Semi-automatic Annotation of Objects in Visual-Thermal Video

Amanda Berg[1,2], Joakim Johnander[1,3], Flavie Durand de Gevigney[4],
Jörgen Ahlberg[1,2], Michael Felsberg[1]
[1]Computer Vision Laboratory, Linköping University, Sweden
[2]Termisk Systemteknik AB, Sweden
[3]Zenuity AB, Sweden
[4]Grenoble INP, France
{amanda.,jorgen.ahl}berg@termisk.se, {amanda.,jorgen.ahl,michael.fels}berg@liu.se,
joakim.johnander@{zenuity,liu}.se, flavie.duranddegevigney@grenoble-inp.org

Abstract

Deep learning requires large amounts of annotated data. Manual annotation of objects in video is, regardless of annotation type, a tedious and time-consuming process. In particular, for scarcely used image modalities human annotation is hard to justify. In such cases, semi-automatic annotation provides an acceptable option.

In this work, a recursive, semi-automatic annotation method for video is presented. The proposed method utilizes a state-of-the-art video object segmentation method to propose initial annotations for all frames in a video based on only a few manual object segmentations. In the case of a multi-modal dataset, the multi-modality is exploited to refine the proposed annotations even further. The final tentative annotations are presented to the user for manual correction.

The method is evaluated on a subset of the RGBT-234 visual-thermal dataset reducing the workload for a human annotator with approximately 78% compared to full manual annotation. Utilizing the proposed pipeline, sequences are annotated for the VOT-RGBT 2019 challenge.

1. Introduction

Manual ground truth annotation of video sequences is a labour-intensive process, which is inevitable in many computer vision applications. In the case of deep learning, where large amounts of training data are required [38], the need for efficient annotation is particularly important. In recent years, many annotation tools that facilitate the process have emerged [16, 33, 48]. Semi-automatic annotation methods assist the user by proposing initial annotations later corrected by the user.

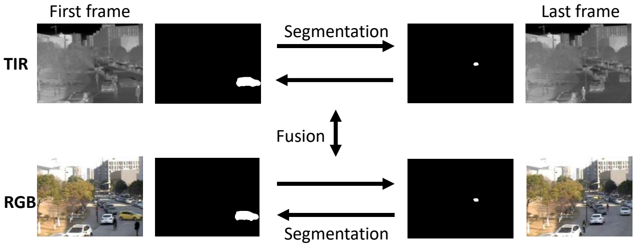


Figure 1: Overview of the proposed method. The video object segmentation algorithm is applied both forward and backward in both thermal (TIR) and RGB before fusion between the two modalities.

In this paper, a novel, recursive, semi-automatic annotation method is proposed. Based on only a few initial manual segmentations, a state-of-the-art video object segmentation (VOS) algorithm [22] proposes segmentations for all frames in a video sequence. Our method recursively recommends where additional manual annotations are needed based on forward-backward segmentation consistency, and the VOS algorithm is run from the additional annotations. Thus, the proposed method generates dense segmentation labels in a video, based only a handful manually annotated frames. In the case of a multi-modal dataset, segmentation results for all modalities are merged. The proposed annotations serve as a tentative annotation set to be further refined by the user, considerably reducing the workload compared to full manual annotation.

From dense segmentation labels, it is possible to extract a number of different types of annotations, e.g., center points, and axis-aligned or rotated bounding boxes. As an example, the final corrected segmentation masks of the evaluation dataset are utilized to generate rotated ground truth bounding boxes for the VOT-RGBT 2019 tracking challenge.

Contributions

- A novel, recursive semi-automatic annotation method for, but not limited to, multi-modal video.
- A novel, automatic failure detection method based on forward-backward consistency.
- A dataset annotated with rotated bounding boxes that will be utilized in the VOT-RGBT 2019 tracking challenge.

2. Background

2.1. Related Work

The process of ground truth annotation has become a fundamental task in the development of computer vision applications. Due to its time-consuming nature, various tools and strategies to facilitate the annotation task have emerged. Popular annotation tools are *e.g.* CVAT [33], ViPER [16], and LabelMe [48]. Other options are crowd sourcing [38] or family members [4]. Depending on the annotation type, the effort can be reduced by, for example, interpolation between so called key frames [8]. Through the right strategy it is possible to reduce the annotation time significantly, *e.g.* by marking extreme points instead of bounding box corners [34].

