Channel Coded Distribution Field Tracking for Thermal Infrared Imagery

Amanda Berg[1,2], Jörgen Ahlberg[1,2], Michael Felsberg[2]
[1]Termisk Systemteknik AB, Diskettgatan 11 B, 583 35 Linköping, Sweden
[2]Computer Vision Laboratory, Dept. EE, Linköping University, 581 83 Linköping, Sweden
{amanda.,jorgen.ahl}berg@termisk.se, {amanda.,jorgen.ahl,michael.fels}berg@liu.se

Abstract

We address *short-term, single-object tracking*, a topic that is currently seeing fast progress for visual video, for the case of *thermal infrared (TIR) imagery*. The fast progress has been possible thanks to the development of new template-based tracking methods with online template updates, methods which have not been explored for TIR tracking. Instead, tracking methods used for TIR are often subject to a number of constraints, *e.g.*, warm objects, low spatial resolution, and static camera. As TIR cameras become less noisy and get higher resolution these constraints are less relevant, and for emerging civilian applications, *e.g.*, surveillance and automotive safety, new tracking methods are needed.

Due to the special characteristics of TIR imagery, we argue that template-based trackers based on *distribution fields* should have an advantage over trackers based on spatial structure features. In this paper, we propose a template-based tracking method (ABCD) designed specifically for TIR and not being restricted by any of the constraints above. In order to avoid background contamination of the object template, we propose to exploit background information for the online template update and to adaptively select the object region used for tracking. Moreover, we propose a novel method for estimating object scale change. The proposed tracker is evaluated on the VOT-TIR2015 and VOT2015 datasets using the VOT evaluation toolkit and a comparison of relative ranking of all common participating trackers in the challenges is provided. Further, the proposed tracker, ABCD, and the VOT-TIR2015 winner SRDCFir are evaluated on maritime data. Experimental results show that the ABCD tracker performs particularly well on thermal infrared sequences.

1. Introduction

Tracking of objects in video is a problem that has been subject to extensive research. In recent years, the sub-topic of *short-term single-object tracking* (STSO) has seen significant progress and is important as it is at the core of more complex (long-term, multi-camera, multi-object) tracking systems. Indicators of the popularity of this research topic are challenges and benchmarks like the recurring Visual Object Tracking (VOT) challenge [16, 17, 18], the Online Object Tracking (OTB) benchmarks [28, 29], and the series of workshops on Performance Evaluation of Tracking and Surveillance (PETS) [30].

Thermal infrared tracking has historically been of interest mainly for military purposes. Thermal cameras have delivered noisy images with low resolution, used mainly for tracking small objects (point targets) against colder backgrounds. However, in recent years, thermal cameras have decreased in both price and size while image quality and resolution has improved, which has opened up new application areas [9]. Thermal cameras are now commonly used, *e.g.*, in cars and in surveillance systems. The main advantages of thermal cameras are their ability to see in total darkness, their robustness to illumination changes and shadow effects, and less intrusion on privacy. In 2015, the first challenge on STSO tracking in thermal infrared imagery (VOT-TIR) was organized [6], an indication of an increasing interest from the community.

In this paper, we will discuss the differences between thermal and visual tracking, argue that template-based trackers based on distribution fields are suited for thermal tracking, and propose three enhancements to such tracking methods: First, we propose a method for improving distribution field-based trackers using background weighting of object template updates. Second, we propose a method for improving the search phase in such trackers using an adaptive object region. Third, a scale estimation technique employing background information is evaluated. Finally, we show that these improvements are complementary, and evaluate the resulting tracker on both RGB and TIR sequences.

2. Thermal imaging and tracking

The infrared wavelength band is usually divided into different bands according to their different properties: near infrared (NIR, wavelengths 0.7–1 μm), shortwave infrared (SWIR, 1–3 μm), midwave infrared (MWIR, 3–5 μm), and longwave infrared (LWIR, 7.5–12 μm). Other definitions exist as well. LWIR, and sometimes MWIR, is commonly referred to as thermal infrared (TIR). TIR cameras should not be confused with NIR cameras that are dependent on illumination and in general behave in a similar way as visual cameras.

In thermal infrared, most of the captured radiation is *emitted* from the observed objects, in contrast to visual and near infrared, where most of the radiation is *reflected*. Thus, knowing or assuming material and environmental properties, temperatures can be measured using a thermal camera (*i.e.*, the camera is said to be *radiometric*).

There are two common misconceptions regarding object tracking in TIR. One is that it is all about *hotspot tracking*, that is, tracking warm (bright) objects against a cold (dark) background. In certain military applications, such as missile warning, this assumption is valid, but for most other applications the situation is more complex and hotspot tracking less suitable (this is backed by our experimental results given in Sec. 5). The other misconception is that TIR tracking is identical to tracking in grayscale visual imagery, and, as a consequence, that a tracker that is good for visual tracking is good for TIR tracking. However, there are differences between the two types of imagery that indicate that this is not the case:

First, there are no shadows in TIR. A tracker that is optimized to handle shadows might thus be suboptimal for TIR (*e.g.* a tracker employing foreground/background detection that includes shadow removal [25]).

Second, the noise characteristics are different. Compared to a visual camera, a TIR camera typically has more blooming[1], lower resolution, and a larger percentage of dead pixels. As a consequence, a tracker that depends heavily on features based on (high resolution) spatial structure is presumably suboptimal for TIR imagery.

