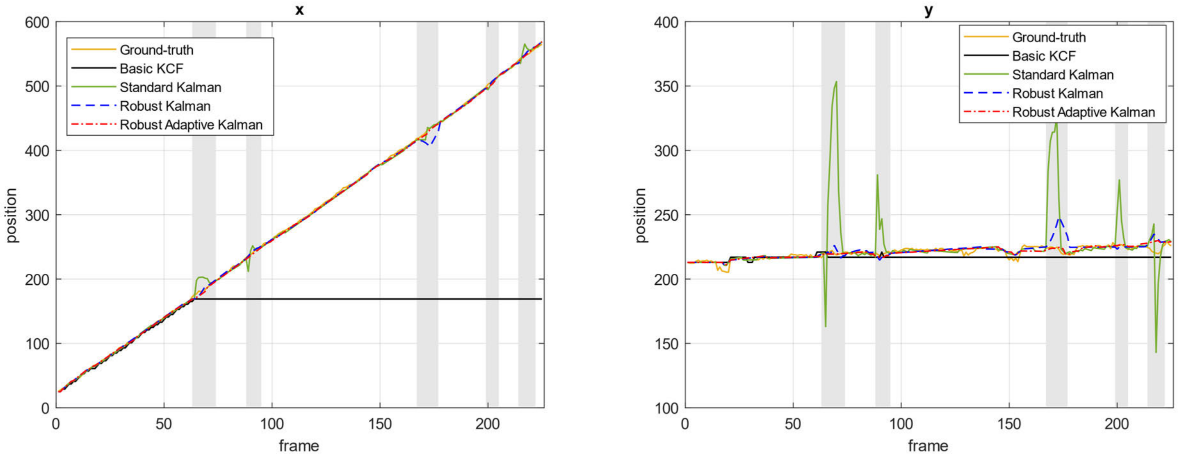
FIGURE 8. Δ-dependence on the response map peak value (PV).

In Fig. 9 and 10, the shaded parts represent occlusions that completely obscured the tracked object. The width of the shaded part represents the duration of occlusion. In the scenario with static occlusions in Fig. 9, it can be seen that the basic KCF tracker remains stuck at the position of the first occlusion and further tracking of the object stops. The proposed method using the standard Kalman filter, although with a large error when the first occlusion occurs, continues to track the object. However, when the second occlusion occurs, such a detection appears in the extended search area (multiple search windows) that causes a huge position error and shifts the search, so the object of interest is no longer in the search area, resulting in the loss of the object. By using the robust Kalman filter, the tracking remained uninterrupted, and the tracking error was significantly lower. When comparing two robust Kalman filters, with fixed and adaptive parameter Δ, it can be seen from Fig. 9 that the robust adaptive one has a



(a) (b)

FIGURE 9. Comparison of the ground truth position and results of the basic KCF algorithm, proposed method with standard Kalman filter, robust Kalman filter ($\Delta = 1.5$) and robust adaptive Kalman filter for the first evaluation sequence in: a) horizontal direction (x), b) vertical direction (y).

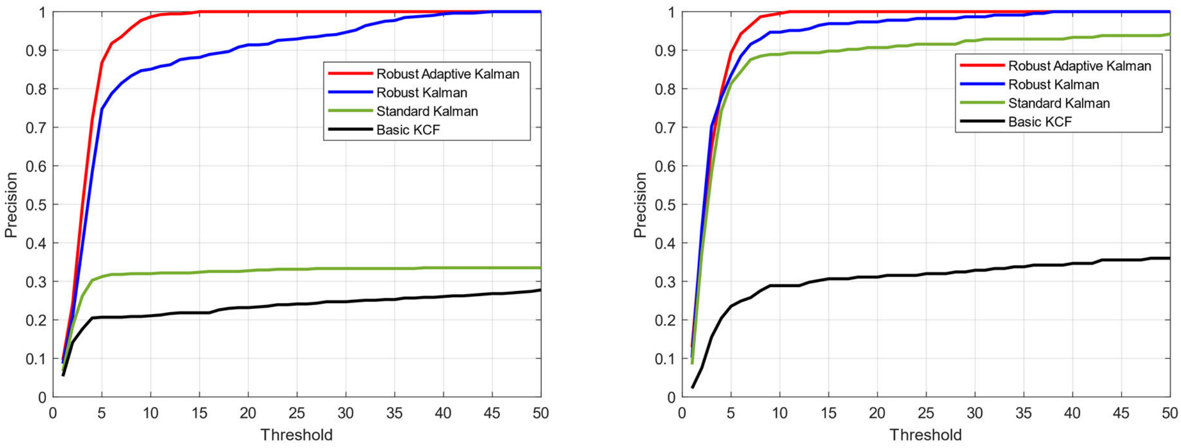


(a) (b)