Semi-automatic approaches to facilitate the annotation process exist for many of the annotation tasks, for example, human activity recognition [9], video object detection and segmentation [41], semantic tagging of large corpora [15], and animal behaviour [23]. Most available methods are interactive and require a human-in-the-loop approach. There are methods that assist the user during the annotation process on a *frame-based* level and methods that assist the user on a *sequence-based* level. Examples of frame-based methods are PolygonRNN [11], PolygonRNN++ [1], and ByLabel [37] for object instance segmentation.

The proposed method assists the user on a sequence-based level. Examples of other sequence-based methods are [2, 7, 8, 31, 44]. Bianco *et al.* [7] combines several methods to semi-automatically propose annotations, while Biresaw *et al.* [8] (iVAT) uses automated tracking and other computer vision methods together with interpolation for assisting manual annotation. Manen *et al.* [31] propose a semi-automatic annotation method for multi object tracking in video, where the user tracks the object with the cursor and these weak annotations are transformed to dense box trajectories. Another method is proposed by Adhikari *et al.* [2], where a model is trained based on manual annotations on the first half of the dataset and the model then proposes annotations for the second half. Wang *et al.* [44] uses an object detector and tracker to facilitate bounding box annotation.

Several of the previous methods are specific to RGB and cannot be used directly on thermal (TIR) data. The default approach to object segmentation in TIR is thresholding based on temperature [5]. This approach does, however, assume that the object has an evenly distributed temperature, that the object maintains its temperature over time, and that no other object in the scene has the same temperature. In most cases, the situation is more complex and a more advanced segmentation method is needed.

Video object segmentation (VOS) is the task of tracking *and* segmenting one or multiple target objects in a video sequence, given a first frame annotation. In recent years, the interest for this problem has surged, and the overall understanding of and performance on the task improves every year [3, 10, 12, 13, 14, 19, 20, 28, 32, 35, 42, 45, 47]. Today, there are computationally efficient and accurate methods that can track and segment a given target given sufficiently benevolent conditions. They are, however, still prone to drift or failure in tougher scenarios, such as during occlusions or significant appearance changes.

The transferability of visual video object segmentation methods to thermal infrared data has been investigated by Yoon *et al.* [39]. They concluded that their object segmentation method, trained on visual images, was transferable to thermal infrared video by just fine-tuning the method on the first frame. The two modalities, TIR and visual, are known to be complementary in many cases [21, 30], *e.g.* for object tracking [40], detection [29], as well as segmentation [49]. In this work, a video object segmentation method trained on visual images was applied to both TIR and visual images and the segmentation results were merged.

2.2. The Visual Object Tracking Challenge

The Visual Object Tracking (VOT) challenge [24, 25, 26] was introduced in 2013 as a challenge for single object, short term tracking. Since then, it has been arranged annually and become one of the most respected benchmarks for short-term tracking methods. In 2015 the VOT Thermal Infrared challenge (VOT-TIR) [17] using the LTIR [6] dataset, was launched as the first thermal infrared short-term tracking benchmark. The dataset was updated in [18] but the VOT committee decided to go towards RGBT for VOT2019, thus a new dataset is needed.

Li *et al.* [27], proposed a visual spectrum (RGB) - thermal infrared (TIR) tracking benchmark (RGBT-234) that utilizes the VOT evaluation toolkit. The dataset consists of 234 TIR and RGB video pairs of varying length. In total, the dataset contains around 234K frames. The available annotations are axis-aligned bounding boxes. In order for the dataset to be useful in the VOT-RGBT 2019 challenge, rotated bounding boxes are needed, but manual annotation had been prohibitively expensive. This problem has been addressed by the proposed method.

3. Method

We propose to semi-automate the annotation process via automatic annotation of easy parts of a video. Difficult parts that require additional manual annotation are automatically detected. The proposed algorithm is recursive in nature, where in each iteration high-impact manual annotations are requested and subsequently utilized to improve the automatic annotation. Pseudo-code of the algorithm is provided in Algorithm 1 and an overview is given below.

Algorithm 1 The proposed recursive method for semi-automatic annotation of one sequence.

```
$S = \{\}$
for $m \leftarrow 1$ to $M$ do
    $S_m = \{\}$
    $A_{m,1} \leftarrow$ manualAnnotation(1)
    $A_{m,J} \leftarrow$ manualAnnotation($J$)
    $S_m^f \leftarrow$ VOSforward($A_{m,1}$)
    $S_m^b \leftarrow$ VOSbackward($A_{m,J}$)
    $S_m \leftarrow \{S_{m,1}^f\} + \{S_{m,J}^b\}$
    $k = 1$
    $j_{k,m} \leftarrow$ findFailure($S^m$)
    while failure found do
        $A_{m,j} \leftarrow$ manualAnnotation($j_{k,m}$)
        $S_j^f \leftarrow$ VOSforward($A_{m,j}$)
        $S_j^b \leftarrow$ VOSbackward($A_{m,j}$)
        $S_m \leftarrow S_m + \{S_{m,j}^f\} + \{S_{m,j}^b\}$
        $k \leftarrow k + 1$
        $j_{k,m} \leftarrow$ findFailure($S^m$)
    $S \leftarrow S + \{S_m\}$
$T \leftarrow$ manualCorrection(merge($S$))
```