Third, visual color patterns are discernible in TIR only if they correspond to variations in material or temperature. Again, the consequence is that trackers relying on (high resolution) spatial patterns might be suboptimal for TIR imagery. Moreover, re-identification and resolving occlusions might need to be done differently. For example, two persons with differently patterned or colored clothes might look similar in TIR.

Fourth, in most applications, the emitted radiation change much slower than the reflected radiation. That is, an object moving from a dark room into the sunlight will not immediately change its appearance (as it would in visual imagery). Thus, trackers that exploit the absolute levels (for example, distribution field trackers) should have an advantage in TIR. This is especially relevant for radiometric 16-bit cameras, since they have a dynamic range large enough to accommodate relevant temperature intervals without adapting the dynamic range to each frame.

3. Related Work

A common approach to TIR tracking is to combine a detector with a motion tracking filter such as a Kalman filter (KF) or a particle filter (PF). The detector is typically based on thresholding, *i.e.*, hotspot detection, or a pre-trained classifier. This approach is intuitive when tracking warm objects in low-resolution images and has historically been the main use of thermal cameras since typical objects of interest often generate kinetic energy in order to move (*e.g.* airborne and ground vehicles). Extensions include the work of Padole and Alexandre [21, 22], who use a PF for motion tracking and combine spatial and temporal information. Lee et al. [20] improve tracking performance in the case of repetitive patterns using a KF and a curve matching framework. Gade and Moeslund [10] use hotspot detection followed by splitting and connection of blobs in order to track sports players and maintain the players' identities. Goubet et al. [11] also rely on high contrast between object and background. In contrast, Skoglar et al. [27] use a pre-trained boosting based detector and improve tracking in a surveillance application using road network information and a multimodal PF. Portmann et al. [24] combine background subtraction and a part-based detector using a support vector machine to classify Histograms of Gradients. Jüngling and Arens [15] combine tracking and detection in a single framework, extracting SURF features and using a KF for predicting the motion of individual features.

Many published methods on TIR tracking have strong constraints, for example assuming static camera and/or background [10, 11, 21], pre-trained detector (implying known object class) [15, 24, 27], and high object-background contrast [10, 11, 21].

In the visual domain, methods based on matching a template (object model) that is trained and updated online is currently subject to intensive research [28]. Few papers can be found where the recent ideas leading to such fast progress for visual video are transferred to TIR tracking. The existing template-based TIR-tracking methods often assume small targets and low signal-to-noise ratio. Further, they frequently use separate detection and tracking phases where the target tracking step is based on spatio-temporal correlation. In contrast, many of the methods used in the recent RGB-tracking benchmarks employ joint detection and tracking. Some TIR template-based tracking methods activate template matching only for the purpose of recovery [2, 19]. Bal et al. [2] base the activation on a Cartesian distance metric while Lamberti et al. [19] use a motion prediction-based metric. The latter strategy improves the robustness of the tracker. Johnston et. al. [14] use a dual domain approach with AM-FM consistency checks. The

[1] TIR cameras have blooming, but not the same kind of blooming as CCD arrays

method automatically detects when a template update is needed through a combination of pixel and modulation domain correlation trackers.

Alam and Bhuiyan [1] provide an overview of the characteristics of matched filter correlation techniques for detection and tracking in TIR imagery. The filters are, however, pre-trained and not adaptively updated. He et al. [13] employs some of the ideas of RGB-tracking methods and presents an infrared target tracking method under a tracking-by-detection framework based on a weighted correlation filter.

Methods based on *distribution field tracking* (DFT) [26] rely neither on color nor on features based on sharp edges. Instead, the object model is a distribution field, *i.e.*, an array of probability distributions; one distribution for each pixel in the template. In [26], the probability distributions are represented by local histograms. Felsberg [5] showed that changing the representation of the probability distributions to channel coded vectors [7, 12] improves tracking performance. A channel vector basically consists of sampled values of overlapping basis functions, *e.g.*, $\cos^2$ or B-spline functions. Each of the N elements (channels) of the channel vector describe to what extent the encoded value activates the n'th basis function.

In the particular case of EDFT [5] (Enhanced Distribution Field Tracking), the object model is built by channel encoding of an image patch of the object which is then convolved with a 2D Gaussian kernel. This model is updated in each new frame using an update factor $\alpha \in [0, 1]$ as

$$m_{obj}^t(i, j, n) = (1 - \alpha)m_{obj}^{t-1}(i, j, n) + \alpha p^t(i, j, n) \quad (1)$$

where m_{obj}^t is the object model at time t, and p is the best matching channel encoded patch in the current frame. $i \in [0, I - 1]$ and $j \in [0, J - 1]$ where I, J are the width and height of the model and $n \in [0, N - 1]$ where N is the number of channels.

The best matching patch p is found by performing a local search starting from a predicted image position. The search procedure is equal to that of DFT [26]. A distance measure, d, is calculated as the absolute difference of the object model and the channel encoded query patch, q (2). p is found at the position where d is minimized.

$$d = \sum_{i=0}^{I-1} \sum_{j=0}^{J-1} \sum_{n=0}^{N-1} |m_{obj}(i, j, n) - q(i, j, n)| \quad (2)$$

4. Proposed tracking method

The proposed tracking method is based on the Enhanced Distribution Field Tracking (EDFT). The word "enhanced" refers to the modifications of the DFT tracker introduced in [5], the most important being the change of representation of the distribution field from soft histograms to B-spline kernel channel coded features. EDFT has achieved good results in the VOT challenges 2013 and 2014 for visual sequences. EDFT neither relies on color nor on features based on sharp edges which makes it suitable for thermal infrared imagery, however, as all template-based trackers it suffers from background contamination and, moreover, it cannot adaptively rescale to changing object size.