FIGURE 10. Comparison of the ground truth position and results of the basic KCF algorithm, proposed method with standard Kalman filter, robust Kalman filter ($\Delta = 1.5$) and robust adaptive Kalman filter for the second evaluation sequence in: a) horizontal direction (x), b) vertical direction (y).

smaller error and overcomes occlusions more smoothly. In the scenario shown in Fig. 10 with static and moving occlusions, the basic KCF tracker also remains stuck at the position of the first occlusion. The proposed tracking algorithm manages to overcome all occlusions. Although there is no object loss in this scenario, the standard Kalman filter leads to huge tracking errors when occlusions occur, which is especially visible in the vertical direction. These errors in pan-tilt systems can lead to a sudden movement of the system and FOV of the camera, which may result in the object being out of the camera's FOV. When the object appears after occlusion, the detection confidence increases, which also increases the saturation threshold Δ of the influence function in the robust adaptive Kalman filter. This enables faster adaptation of the proposed robust tracking algorithm and re-detection of the object. The result is that the estimator converges faster and occlusions are overcome more smoothly. In both scenarios, until the occurrence of the first occlusion, the performances of all algorithms are almost identical because the proposed algorithm is designed to rely on the basic KCF in regular situations.

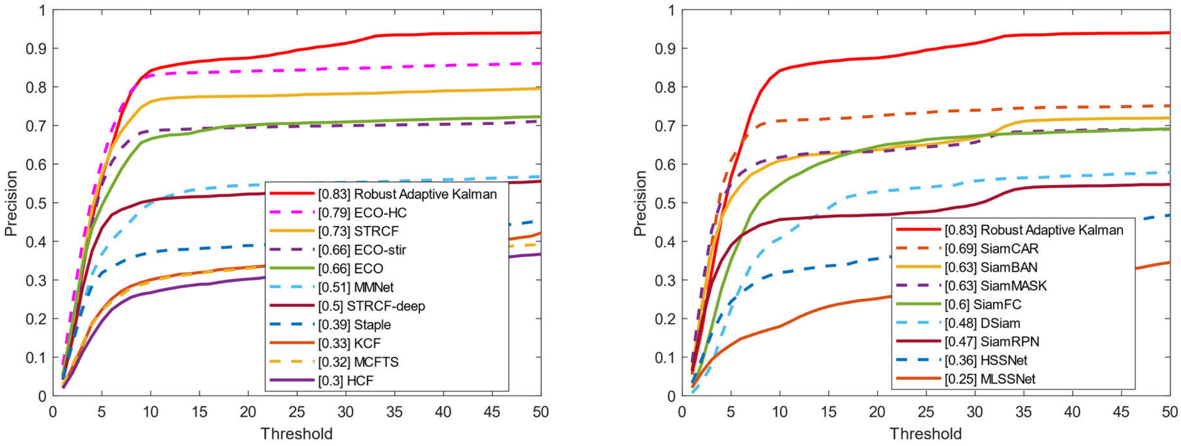
From a robustness perspective, the M-robustified Kalman filter with an adaptive saturation threshold Δ, related to Huber's robust influence function, was shown to suppress the observation outliers in various scenarios with a breakdown point of 25 %. The breakdown point is defined as the largest fraction of contamination for which the robust estimator yields an acceptable maximum estimation bias. In the proposed tracking method, outliers may occur when the extended search area is activated. In the case when the occlusion lasts too long or there are many consecutive occlusions, so the extended search area is active for a long period,



(a)

(b)

FIGURE 11. Precision plots for the basic KCF algorithm, proposed method with standard Kalman filter, robust Kalman filter ($\Delta = 1.5$) and robust adaptive Kalman filter for the a) first evaluation sequence b) second evaluation sequence.



(a)

(b)

FIGURE 12. Precision plots comparison of the proposed method and: a) discriminative correlation filter-based trackers b) Siamese network-based trackers.

the percentage of outliers in the measurement sequence can increase and reach a breakdown point. This may result in the divergence of the Kalman filter and object loss.