3.1. Overview

The proposed pipeline recursively suggests a frame index $j_{k,m}$ to be manually annotated for each iteration k and modality m, based on forward-backward consistency of previously segmented frames. Frames are segmented using a video object segmentation (VOS) method. Manually annotated frames $A_{m,j}$ serve as initialization for the VOS algorithm [22], see Section 3.2. The VOS algorithm is applied both forward (VOSforward) and backward (VOSbackward) starting at $A_{m,j}$, see illustrative examples in Figures 1 and 2. The resulting set of segmentations S_j^f and S_j^b are appended to the set of segmentations S_m for that modality.

The proposed pipeline is initialized by running the VOS algorithm forward from the first $A_{m,1}$ and backward from the last $A_{m,J}$ frame of the sequence. In each iteration, failures are automatically detected based on the current set of multiple segmented frames $S_m = \{I_{j,m,n}^f, I_{j,m,n}^b\}$ by exploiting the consistency as described in Section 3.3. $I_{j,m,n}^f$ and $I_{j,m,n}^b \in \{0,1\}^{W \times H}$ are $W \times H$ binary segmentation masks for forward and backward runs respectively. $j = 1, ..., J$ is the frame index, $m = 1, ..., M$ the modality index, and $n = 1, ..., N_m$ the number of segmentation masks for frame j and modality m. In each iteration k and modality m, there will be $k + 2$ manually annotated segmentations for that sequence. At the same time, there will be $k + 2$ segmentations by the video object segmentation method for each frame j. See example in Figure 2. This also implies that the total number of segmentations per frame j and modality will be $N_m = K_m + 2$ where K_m is the total number of iterations needed for modality m.

The final set of semi-automatically annotated frames, $T = \{T_1, ..., T_J\} \in \{0,1\}^{W \times H}$, is found by manual correction of the fused segmentation results of all modalities and iterations, further described in Section 3.4. An overview can also be seen in Figure 1.

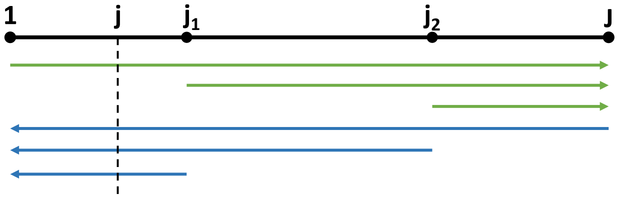


Figure 2: The black, solid, line represents frames in a sequence of length J. There are four manually annotated segmentation masks at frames $1, j_1, j_2, J$. The VOS algorithm is run both forward (green lines) and backward (blue lines) from each manual segmentation, resulting in a total of four segmentation masks ($N_m = 4$) for each frame j in the sequence. The number of iterations is two ($k = 2$).

3.2. Video Object Segmentation

In this work, we utilize the state-of-the-art method for video object segmentation (VOS) proposed in [22], which achieves high performance at low computational cost. It is a deep neural network, comprising a feature extractor, a tracking module, and an upsampling module. The tracking module is a recurrent neural network specifically constructed for the VOS task, and the core idea is to generatively model the deep feature generation, conditioned on whether the feature corresponds to foreground or background. Based on the given annotated frame, the parameters of this model is inferred in closed form as a layer within the network. In subsequent frames, the model is used to relocate and segment the target by calculating the posterior class probabilities. Thus, the parameters of the generative model are not part of the neural network parameters, and are instead learnt online for a given target. The neural network parameters are trained with respect to this process and with the VOS datasets DAVIS2017 [36] and YouTubeVOS [46]. Note that while no TIR data was used for training of the VOS method, we experienced it to perform sufficiently well also on TIR data.

3.3. Failure Detection

We propose to detect failures based on the consistency between forward and backward applications of the VOS algorithm. The idea is that if both the forward and backward runs are correctly segmenting, their predictions will overlap. If either of them loses a part of the target or begins to segment a distractor, the overlap will decrease - and if either or both of the methods completely fail, the overlap should in most cases be equal to zero. The forward-backward consistency is further described in Section 3.3.1. We hypothesize that the best choice for manual annotation is the center of a cluster of failed frames, see Section 3.3.2.