4.1. Background contamination in tracking

In all template-based tracking methods, there is a risk that the spatial region to be tracked contains background pixels. This leads to two problems; in the search phase respectively the template update phase. First, if the region contains background pixels, which is highly probable in practice, the tracker will try to track not only the object but also some of the surrounding background. Second, if the template (object model) is continuously updated, background pixels might be included. With increasing number of background pixels in the model, the risk of losing track grows. We address these two problems in two different ways below.

The same principle of how to build an object model (1) in EDFT can be used to create and update a background model. Each pixel is represented by a channel vector, and in each frame, the background model m_{bg} is updated using a background update factor β.

$$m_{bg}^t(x, y, n) = (1 - \beta)m_{bg}^{t-1}(x, y, n) + \beta z^t(x, y, n) \quad (3)$$

$x \in [0, W - 1]$ and $y \in [0, H - 1]$ where W and H are the width and height of the image respectively. $z^t(x, y, n)$ is the current frame, the image at time t. Our approach is to use the additional information from the background model when tracking an object. Note that we do not need to build the model in advance – the background model is continuously built and updated for a region around the object (we use $I + 50$ and $J + 30$ pixels) only. If the camera is moving, the background model is not corrected for the camera motion.

Also note that any other model for background pixel distributions can be used, for example a mixture of Gaussians, a kernel density estimator [23] or a soft histogram [26]. However, since the channel coded distribution is already available from the EDFT, we can use this without extra computational cost.

4.2. Background weighted model update (B-EDFT)

We propose a method to mitigate the problem of background pixels contaminating the process of updating the object model. From the background model described above, a soft mask consisting of the probabilities of pixels belonging to the foreground can be made. The ℓ_1 norm of a second order B-spline kernel channel vector is one [8]. This implies that $\sum_{n=0}^{N-1} |m_{bg}(i, j, n) - p(i, j, n)| \in [0, 2]$ for some

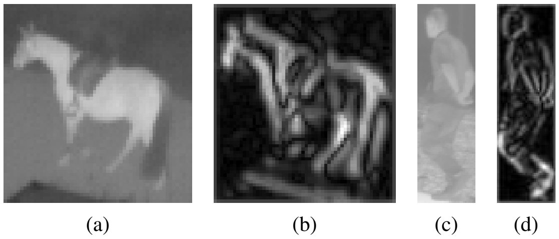


Figure 1: Example of image patch and corresponding foreground mask from sequences (a) horse (image) (b) horse (mask) (c) crouching (image) (d) crouching (mask).

pixel (i, j). $m_{bg}(i, j, n)$ is the background model at the position of the best matching patch $p(i, j, n)$, and N is the number of channels. The individual elements of the mask $b\ (I \times J) \in [0, 1]$ are then calculated as

$$b(i, j) = \frac{\sum_{n=0}^{N-1} |m_{bg}(i, j, n) - p(i, j, n)|}{2}. \tag{4}$$

A lower value $b(i, j)$ indicates a higher similarity between pixel (i, j) and the background. The elements of the foreground mask b can be used as weights for the different pixels when the object model is updated.

In order to incorporate the mask into the update of the object model, (1) is modified to

$$m^t_{obj}(i, j, n) = (1-\alpha b(i, j))m^{t-1}_{obj}(i, j, n)+\alpha b(i, j)p(i, j, n). \tag{5}$$

That is, the more likely a pixel is to belong to the background, the slower the corresponding distribution in the object model is updated. Thus, the risk of background contamination in the model is reduced.

In Fig. 1, two examples of foreground masks are provided. Note that in the horse sequence, the camera is moving and there is a high contrast between the object and the background. Hence, the outline of the object in the mask is wider and has higher intensity compared to the outline of the person in the crouching sequence.

If the object slows down to a stand still, it will be considered background and the elements of the foreground mask will be low. Hence, the object model will not be updated but since the object do not change significantly the tracker will be able to localize the object anyway.

4.3. Adaptive object region (A-EDFT)

Second, the problem of the tracked region encompassing background pixels in the search phase is addressed. A subregion suitable for tracking is adaptively selected. In visual imagery, this might correspond to choosing good features to track, whereas in thermal imagery, selecting a region is typically more suitable. In thermal imagery, there is less structure and sharp edges, and the tracker exploits intensity rather than structure. Thus, within the initial region to track we select an inner region that is used for the actual tracking. In this particular case, the region is a bounding box.

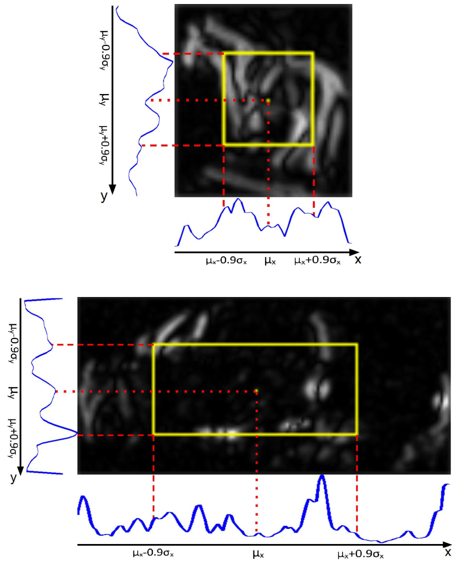


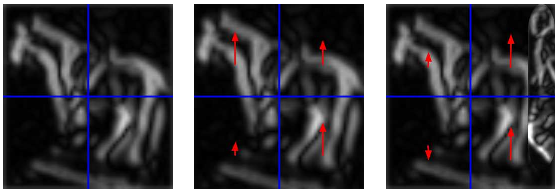
Figure 2: Illustration of how the inner bounding box is adaptively selected for sequence horse (left) and car (right). The blue curves represent the projected pixel values of the mask onto the x- and y-axis respectively.