Fig. 11 shows precision plots comparison of these algorithms, and confirm the performance of the algorithms in the scenarios shown in Fig. 9 and 10. The tracking precision measure is expressed as the percentage of video sequence frames in which the estimated locations of the tracked object are within a specified threshold from the ground-truth positions (measured as the Euclidean distance in pixels).

Performance comparison of the proposed algorithm in scenarios with occlusions is performed with two classes of state-of-the-art trackers: discriminative correlation filters and deep Siamese networks, which have been recognized as the dominant video tracking paradigms [18]. We selected traditional discriminative correlation filters: Staple [47], KCF [19], and STRCF [48], as well as deep learning based discriminative correlation filters trained for visual object tracking: HCF [49], ECO [50], ECO-HC [50], and STRCF-deep [48], and trained for thermal object tracking: MCFTS [51], ECO-stir [52], and MMNet [53]. From the class of deep Siamese networks, trackers trained for visual object tracking were selected: SiamFC [54], DSiam [55], SiamRPN [56], SiamMASK [57], SiamCAR [58], and SiamBAN [59], as well as those trained for thermal infrared tracking: HSSNet [60] and MLSSNet [61]. Selected trackers were tested on all SWIR video sequences from database. The results are presented in Fig. 12, using the precision plots and Area Under Curve (AUC) metric for each tracker.

As shown in Fig. 12, the proposed method demonstrates significantly better performance in tracking the object of interest in the presence of occlusions compared to other