3.3.1 Forward backward consistency

The proposed method for failure detection is based on a consistency score $c_{m,j}$. The frame- and modality-wise consistency score $c_{m,j} \in [0,1]$ for modality $m = 1, ..., M$ and frame $j = 1, ..., J$ is found via minimization of:

$$\sum_{m=1}^{M} |c_{m,j} - o^{fb}_{m,j}| + |\prod_{m=1}^{M}(c_{m,j}) - o^{f}_{j}| + |\prod_{m=1}^{M}(c_{m,j}) - o^{b}_{j}| \tag{1}$$

We motivate (1) by arguing that a frame in timestep j and modality m has consistent annotations if i) the closest forward, and closest backward passes agree (term 1); ii) the closest forward pass segmentations of different modalities agrees (term 2); and iii) if the closest backward pass segmentations of different modalities agree (term 3). Each term is further described below.

The first term in (1), $\sum_{m=1}^{M} |c_{m,j} - o^{fb}_{m,j}|$ enforces *forward-backward consistency*. $o^{fb}_{m,j} \in [0,1]$ is defined as the intersection over union between all forward and backward segmentations for modality m and frame j as in:

$$o^{fb}_{m,j} = \text{IoU}(I^{f}_{m,j}, I^{b}_{m,j}) = \frac{I^{f}_{m,j} \cap I^{b}_{m,j}}{I^{f}_{m,j} \cup I^{b}_{m,j}} \tag{2}$$

where

$$I^{f}_{m,j} = \bigcup_{n=1}^{N^{f}_{m,j}} I^{f}_{m,j,n} \tag{3}$$

and

$$I^{b}_{m,j} = \bigcup_{n=1}^{N^{b}_{m,j}} I^{b}_{m,j,n} \tag{4}$$

in the case of multiple forward/backward segmentations per frame. An illustration is provided in Figure 3. $I^{f}_{m,j}$ and $I^{b}_{m,j} \in \{0,1\}^{W \times H}$ are the binary segmentation masks for the forward and backward frames respectively. $N^{f}_{m,j}$ is the number of forward segmentations for modality m and frame j and $N^{b}_{m,j}$ is the number of backward segmentations. Following the reasoning about the total number of segmentations per frame in Section 3.1, $N_m = N^{f}_{m,j} + N^{b}_{m,j} = K_m + 2$.

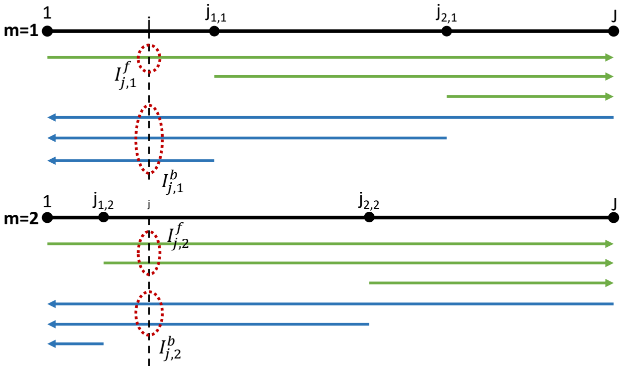


Figure 3: The black line represents frames in a sequence of length J for modalities $m = 1$ and $m = 2$. There are four manual segmentation masks at frames $1, j_{m,1}, j_{m,1}, J$. The VOS algorithm is run both forward (green lines) and backward (blue lines) from each manual segmentation resulting in a total of four segmentations for each frame j in the sequence. The segmentation masks used in the calculation of $I^{f}_{m,j}$ and $I^{b}_{m,j}$ are marked with red circles.

The second term in (1), $|\prod_{m=1}^{M}(c_{m,j}) - o^{f}_{j}|$ enforces *forward consistency* over modalities. Note that the product $\prod_{m=1}^{M}$ is only calculated over consistency scores $c_{m,j}$. The forward overlap $o^{f}_{j} \in [0,1]$ is defined as the intersection over union between the forward segmentations $I^{f}_{m,j}$ of different modalities:

$$o^{f}_{j} = \text{IoU}(I^{f}_{1,j}, I^{f}_{2,j}, ..., I^{f}_{M,j}) = \frac{\bigcap_{m=1}^{M} I^{f}_{m,j}}{\bigcup_{m=1}^{M} I^{f}_{m,j}} \tag{5}$$

Similarly, the third term in (1), $|\prod_{m=1}^{M}(c_{m,j}) - o^{b}_{j}|$ enforces *backward consistency* over modalities. The backward overlap $o^{b}_{j} \in [0,1]$ is defined as:

$$o^{b}_{j} = \text{IoU}(I^{b}_{1,j}, I^{b}_{2,j}, ..., I^{b}_{M,j}) = \frac{\bigcap_{m=1}^{M} I^{b}_{m,j}}{\bigcup_{m=1}^{M} I^{b}_{m,j}} \tag{6}$$

The minimizer of (1) is obtained with the MATLAB routine `fmincon`, using the default interior-point algorithm.