An illustration of how the inner bounding box is selected from the first mask is provided in Fig. 2. The values within the mask are projected onto the x- and y-axis respectively (blue curves) and the mean, μ_x, μ_y, and standard deviations, σ_x, σ_y, of each axis are computed. The inner bounding box is placed at (μ_x, μ_y) with width $1.8\sigma_x$ and height $1.8\sigma_y$. Also, the outer bounding box, the one being reported as tracking result, is centred around (μ_x, μ_y). The size of the inner bounding box is fixed throughout the sequence unless reinitialized or rescaled, see next section.

There is a major positive effect of using an inner region where size and placement are estimated adaptively. Even if the detector (user, annotator) has marked a region larger than the actual object, the tracker will still select a trackable region instead of being confused by too much background in the object model.

4.4. Scale change estimation

Correct estimation of object scale changes is crucial for correct tracking of an object if the object is subject to large scale variations. This is, for example, the case if the object extends or reduces its distance to the camera. We propose to exploit the probability mass within the foreground mask, $b(i, j)$, in order to detect scale changes of the tracked object. If the scale of the object increases, the probabil-



(a) (b) (c)

Figure 3: Scaling principle examples: (a) Horse mask with overlaid grid lines. (b) The probability mass (pm) within each grid cell has increased, thus, the size should be increased. (c) Another object enters from the right, and the pm in the rightmost grid cells have increased. The pm in the lower left cell has not, therefore, the size should not be increased.

ity mass within the mask will also increase. However, if another foreground object passes behind or in front of the tracked object, the probability mass within the object patch will also increase. Therefore, the flow of probability mass must be considered. A scale change implies probability mass changes in all directions while another object entering the tracked area will cause the probability mass to change on one side of the patch only. The flow of probability mass is roughly estimated by dividing the image patch into a grid of 2×2 cells, see Fig. 3. If all grid cells have increased their mean probability mass from time $t - s_w$ to t, where s_w is a constant time interval, the scale of the patch is increased by a scale step s_s. The opposite applies to the case of decreasing probability mass. In order to achieve robustness to noise in the probability mass, the mean probability mass value of each grid cell is updated using an update factor and the scale step, s_s, is kept relatively low. Further, in order to avoid rounding errors, the width, I, of the bounding box is updated iff $mod(\Delta I, 2) = 0$ and the height, J, iff $mod(\Delta J, 2) = 0$.

4.5. Combining the three methods (ABCD)

The three proposed methods are independent of each other and can thus easily be combined. The resulting tracker is called ABCD – Adaptive object region + Background weighted scaled Channel coded Distribution field tracking. In the next section, evaluation results for A-EDFT, B-EDFT, the combination AB-EDFT as well as ABCD (AB-EDFT + scale estimation) will be presented.

4.6. Combining with a detector for initialization

The ABCD tracker is a pure tracker, relying on an initial (external) detection. While ABCD is general in the sense that it could be used for any object class, it is specific in the sense that each instance models and tracks a specific object. A detector for initialization is typically specific for the application scenario; in the case of the PETS 2016 scenario, we need to detect boats.

For that reason, an anomaly detector, detecting potential targets that do not fit the distribution of sea pixels, has been implemented. By noticing that the apparent temperature of the sea decreases linearly (with an offset and plenty of noise) with the vertical observation angle, fitting a plane to the image intensity values and marking all pixels being warmer than this plane (with a margin of 5–7 K), a surprisingly efficient anomaly detector is acquired. In addition, small targets, detections in the sky, and spurious detections are removed. Small targets according to a threshold varying with distance and spurious detections by using M/N logic before initialising a track.

When a track is initialised, an instance of the ABCD tracker is created for the specific target being tracked. The anomaly detector is utilised to confirm that the ABCD tracker is still on track. A track that is not confirmed by a detection in five frames is thus declared to be lost. The major effect of this step is to avoid tracking wakes.

5. Evaluation and results

In this section, the evaluation procedure and experimental results are presented. Three datasets have been used for evaluation. The LTIR-dataset[2] [3], the PETS2016 dataset, and four maritime sequences from the EU FP7 project IPATCH, similar but not identical to the PETS2016 maritime imagery. LTIR is a thermal infrared dataset consisting of 20 sequences that has recently been used in the thermal infrared visual object tracking VOT-TIR2015 challenge [6]. LTIR and the maritime sequences are labelled according to the VOT-annotation procedure. That is, there are axis aligned bounding box annotations for each object and frame, local per-frame annotations as well as global per-sequence annotations.