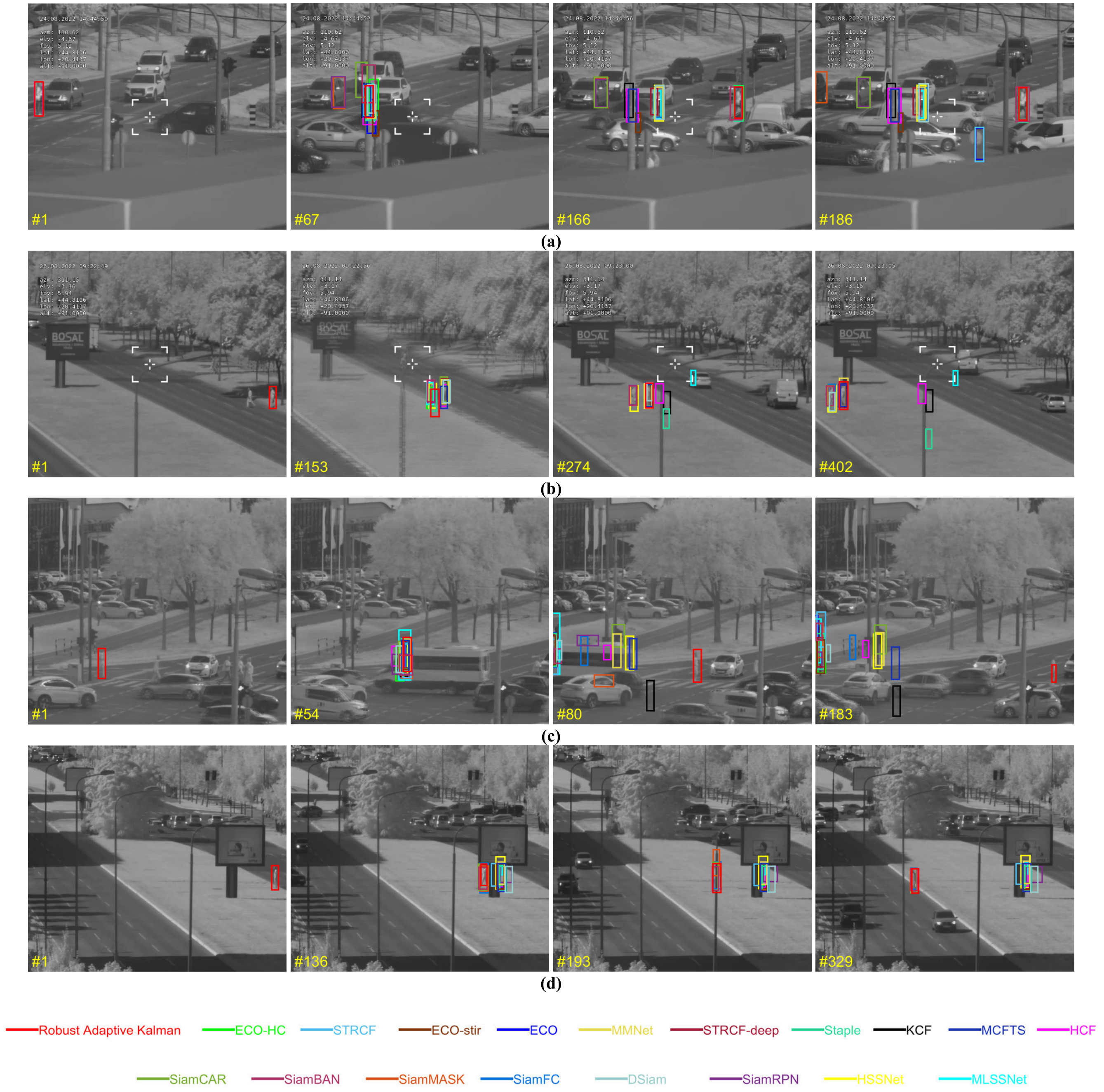


FIGURE 13. Bounding boxes obtained by different tracking algorithms in selected SWIR video sequences with challenging tracking scenarios: a) short-term full static and short-term partial moving occlusions, b) clutter, short-term occlusion and camera shaking, c) long-term complete moving occlusion, d) one long-term and multiple short-term static full occlusions.

discriminative trackers, as well as trackers based on deep Siamese networks. In the majority of cases, especially in the case of long-term full occlusion, analyzed trackers remain stuck at the position where the occlusion occurs. In addition, none of the deep learning-based algorithms were trained on SWIR images. By applying occlusion detection, then an extended search area, and the robust adaptive Kalman filter, it is possible to successfully overcome occlusions, re-detect the object and eliminate outliers, which results in the successful tracking of the object of interest in SWIR imaging.

To clearly demonstrate the performance of the proposed algorithm in comparison to the selected 18 trackers, in Fig. 13 are shown bounding boxes generated by trackers on various SWIR sequences. The first sequence in Fig. 13(a) shows a scenario with short-term full static occlusions and short-term partial moving occlusions. The sequence in Fig. 13(b) shows

TABLE 2. Performance comparison of the proposed tracking algorithm with representative discriminative correlation filter based trackers and Siamese based trackers in terms of speed.

Tracker	Language	Hardware	Speed (FPS)
Robust Adaptive Kalman	Matlab	CPU	58
Staple [47]	Matlab	CPU	8
KCF [19]	Matlab	CPU	276
STRCF [48]	Matlab	GPU	37
HCF [49]	Matlab	CPU	3
ECO [50]	Matlab	GPU	14
ECO-HC [50]	Matlab	GPU	26
STRCF-deep [48]	Matlab	GPU	7
MCFTS [51]	Matlab	GPU	6
ECO-stir [52]	Matlab	GPU	15
MMNet [53]	Matlab	GPU	9
SiamFC [54]	Matlab	GPU	76
DSiam [55]	Matlab	GPU	17
SiamRPN [56]	Python	GPU	29
SiamMASK [57]	Python	GPU	31
SiamCAR [58]	Python	GPU	28
SiamBAN [59]	Python	GPU	28
HSSNet [60]	Matlab	GPU	9
MLSSNet [61]	Matlab	GPU	9

the scenario of tracking an object of interest near another similar object (clutter problem), with short-term occlusion and camera shaking caused by different disturbances, which is a very common case in a real-life scenario. In Fig. 13(c) is given a scenario with moving occlusions, with a particularly pronounced problem of long-term complete moving occlusion. Fig. 13(d) shows a scenario with one long-term and multiple short-term static full occlusions. In all shown sequences, it can be seen that the bounding box estimated by the proposed method with robust adaptive Kalman filter continues to track objects after different types of occlusions.

All trackers were tested on a PC with an i7 2.6 GHz CPU, 32GB RAM, and NVIDIA GeForce RTX 2070 GPU. The results of the performance comparison in terms of speed (average frame rate) are listed in Table 2. The proposed tracking method is implemented in the MATLAB software package and achieves a speed of 58 FPS, which allows real-time processing.