3.3.2 Frame annotation proposal

Given a consistency score vector c_m for all frames $j = 1, ..., J$, a frame index $j_{k,m}$ for iteration k and modality m

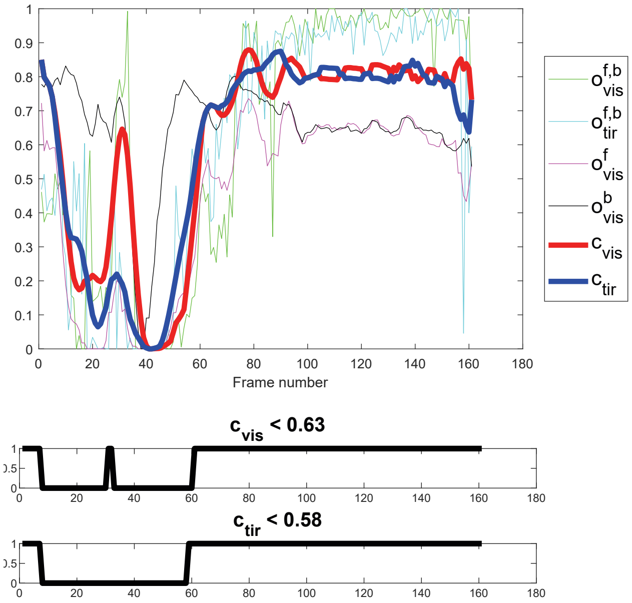


Figure 4: Example of failure detection in the sequence `twowomen` after initialization (i.e. two manual annotations, first and last frame). The red line shows the visual consistency score c_{vis} and the blue line the thermal consistency score c_{tir}. $\alpha_{vis} = 0.63$, $\alpha_{tir} = 0.58$, and the method proposes an additional manual annotation at frame 47 for RGB and frame 33 for TIR.

is proposed for manual annotation. In order to filter out noise, c_m is initially smoothed by a moving average filter with support seven. Scores ($c_{m,j}$), elements of vector c_m, below a threshold α_m are considered to be failure areas, and the frame index of the center of the largest cluster of failed frames is selected as $j_{k,m}$, see Figure 4. We propose to place the annotation at the center of a failure cluster based on the hypothesis that this will maximize the utility.

Note that the score vector, c_m, is calculated separately for each modality m (each element $c_{m,j}$ is calculated as in (1)) which implies that the proposed index $j_{k,m}$ does not have to (but can) be equal for the different modalities.

The threshold α_m is found automatically for each iteration k based on the assumption that the distribution of the consistency scores will be approximately bimodal. The two largest local maxima $p_1, p_2 \in [0, 1]^Z$ (where Z is the total number of local maxima) of the distribution of all consistency scores $c_{m,j}$ are found and α_m is set to the mean between these as in:

$$\alpha_m = \frac{p_1 + p_2}{2}. \tag{7}$$

An illustrative example is given in Figure 5. Another option could be to use e.g. Otsu's algorithm.

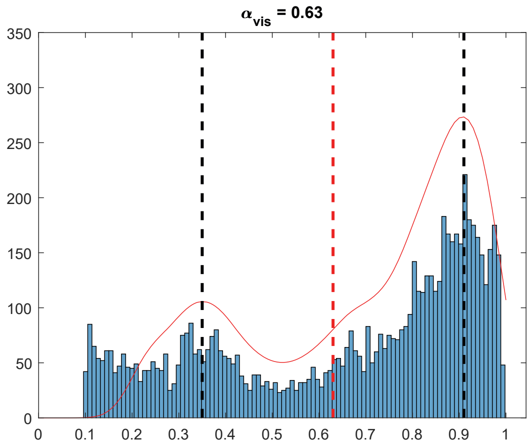


Figure 5: Example of the automatic selection of threshold α_m for the visual modality. Black dotted lines mark the two local maxima p_1, p_2 of the distribution of consistency scores $c_{m,j}$ (red solid line) and the red dotted line marks α_m. Blue bars are histogram bins for the consistency scores.