5.1. Evaluation methodology

The evaluation of trackers has been performed in accordance with the VOT evaluation procedure using the VOT-evaluation toolkit[3]. The tracker is initialized with the ground truth bounding box in the first frame and the performance is evaluated using four performance measures; accuracy, robustness, speed, and ranking. The accuracy at time t, A_t, measures the overlap between the bounding boxes given by the annotated ground truth O_t^G and the tracker O_t^T as

$$A_t = \frac{O_t^G \cap O_t^T}{O_t^G \cup O_t^T} \qquad (6)$$

Robustness measures the failure rate, *i.e.*, the number of times $A_t = 0$ during a sequence. When a failure is detected,

[2] http://www.cvl.isy.liu.se/research/datasets/ltir/

[3] https://github.com/vicoslab/vot-toolkit

	ρ_A	ρ_R	$\hat{\Phi}$
ABCD	0.65	1.30	0.32
AB-EDFT	0.65	1.55	0.31
A-EDFT	0.61	1.70	0.28
B-EDFT	0.65	2.80	0.22
EDFT	0.61	3.00	0.22

Table 1: Average accuracy (ρ_A), robustness (average number of failures) (ρ_R), and expected average overlap ($\hat{\Phi}$) on the VOT-TIR2015 dataset.

the tracker is reinitialized five frames later, and for another 10 frames the achieved accuracy is not included when per-sequence accuracy is computed. Per-sequence accuracy is calculated as the average accuracy for the set of valid frames in all experiment repetitions. The per-experiment measures, ρ_A and ρ_R, are the average accuracy and robustness (number of failures) over all sequences. Finally, the trackers are ranked for each attribute and an average ranking for the two experiments is computed.

The expected average overlap, $\hat{\Phi} = \langle \Phi_{N_s} \rangle$, was introduced in VOT2015. The measure combines accuracy and robustness by averaging the average overlaps, Φ, on a large set of N_s frames long sequences. Φ_{N_s} is the average of per-frame overlaps, Φ_i:

$$\Phi_{N_s} = \frac{1}{N_s} \sum_{i=1:N_s} \Phi_i. \quad (7)$$

If the tracker fails at some point during the N_s frames long sequence, it is not reinitialized.

5.2. Experiments

Five different experiments have been performed. First, the EDFT tracker and the extensions A-EDFT, B-EDFT, AB-EDFT, and ABCD, all of which have been proposed in this paper, are evaluated on the VOT-TIR2015 dataset. Second, the ABCD and EDFT trackers are evaluated on the VOT2015 (RGB sequences) and VOT-TIR2015 (TIR sequences) datasets against other participating trackers in the challenges. Third, the ranking results of the ABCD tracker are added to the confusion matrix in [6]. Fourth, the ABCD and SRDCFir [6] (winner of VOT-TIR2015) trackers are evaluated on four maritime TIR sequences. Finally, the ABCD tracker in combination with an anomaly detector is applied to the PETS 2016 dataset, resulting in qualitative results only.

5.3. Results

Experiment 1. Table 1 lists the average accuracy and average number of failures per sequence for the EDFT tracker as well as the proposed extensions A-EDFT, B-EDFT, AB-EDFT, and ABCD on the VOT-TIR2015 dataset. It is clear that B-EDFT mainly improves accuracy while A-EDFT improves robustness. When both extensions are combined in AB-EDFT, the robustness is improved even further. Adding the ability to scale to AB-EDFT in ABCD improves robustness without a decrease in accuracy.

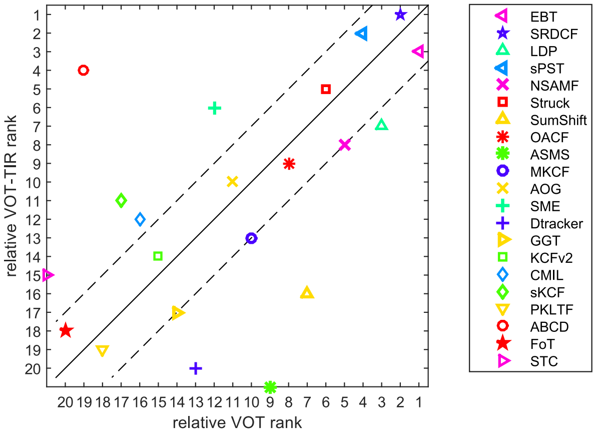


Figure 4: Comparison of relative ranking of 21 trackers in VOT and VOT-TIR.