VI. CONCLUSION

This paper presents a feasible new algorithm for object tracking in the SWIR spectral domain, which integrates the KCF algorithm and a robust adaptive version of the Kalman filter. Although the KCF algorithm is generally suitable for object tracking in real-time, when the size, orientation, and appearance of the object change, its performance drops. Especially in the case of occlusions, the KCF tracker remains stuck at the position of the occlusion occurrence. Therefore, the paper first proposes improvements of the KCF algorithm in object size estimation, adaptive update of the target appearance model, and occlusion detection. For prediction purposes and object re-detection, the Kalman filter and extended search area were used, so that tracking can be continued even in the case of full occlusion. Kalman filter can be used only when a correct a priory description of the system state dynamics is provided, and the measurement noise follows the Gaussian distribution. Therefore, a detailed statistical analysis of the proposed SWIR object tracking method was provided. The analysis showed that the real data is contaminated by a small percentage of large intensity measurement errors. Although using an extended search area helps in better re-detection of the object after occlusion, it may introduce large intensity position errors in measurement data that can lead to object loss. These errors can be treated as outliers. Therefore, tracking method is robustified by applying the nonlinear measurement residuals processing using the nonlinear Huber's influence function in the Kalman filter estimation step. Additionally, the proposed robust Kalman filter adapts to the scene conditions by tuning the saturation parameter of the influence function based on the detection confidence of the basic KCF tracker. The proposed novel tracking method is efficient in object tracking under regular conditions, resistant to outliers in the measurement data when tracking failure is detected, and successful in overcoming occlusions and object re-detection.

The proposed algorithm effectively continued to track the object where the basic KCF got stuck. Also, algorithm with robust adaptive Kalman filter has a significantly lower tracking error than the one using the standard Kalman filter or the robust Kalman filter with a fixed influence function saturation parameter. It is also demonstrated that the algorithm achieves a better performance in tracking of maneuvering object in scenarios with occlusions in comparison to various state-of-the-art tracking algorithms, evaluated on the created dataset of SWIR video sequences. Moreover, the proposed tracking algorithm has the advantages of a simple structure and low computational requirements, and thus achieves real-time performance.

REFERENCES

[1] A. Cavallaro and E. Maggio, *Video Tracking: Theory and Practice*. Hoboken, NJ, USA: Wiley, 2011.

[2] M. Kristan et al., "The tenth visual object tracking VOT2022 challenge results," in *Proc. ECCV Workshops*, 2022, pp. 431–460.

[3] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, R. Pflugfelder, J.-K. Kämäräinen, H. J. Chang, M. Danelljan, L. Cehovin, A. Lukežic, O. Drbohlav, J. Käpylä, G. Häger, S. Yan, J. Yang, Z. Zhang, and G. Fernández, "The ninth visual object tracking VOT2021 challenge results," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2021, pp. 2711–2738.

[4] M. Kristan et al., "The eighth visual object tracking VOT2020 challenge results," in *Proc. ECCV Workshops*, 2020, pp. 547–601.

[5] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, L. Cehovin, G. Fernandez, T. Vojir, G. Hager, G. Nebehay, R. Pflugfelder, A. Gupta, A. Bibi, A. Lukezic, A. Garcia-Martin, A. Saffari, A. Petrosino, and A. S. Montero, "The visual object tracking VOT2015 challenge results," in *Proc. ICCV*, Dec. 2015, pp. 564–586.

[6] Z. Chen, B. Zhong, G. Li, S. Zhang, R. Ji, Z. Tang, and X. Li, "SiamBAN: Target-aware tracking with Siamese box adaptive network," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 4, pp. 5158–5173, Apr. 2023.

[7] Y. Zheng, B. Zhong, Q. Liang, Z. Tang, R. Ji, and X. Li, "Leveraging local and global cues for visual tracking via parallel interaction network," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 4, pp. 1671–1683, Apr. 2023.

[8] M. Felsberg et al., "The thermal infrared visual object tracking VOT-TIR2015 challenge results," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop (ICCVW)*, Dec. 2015, pp. 76–88.

[9] K. Lebeda, S. Hadfield, and R. Bowden, "The thermal infrared visual object tracking VOT-TIR2016 challenge results," in *Proc. Eur. Conf. Comput. Vis. (ECCV) Workshops*, 2016, pp. 824–849.

[10] Q. Liu, Z. He, X. Li, and Y. Zheng, "PTB-TIR: A thermal infrared pedestrian tracking benchmark," *IEEE Trans. Multimedia*, vol. 22, no. 3, pp. 666–675, Mar. 2020.