3.4. Modality Fusion

As exemplified in Figure 2, for each modality at iteration k, there will be $k + 2$ segmentations for each frame in the sequence. Thus, the total number of segmentations per frame for modality m will be $N_m = K_m + 2$ where K_m is the total number of iterations needed for that modality. These segmentations need to be fused in order to form one single segmentation for that frame. Since video object segmentation accuracy generally decreases with number of frames since the initialization frame, the segmentations are weighted according to how close they are to their initialization frame. The weight $w_{j,m,n}$ for frame j, modality m, and segmentation $n = 1, ..., N_m$ where $N_m = K_m + 2$ following the reasoning above, is defined as:

$$w_{j,m,n} = v^{-d} \tag{8}$$

where d is the distance (in number of frames) from the initialization frame. We use the base $v = 2^{\frac{1}{100}}$ which gives $w_{j,m,n} = 0.5$ after 100 frames.

The elements of a binary segmentation mask $T_j \in \{0, 1\}^{W \times H}$ for frame j and pixel $x = 1, ..., W$ and $y = 1, ..., H$ is then calculated as:

$$T_j^{x,y} = \begin{cases} 1 & \text{if } P_j^{x,y} \geq \gamma \\ 0 & \text{if } P_j^{x,y} < \gamma \end{cases} \tag{9}$$

where

$$P_j^{x,y} = \sum_{m=1}^{M} \sum_{n=1}^{N_m} w_{j,m,n} I_{j,m,n}^{x,y} \tag{10}$$

$I_{j,m,n}^{x,y}$ is an element of the binary segmentation mask $I_{j,m,n} \in \{0,1\}^{w \times h}$, $P_j^{x,y} \in \mathbb{R}$, and the threshold γ is defined as:

$$\gamma = \frac{2}{3 \cdot \sum_{m=1}^{M} N_m} \tag{11}$$

The modality fusion assumes a sufficiently minute synchronization between the different modalities, and deviations will lead to a degradation of the results.

4. Experimental Results and Evaluation

In this section, the performance of the semi-automatic annotation method is evaluated. Performance is measured in terms of mean Intersection over Union (mIoU) between the ground truth axis aligned bounding box and the enclosing axis aligned bounding box around the estimated segmentation. We calculate the mIoU by first taking the mean intersection of union over all frames in one sequence, and then averaging that over all sequences.

In addition, we compare mIoUs for the case where ground truth axis aligned bounding boxes are available to support the video object segmentation and the case where such ground truth is unavailable. As a reference, the mIoU for the manually annotated frames is provided as well.

4.1. Dataset

The dataset used in the evaluation is a subset of the RGBT-234 dataset [27]. The thermal sequences of the full RGBT-234 dataset were labelled with global attributes and clustered according to the VOT standard[1]. Based on the clustering, 60 sequences were automatically chosen. During evaluation, it became clear that in seven of the sequences, the object was too small to be properly segmented by the VOS algorithm, this is a limitation of the proposed semi-automatic annotation method. These sequences were manually segmented for the VOR-RGBT 2019 dataset. The subset used for evaluation of the method was, therefore, reduced even further to 53 sequences. A list of all included sequences is provided in Appendix A. Available ground truth for RGBT-234 is provided as axis aligned bounding boxes which is not sufficient for VOT since rotated bounding boxes are needed.

The mIoU between the available ground truth and the bounding boxes around manually annotated segmentations (first and last frame in each sequence) is around 68% for both TIR and RGB. This suggests that the results from any segmentation based semi-automatic annotation method will not be much higher than this for this particular dataset. The reason for this relatively low mIoU is that the available ground truth is not always placed tight around the object. Also, in case of occlusion, the parts not visible in that frame is included in the bounding box while at the same time not manually annotated/segmented.

Table 1: mIoU results over bounding boxes for recursive 2-split (uniformly spread out frames), 53 sequences. k is the number of iterations and f-TIR and f-RGB are the fused results evaluated against the TIR/RGB ground truth.

k	0	1	3
annotations/seq	2	3	5
TIR	0.178	0.361	0.442
RGB	0.255	0.419	**0.478**
f-TIR	0.319	0.424	0.412
f-RGB	**0.324**	**0.429**	0.414

The synchronization of the two modalities as provided in [27] shows deficits. Ground truth bounding boxes for RGB and TIR have less than 50% overlap in 15% of the frames in the entire dataset. The mean overlap is 75%. For that reason, evaluation was done both with and without fusion and in the fusion case, results were calculated both with the infrared ground truth (f-TIR) and the visual ground truth (f-RGB).