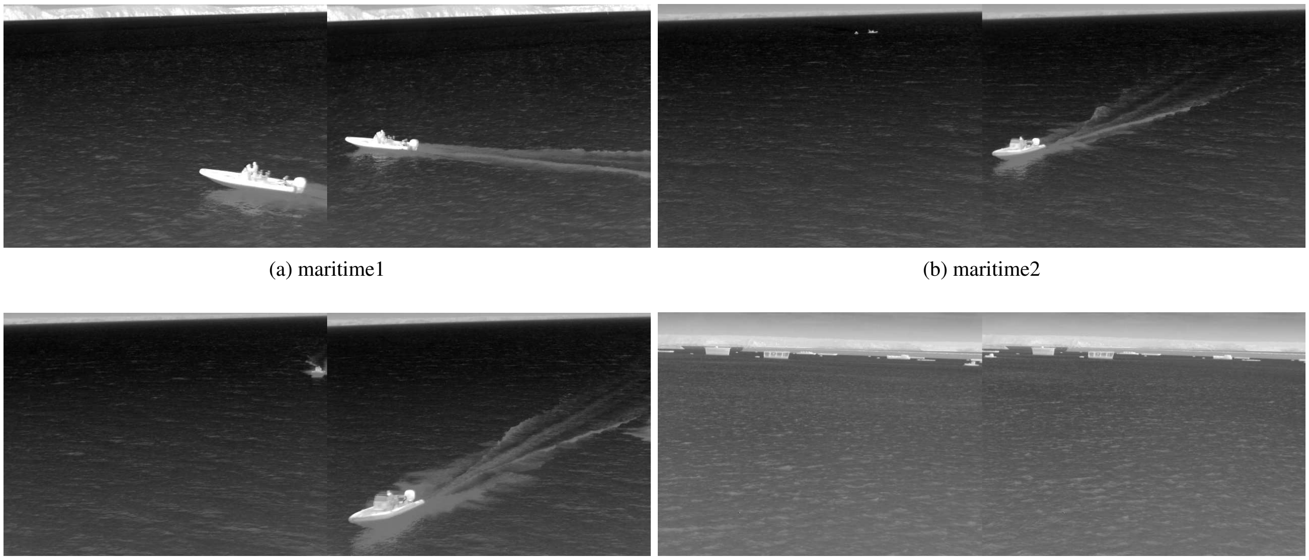
In total, the combined extensions (ABCD) give an increase in average accuracy of 6.6%, a decrease of the average number of failures of 57%, and an increase of the expected average overlap of 45% compared to EDFT.

Regarding speed, the extension from EDFT to ABCD benefits by exploiting information that is already computed, *i.e.*, the channel coded image. That is, the improvement in accuracy and robustness implies no significant addition in computation time. In the VOT-TIR2015 experiments, the ABCD tracker achieved a tracking speed of 6.88 in EFO (equivalent filter operations) units [6]. The VOT toolkit reports the tracker speed in terms of a predefined filtering operation that is automatically performed prior to running the experiments. To put EFO units into perspective, a C++ implementation of a NCC tracker provided in the toolkit runs with average 140 frames per second on a laptop with an Intel Core i5-2557M processor, which equals to approximately 160 EFO units. The ABCD tracker is implemented in Matlab.

Experiments 2 and 3. Table 2 shows the average accuracy, number of failures, and rankings of the EDFT and ABCD trackers when compared to the participating trackers in the VOT2015 and VOT-TIR2015 challenges. The extension of the EDFT tracker provides a significant improvement in ranking for the VOT-TIR2015 challenge while remaining unchanged for the VOT2015 challenge. A comparison of the relative rankings for all common trackers in VOT2015 and VOT-TIR2015, including ABCD, are shown

	VOT2015				VOT-TIR2015			
	ρ_A	ρ_R	$\hat{\Phi}$	r	ρ_A	ρ_R	$\hat{\Phi}$	r
ABCD	0.45	3.12	0.14	50	0.65	1.30	0.32	6
EDFT	0.45	3.50	0.14	49	0.61	3.00	0.22	15

Table 2: Average accuracy (ρ_A), robustness (average number of failures) (ρ_R), expected average overlap ($\hat{\Phi}$), and ranking (r) for the EDFT and ABCD trackers in the baseline experiment of the VOT2015 and VOT-TIR2015 challenges respectively.



(a) maritime1

(b) maritime2

(c) maritime3

(d) maritime4

Figure 5: First and last frame of the four maritime sequences used for evaluation.

in Fig. 4.

Experiment 4. In addition to the VOT2015 and VOT-TIR2015 datasets, the ABCD and SRDCFir (winner of VOT-TIR2015) trackers have been evaluated on four thermal infrared maritime sequences originating from the EU FP7 IPATCH project using the VOT evaluation toolkit. SRDCFir is a correlation filter based tracker that adapts the SRDCF approach proposed in [4] to thermal infrared data by employing other features, like channel coded intensity values. The first and last frame of the four sequences can be seen in Fig. 5. Sequence *maritime1* contains only small scale variations while the boat in *maritime2* and *maritime3* approaches from far away. In contrast, *maritime4* contains a small target against a versatile background. Both trackers proved to be robust on the maritime dataset, SRDCFir had no failures while ABCD had an average number of failures of 0.25. Regarding accuracy, SRDCFir had an average accuracy of 0.71 while ABCD had an average accuracy of 0.60. The SRDCFir tracker had a slightly higher accuracy for sequence *maritime1–3* while ABCD had a marginally higher accuracy on *maritime4*.