[11] Q. Liu, X. Li, Z. He, C. Li, J. Li, Z. Zhou, D. Yuan, J. Li, K. Yang, N. Fan, and F. Zheng, "LSOTB-TIR: A large-scale high-diversity thermal infrared object tracking benchmark," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 3847–3856.

[12] M. P. Hansen and D. S. Malchow, "Overview of SWIR detectors, cameras, and applications," *Proc. SPIE*, vol. 6939, Mar. 2008, Art. no. 69390I.

[13] R. G. Driggers, V. Hodgkin, and R. Vollmerhausen, "What good is SWIR? Passive day comparison of VIS, NIR, and SWIR," *Proc. SPIE*, vol. 8706, Jun. 2013, Art. no. 87060L.

[14] Z. Kandylakis, K. Vasili, and K. Karantzalos, "Fusing multimodal video data for detecting moving objects/targets in challenging indoor and outdoor scenes," *Remote Sens.*, vol. 11, no. 4, p. 446, Feb. 2019.

[15] M. Pavlović, P. Milanović, M. Stanković, D. Perić, I. Popadić, and M. Perić, "Deep learning based SWIR object detection in long-range surveillance systems: An automated cross-spectral approach," *Sensors*, vol. 22, no. 7, p. 2562, 2022.

[16] C. Kwan, B. Chou, J. Yang, and T. Tran, "Compressive object tracking and classification using deep learning for infrared videos," *Proc. SPIE*, vol. 10995, May 2019, Art. no. 1099506.

[17] C. Kwan, B. Chou, J. Yang, A. Rangamani, T. Tran, J. Zhang, and R. Etienne-Cummings, "Target tracking and classification using compressive sensing camera for SWIR videos," *Signal, Image Video Process.*, vol. 13, no. 8, pp. 1629–1637, Nov. 2019.

[18] S. Javed, M. Danelljan, F. S. Khan, M. H. Khan, M. Felsberg, and J. Matas, "Visual object tracking with discriminative filters and Siamese networks: A survey and outlook," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 5, pp. 6552–6574, May 2023.

[19] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 583–596, Mar. 2015.

[20] N. Latinović, I. Popadić, B. Tomić, A. Simić, P. Milanović, S. Nijemčević, M. Perić, and M. Veinović, "Signal processing platform for long-range multi-spectral electro-optical systems," *Sensors*, vol. 22, no. 3, p. 1294, Feb. 2022.

[21] V. Barnett and T. Lewis, *Outliers in Statistical Data*. Hoboken, NJ, USA: Wiley, 1994.

[22] P. J. Huber and E. M. Ronchetti, *Robust Statistics*. Hoboken, NJ, USA: Wiley, 2009.

[23] C. Xiu and Y. Ma, "Kernel correlation filter tracking strategy based on adaptive fusion response map," *IET Image Process.*, vol. 16, no. 4, pp. 937–947, Mar. 2022.

[24] R. Xia, Y. Chen, and B. Ren, "Improved anti-occlusion object tracking algorithm using unscented Rauch-Tung-Striebel smoother and kernel correlation filter," *J. King Saud Univ., Comput. Inf. Sci.*, vol. 34, no. 8, pp. 6008–6018, Sep. 2022.

[25] F. Chen, W. Xie, and T. Xia, "Target tracking algorithm based on kernel correlation filter with anti-occlusion mechanisms," in *Proc. 15th IEEE Int. Conf. Signal Process. (ICSP)*, vol. 1, Dec. 2020, pp. 220–225.

[26] Y. Zhou, W. Yang, and Y. Shen, "Scale-adaptive KCF mixed with deep feature for pedestrian tracking," *Electronics*, vol. 10, no. 5, p. 536, Feb. 2021.

[27] T. Zhou, M. Zhu, D. Zeng, and H. Yang, "Scale adaptive kernelized correlation filter tracker with feature fusion," *Math. Problems Eng.*, vol. 2017, pp. 1–8, Jan. 2017.

[28] L. Gan and Y. Ma, "Long-term correlation filter tracking algorithm based on adaptive feature fusion," in *Proc. SPIE*, vol. 12083, pp. 253–263, Feb. 2022.