4.2. Failure Detection

Two different approaches for failure detection and how to choose which frames to annotate next are evaluated:

1. The $k + 2$ annotated frames are uniformly spread out.

2. The proposed method, as described in Section 3.3, based on forward/backward consistency.

4.2.1 Uniformly spread out frames

In this experiment, manual annotations were placed at the first and last frames, half the sequence, a quarter of the sequence, and three quarters of the sequence. Mean Intersection over Union is presented in Table 1. The mIoU is degraded when modalities are fused (f-TIR and f-RGB) for $k > 4$.

4.2.2 The proposed method

For the proposed method, the proposed, automatic failure detection method was employed. Results can be seen in Table 2. Also here, mIoU is degraded for $k > 4$ when modalities are fused.

Compared to the approach where frames are uniformly selected, the proposed approach requires less annotations/sequence while achieving similar results. Regarding modality fusion, we believe that the large discrepancy in ground truth between TIR and RGB affects the fused case much stronger than the single modality case.

[1] https://github.com/votchallenge/clustering

Table 2: mIoU results over bounding boxes for the proposed failure detection method, 53 sequences. k is the number of iterations and f-TIR and f-RGB are the fused results evaluated against the TIR/RGB ground truth.

k	0	1	2	3
annotations/seq	2.00	2.97	3.95	4.93
TIR	0.178	0.356	0.405	0.421
RGB	0.255	0.413	**0.446**	**0.458**
f-TIR	0.319	**0.437**	0.424	0.421
f-RGB	**0.324**	**0.437**	0.423	0.420

The benefit of the proposed approach for failure detection is that is does not require any ground truth. However, as the decision is based only on the consistency, it is prone to failure in scenarios where the VOS method makes the same mistake in all modalities.

4.3. The VOT-RGBT 2019 Dataset

The proposed method can be assisted by existing ground truth axis aligned bounding box annotations if available. The search area for the segmentation method is then limited to that of an enlarged region around the ground truth bounding box. This approach reduces the number of drift cases and was the employed approach when producing the VOT-RGBT 2019 dataset. Mean Intersection over Union can be seen in Table 3.

The proposed method produces object segmentations. For the VOT-RGBT 2019 dataset, rotated bounding boxes were required and automatically extracted from the segmentations using [43]. More details on all included sequences together with their global attributes can be seen in Appendix A. Depending on the application, different types of annotations can be extracted. For example, center coordinates, axis-aligned bounding boxes, or the segmentations themselves.

The mean number of frames per sequence is 335. After the semi-automatic annotation, we estimate that about 20% of the frames had to be corrected. Compared to manual annotation of all frames, the proposed method thus reduces the workload with about 78%. That is if correcting frames is equated with manual annotation from scratch in terms of time-consumption, the five manual annotations that were needed are included, and inspection of segmentation results of all frames is excluded.

4.4. Qualitative results

In Figure 6, we present two examples of successful semi-automatic annotations using the proposed method as well as two examples of failure cases. In Figure 6a and 6b, the object is correctly segmented (white outline) and a rotated bounding box (green) encloses the segmentation. We argue that the rotated bounding box is a more accurate annotation of the object than the ground truth axis-aligned bounding box (red) in some cases, e.g. in Figure 6a. In Figure 6b, the ground truth annotation only encloses the person, while the segmentation includes both the person and the bag. Depending on the annotation task, this can either be an advantage or a disadvantage.

Table 3: mIoU results over bounding boxes for the proposed method where a bounding box refinement technique was used to limit the search area for the VOS algorithm. The method was evaluated on 53 sequences. k is the number of iterations and f-TIR and f-RGB are the fused results evaluated against the TIR/RGB ground truth.

k	0	1	2	3
annotations/seq	2	3	4	5
TIR	0.184	0.355	0.420	0.446
RGB	0.256	0.395	0.459	0.476
f-TIR	0.333	0.462	0.491	0.496
f-RGB	**0.339**	**0.467**	**0.496**	**0.500**