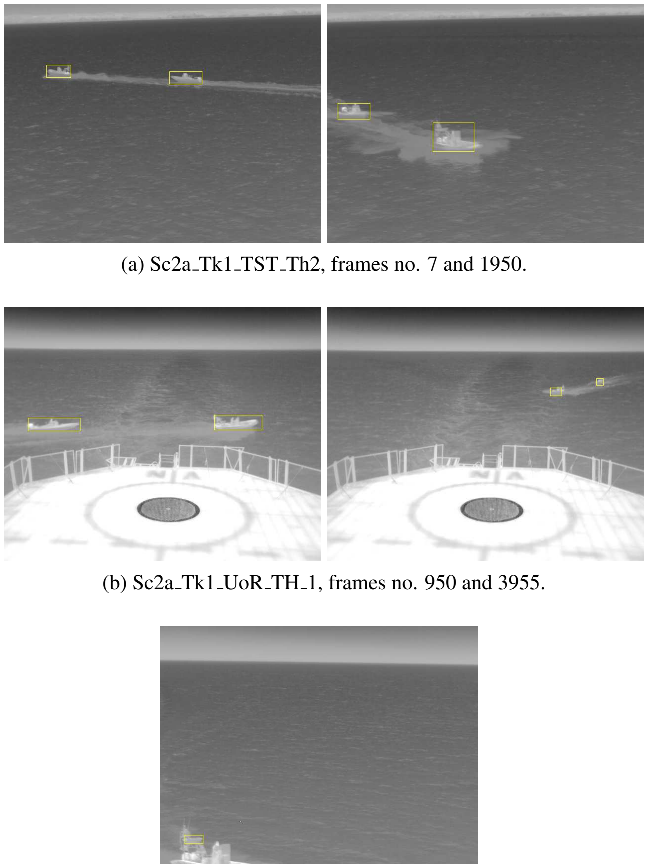
Experiment 5. The ABCD tracker combined with the anomaly detector was applied to the three thermal image sequences in the PETS 2016 dataset. The results are shown in Table 3. Since the ground truth is not part of the PETS 2016 dataset, all numbers come from visual inspection and, hopefully, sound judgment of what constitutes a successful track. A few frames are shown in Fig. 6. For the first sequence, detection and tracking is done fully satisfactory. For the second sequence, there is a short false track for a few frames on the engine of one of the skiffs. For the third sequence, the track initialization does not work as intended; for the ABCD to get a good template, tracks are initialized only for objects that are detached from the image borders. In the third sequence, the target is always connected to the left or bottom image border, and the detector instead locks on a part of the skiff separated from its main part (see Fig. 6c).

5.4. Discussion

The results of the evaluation of the different proposed extensions of the EDFT tracker presented in Table 1 indicate that the combination of extensions (ABCD) is favourable to each individual extension itself. Further, when the ABCD

Sequence	Targets	Detected	False
Sc2a_Tk1_TST_Th2	4	4	0
Sc2a_Tk1_UoR_TH_1	4	4	1
Sc3_Tk2_TST_Th1	1	1	0

Table 3: Tracking results on the PETS 2016 dataset. The number of targets to be detected/tracked in each sequence, the number of targets that were successfully detected and tracked, and the number of false tracks created during the sequence.



(a) Sc2a_Tk1_TST_Th2, frames no. 7 and 1950.

(b) Sc2a_Tk1_UoR_TH_1, frames no. 950 and 3955.

(c) Sc3_Tk2_TST_Th1, frame 5350.

Figure 6: Tracking results in the PETS 2016 dataset.

and EDFT trackers were evaluated on the VOT2015 dataset, Table 2 and Fig. 4, it became clear that the extensions are particularly beneficial to thermal infrared data. Thermal infrared imagery contains less distinct edges and structures for the tracker to attach to. Therefore, the ABCD tracker was designed to exploit the absolute values of the object rather than relying on the spatial structure. A distribution field approach was utilized for this purpose. Background information was incorporated in the template update phase as well as in the choice of an inner bounding box, reducing the background contamination of the object model.

Further, it should be emphasised that the size of targets differ in the VOT2015 and VOT-TIR2015 datasets. VOT has, in general, more high-resolution targets than VOT-TIR. Higher resolution targets provide more spatial structure for trackers to exploit, making them more easily tracked for trackers employing such features.

On the limited maritime dataset, the SRDCFir tracker had better accuracy for the three sequences that had more high-resolution targets. ABCD performed better for targets with low resolution without much spatial structure. Since SRDCFir is designed for sequences with more high-resolution targets and more structure, the results are as expected.

6. Conclusions

We have compared trackers based on spatial structure features with trackers based on distribution fields and come to the conclusion that distribution fields are more suitable for TIR tracking. Since the state-of-the-art distribution field tracker (EDFT) was not able to adapt to changing object scale, we have developed a novel method to achieve adaptivity. Moreover, we make the observation that template-based trackers have two inherent problems of background information contaminating the template of the object to be tracked; in the search phase and in the template update phase. We propose how to mitigate both these problems by exploiting a channel coded background distribution and show that this improves the tracking. The resulting tracker, ABCD, has been evaluated on both RGB and TIR sequences and the results show that the proposed extensions are particularly beneficial for TIR imagery. In addition, ABCD performs comparably to the VOT-TIR2015 winner on maritime data. We have also shown an example of how to combine the ABCD tracker with an anomaly detector for the specific scenario in the PETS 2016 dataset, and successfully track the targets in the image sequences.

Acknowledgements

The research was funded by the Swedish Research Council through the project Learning Systems for Remote Thermography, grant no. D0570301, as well as by the European Community Framework Programme 7, Privacy Preserving Perimeter Protection Project (P5), grant agreement no. 312784, and Intelligent Piracy Avoidance using Threat detection and Countermeasure Heuristics (IPATCH), grant agreement no. 607567. The Swedish Research Council has also funded the research through a framework grant for the project Energy Minimization for Computational Cameras (2014-6227) and the work has been supported by ELLIIT, the Strategic Area for ICT research, funded by the Swedish Government.

References

[1] M. S. Alam and S. M. A. Bhuiyan. Trends in correlation-based pattern recognition and tracking in forward-looking infrared imagery. *Sensors*, 14(8):13437, 2014.

[2] A. Bal and M. S. Alam. Automatic target tracking in FLIR image sequences using intensity variation function and template modeling. *IEEE Transactions on Instrumentation and Measurement*, 54(5):1846–1852, Oct 2005.