[29] L. Liu, T. Feng, and Y. Fu, "Learning multifeature correlation filter and saliency redetection for long-term object tracking," *Symmetry*, vol. 14, no. 5, p. 911, Apr. 2022.

[30] J. Zhang, S. Jiang, Y. Zhang, X. Liu, D. Wang, and F. Qiu, "Long-term tracking algorithm using deep features and a single shot multibox detector," *J. Electron. Imag.*, vol. 27, no. 5, Sep. 2018, Art. no. 053019.

[31] Y. Liu, Y. Liao, C. Lin, Y. Jia, Z. Li, and X. Yang, "Object tracking in satellite videos based on correlation filter with multi-feature fusion and motion trajectory compensation," *Remote Sens.*, vol. 14, no. 3, p. 777, Feb. 2022.

[32] J. Ni, X. Zhang, P. Shi, and J. Zhu, "An improved kernelized correlation filter based visual tracking method," *Math. Problems Eng.*, vol. 2018, pp. 1–12, Dec. 2018.

[33] X. Chen, X. Xu, Y. Yang, H. Wu, J. Tang, and J. Zhao, "Augmented ship tracking under occlusion conditions from maritime surveillance videos," *IEEE Access*, vol. 8, pp. 42884–42897, 2020.

[34] T. Li, S. Zhao, Q. Meng, Y. Chen, and J. Shen, "A stable long-term object tracking method with re-detection strategy," *Pattern Recognit. Lett.*, vol. 127, pp. 119–127, Nov. 2019.

[35] J. Wang, H. Yang, N. Xu, C. Wu, Z. Zhao, J. Zhang, and D. O. Wu, "Long-term target tracking combined with re-detection," *EURASIP J. Adv. Signal Process.*, vol. 2021, no. 1, pp. 1–16, Dec. 2021.

[36] H. Wei, "A UAV target prediction and tracking method based on KCF and Kalman filter hybrid algorithm," in *Proc. 2nd Int. Conf. Consum. Electron. Comput. Eng. (ICCECE)*, Jan. 2022, pp. 711–718.

[37] H. Zhang, H. Zhan, L. Zhang, F. Xu, and X. Ding, "A Kalman filter-based kernelized correlation filter algorithm for pose measurement of a micro-robot," *Micromachines*, vol. 12, no. 7, p. 774, Jun. 2021.

[38] M. Müller, A. Bibi, S. Giancola, S. Alsubaihi, and B. Ghanem, "TrackingNet: A large-scale dataset and benchmark for object tracking in the wild," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 300–317.

[39] H. Fan, L. Lin, F. Yang, P. Chu, G. Deng, S. Yu, H. Bai, Y. Xu, C. Liao, and H. Ling, "LaSOT: A high-quality benchmark for large-scale single object tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5374–5383.

[40] A. Berg, J. Ahlberg, and M. Felsberg, "A thermal object tracking benchmark," in *Proc. 12th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Aug. 2015, pp. 1–6.

[41] S. Challa, M. R. Morelande, D. Mušicki, and R. J. Evans, *Fundamentals of Object Tracking*. Cambridge, U.K.: Cambridge Univ. Press, 2011.

[42] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 3464–3468.

[43] M. S. Grewal and A. P. Andrews, *Kalman Filtering: Theory and Practice With MATLAB*. Hoboken, NJ, USA: Wiley, 2015.

[44] Z. Banjac and B. Kovačević, "Robustified Kalman filtering using both dynamic stochastic approximation and M-robust performance index," *Tech. Gazette*, vol. 29, no. 3, pp. 907–914, 2022.

[45] B. Kovačević and Ž. Durovicć, "*Fundamentals of Stochastic Signals, Systems and Estimation Theory With Worked Examples*. Berlin, Germany: Springer, 2011.

[46] N. Vlahović and Z. Djurovic, "Robust tracking of moving objects using thermal camera and speeded up robust features descriptor," *Int. J. Adapt. Control Signal Process.*, vol. 35, no. 4, pp. 549–566, Apr. 2021.

[47] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr, "Staple: Complementary learners for real-time tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1401–1409.

[48] F. Li, C. Tian, W. Zuo, L. Zhang, and M. Yang, "Learning spatial-temporal regularized correlation filters for visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4904–4913.

[49] C. Ma, J. Huang, X. Yang, and M. Yang, "Hierarchical convolutional features for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3074–3082.

[50] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ECO: Efficient convolution operators for tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6638–6646.

[51] Q. Liu, X. Lu, Z. He, C. Zhang, and W.-S. Chen, "Deep convolutional neural networks for thermal infrared object tracking," *Knowl.-Based Syst.*, vol. 134, pp. 189–198, Oct. 2017.

[52] L. Zhang, A. Gonzalez-Garcia, J. van de Weijer, M. Danelljan, and F. S. Khan, "Synthetic data generation for end-to-end thermal infrared tracking," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 1837–1850, Apr. 2019.

[53] Q. Liu, X. Li, Z. He, N. Fan, D. Yuan, W. Liu, and Y. Liang, "Multi-task driven feature models for thermal infrared tracking," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 7, pp. 11604–11611.

[54] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-convolutional Siamese networks for object tracking," in *Proc. Comput. Vis. ECCV Workshops*, vol. 14. Amsterdam, The Netherlands: Springer, Oct. 2016, pp. 850–865.

[55] Q. Guo, W. Feng, C. Zhou, R. Huang, L. Wan, and S. Wang, "Learning dynamic Siamese network for visual object tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1763–1771.

[56] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, "High performance visual tracking with Siamese region proposal network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8971–8980.

[57] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. S. Torr, "Fast online object tracking and segmentation: A unifying approach," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1328–1338.

[58] D. Guo, J. Wang, Y. Cui, Z. Wang, and S. Chen, "SiamCAR: Siamese fully convolutional classification and regression for visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6269–6277.

[59] Z. Chen, B. Zhong, G. Li, S. Zhang, and R. Ji, "Siamese box adaptive network for visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6668–6677.

[60] X. Li, Q. Liu, N. Fan, Z. He, and H. Wang, "Hierarchical spatial-aware Siamese network for thermal infrared object tracking," *Knowl.-Based Syst.*, vol. 166, pp. 71–81, Feb. 2019.

[61] Q. Liu, X. Li, Z. He, N. Fan, D. Yuan, and H. Wang, "Learning deep multi-level similarity for thermal infrared object tracking," *IEEE Trans. Multimedia*, vol. 23, pp. 2114–2126, 2021.

MILOŠ PAVLOVIĆ (Member, IEEE) received the B.Sc. and M.Sc. degrees from the School of Electrical Engineering, University of Belgrade, Serbia, in 2018 and 2019, respectively, where he is currently pursuing the Ph.D. degree with the Signals and Systems Department. After graduating, he started his professional career with the Vlatacom Institute of High Technologies as a Research and Development Engineer. His current research interests include artificial intelligence, computer vision, digital signal, and image processing.

ZORAN BANJAC (Member, IEEE) received the B.Sc. degree from the Military Technical Institute, University of Belgrade, in 1993, and the M.Sc. and Ph.D. degrees from the School of Electrical Engineering, University of Belgrade, Serbia, in 1998 and 2004, respectively. After graduating, he was with the Institute for Applied Mathematics and Electronics, IPME, Belgrade, where his last position was the Head of scientific research. From 2004 to 2007, he was an Assistant Professor with Singidunum University, Belgrade. From 2006 to 2017, he was a Professor of applied studies with the School of Electrical and Computers Engineering, Belgrade. He has been with the Vlatacom Institute, since 2017, where he is the Head of the Crypto Department. He is the author of four books, more than 50 papers published in international and national journals and conferences, and four technical solutions. His current research interests include the design of crypto protection systems and signal processing.

BRANKO KOVAČEVIĆ (Senior Member, IEEE) received the Ph.D. degree from the University of Belgrade, Belgrade, Serbia, in 1984. In 1981, he joined the Faculty of Electrical Engineering, University of Belgrade, where he is currently a Professor Emeritus. He is the author of eight books and more than 80 articles in scientific journals. His current research interests include robust estimation, system identification, adaptive and nonlinear filtering, optimal and adaptive control, and digital signal processing. He is a member of the EURASIP, the WSAES, and the National Association ETRAN, and a Corresponding Member of the Academy of Engineering Sciences of Serbia. He was a recipient of the Outstanding Research Prize of the Institute of Applied Mathematics and Electronics, the Prize of the Serbian Association for Informatics, and the Prize of the Association of Radio Systems Engineers. He is a Reviewer of IEEE Transactions, *Automatica* (IFAC), and *Signal Processing*.