[3] A. Berg, J. Ahlberg, and M. Felsberg. A thermal object tracking benchmark. In *Advanced Video and Signal Based Surveillance (AVSS), 2015 12th IEEE International Conference on*, 2015.

[4] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg. Learning Spatially Regularized Correlation Filters for Visual Tracking. In *ICCV*, 2015.

[5] M. Felsberg. Enhanced Distribution Field Tracking using Channel Representations. In *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW), 2013*, pages 121–128. IEEE, 2013.

[6] M. Felsberg, A. Berg, G. Häger, J. Ahlberg, M. Kristan, J. Matas, A. Leonardis, L. Čehovin, G. Fernandez, and et al. The Thermal Infrared Visual Object Tracking VOT-TIR2015 challenge results. In *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, pages 639–651, 2015.

[7] M. Felsberg, P.-E. Forssén, and H. Scharr. Channel smoothing: Efficient robust smoothing of low-level signal features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(2):209–222, February 2006.

[8] P.-E. Forssén. *Low and Medium Level Vision using Channel Representations*. PhD thesis, Linköping University, Sweden, SE-581 83 Linköping, Sweden, March 2004. Dissertation No. 858, ISBN 91-7373-876-X.

[9] R. Gade and T. Moeslund. Thermal cameras and applications: A survey. *Machine Vision & Applications*, 25(1), 2014.

[10] R. Gade and T. B. Moeslund. Thermal tracking of sports players. *Sensors*, 14(8):13679–13691, 2014.

[11] E. Goubet, J. Katz, and F. Porikli. Pedestrian tracking using thermal infrared imaging. In *SPIE Conference on Infrared Technology and Applications*, volume 6206, pages 797–808, June 2006.

[12] G. H. Granlund. An associative perception-action structure using a localized space variant information representation. In *Proceedings of Algebraic Frames for the Perception-Action Cycle (AFPAC)*, Kiel, Germany, September 2000.

[13] Y.-J. He, M. Li, J. Zhang, and J.-P. Yao. Infrared target tracking via weighted correlation filter. *Infrared Physics and Technology*, 73:103–114, Nov. 2015.

[14] C. M. Johnston, N. Mould, J. P. Havlicek, and G. Fan. Dual domain auxiliary particle filter with integrated target signature update. In *Computer Vision and Pattern Recognition Workshops, 2009. CVPR Workshops 2009. IEEE Computer Society Conference on*, pages 54–59, June 2009.

[15] K. Jüngling and M. Arens. Local feature based person detection and tracking beyond the visible spectrum. In R. Hammoud, G. Fan, R. W. McMillan, and K. Ikeuchi, editors, *Machine Vision Beyond Visible Spectrum*, volume 1 of *Augmented Vision and Reality*, pages 3–32. Springer Berlin Heidelberg, 2011.

[16] M. Kristan et al. The Visual Object Tracking VOT2013 Challenge Results. In *The IEEE International Conference on Computer Vision (ICCV) Workshops*, December 2013.

[17] M. Kristan et al. The Visual Object Tracking VOT2014 challenge results. In *Workshop on Visual Object Tracking Challenge (VOT2014) - ECCV*, LNCS, pages 1–27. Springer, Sept. 2014.

[18] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, L. Čehovin, G. Fernández, T. Vojíř, G. Nebehay, R. Pflugfelder, and G. Häger. The Visual Object Tracking VOT2015 challenge results. In *ICCV workshop on VOT2015 Visual Object Tracking Challenge*, 2015.

[19] F. Lamberti, A. Sanna, and G. Paravati. Improving robustness of infrared target tracking algorithms based on template matching. *IEEE Transactions on Aerospace and Electronic Systems*, 47(2):1467–1480, April 2011.

[20] S. Lee, G. Shah, A. Bhattacharya, and Y. Motai. Human tracking with an infrared camera using a curve matching framework. *EURASIP Journal on Advances in Signal Processing*, 2012(1), 2012.

[21] C. Padole and L. Alexandre. Wigner distribution based motion tracking of human beings using thermal imaging. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 9–14, June 2010.

[22] C. N. Padole and L. A. Alexandre. Motion based particle filter for human tracking with thermal imaging. *Emerging Trends in Engineering & Technology, International Conference on*, 0:158–162, 2010.

[23] E. Parzen. On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):pp. 1065–1076, 1962.

[24] J. Portmann, S. Lynen, M. Chli, and R. Siegwart. People Detection and Tracking from Aerial Thermal Views. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[25] A. Prati, I. Mikic, M. M. Trivedi, and R. Cucchiara. Detecting moving shadows: algorithms and evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(7):918–923, July 2003.

[26] L. Sevilla-Lara and E. G. Learned-Miller. Distribution fields for tracking. In *CVPR*, pages 1910–1917. IEEE, 2012.

[27] P. Skoglar, U. Orguner, D. Törnqvist, and F. Gustafsson. Pedestrian Tracking with an Infrared Sensor using Road Network Information. *EURASIP Journal on Advances in Signal Processing*, 1(26):2012a–, 2012.

[28] Y. Wu, J. Lim, and M.-H. Yang. Online object tracking: A benchmark. *CVPR*, 0:2411–2418, 2013.

[29] Y. Wu, J. Lim, and M.-H. Yang. Object tracking benchmark. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PP(99):1–1, 2015.

[30] D. P. Young and J. M. Ferryman. PETS metrics: On-line performance evaluation service. In *Proceedings of the 14th International Conference on Computer Communications and Networks*, ICCCN '05, pages 317–324, 2005.