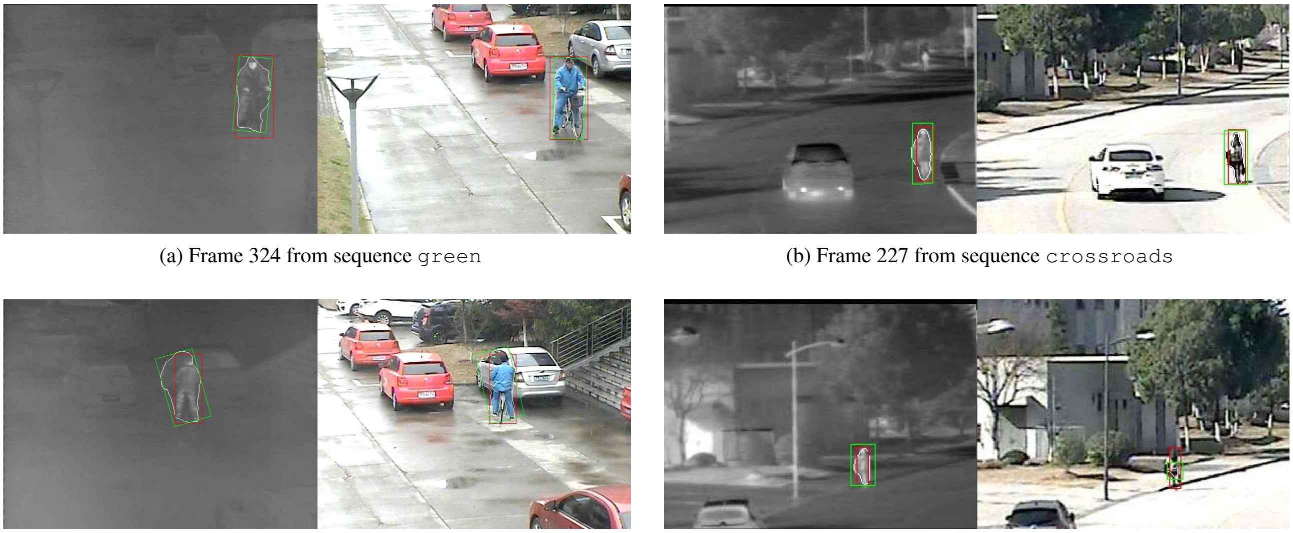
The employed VOS-algorithm sometimes includes background in the object segmentation, example in Figure 6c. In this example, the same failure happens in both TIR and RGB, which does not have to be the case. Figure 6d shows an example of when the VOS-algorithm fails to segment the whole object.

5. Conclusion

We have proposed a recursive semi-automatic annotation pipeline utilizing a Video Object Segmentation (VOS) algorithm to automatically propose tentative segmentations for frames in multi-modal video. Tentative segmentations are corrected by a human annotator, significantly reducing the workload compared to full manual segmentation. We show that using only five manual annotations per sequence, workload can be reduced with about 78%. The final segmentations can be use to generate a range of different types of annotations, e.g. center points, axis aligned bounding boxes, or rotated bounding boxes.

The proposed pipeline is preferably combined with our novel, automatic, failure detection method based on forward-backward consistency also proposed in this work. The latter is especially beneficial in the case of sequences with difficult passages that are non-uniformly distributed.

We also propose, in the case of available axis-aligned bounding boxes, to use these bounding boxes to assist the VOS algorithm and limit the search space in order to reduce the number of drift cases. This approach was utilized in the creation of the VOT-RBGT234 dataset and led to a



(a) Frame 324 from sequence `green`

(b) Frame 227 from sequence `crossroads`

(c) Frame 53 from sequence `green`

(d) Frame 612 from sequence `crossroads`

Figure 6: Examples of successful ((a) and (b)) as well as failed ((c) and (d)) semi-automatic annotations in both TIR (left) and RGB (right). The white outline shows our object segmentation, the red box is the ground truth axis aligned bounding box, and the green box is the rotated bounding box based on our segmentation. The frames from sequence `crossroads` have been cropped for the sake of visualization.

reduction of the workload with about 78%.

Future work include incorporation of frame-wise annotations of occlusions in order to prevent the failure detection to get stuck in occlusion areas. Also, the hypothesis regarding RGB-TIR inconsistency affecting the fused results much stronger could be verified in a synthetic experiment.

Acknowledgements

The research was supported by the Swedish Research Council through the project Learning Systems for Remote Thermography (2013-5703), the project ELLIIT (the Strategic Area for ICT research, funded by the Swedish Government), the Wallenberg AI, Autonomous Systems and Software Program (WASP), and the Visual Sweden project n-dimensional Modelling (VS1810-Q).

References

[1] D. Acuna, H. Ling, A. Kar, and S. Fidler. Efficient Interactive Annotation of Segmentation Datasets with Polygon-RNN++. In *CVPR*, 2018. 2

[2] B. Adhikari, J. Peltomaki, J. Puura, and H. Huttunen. Faster Bounding Box Annotation for Object Detection in Indoor Scenes. In *2018 7th European Workshop on Visual Information Processing (EUVIP)*, pages 1–6. IEEE, nov 2018. 2

[3] L. Bao, B. Wu, and W. Liu. CNN in MRF: Video Object Segmentation via Inference in A CNN-Based Higher-Order Spatio-Temporal MRF. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5977–5986, 2018. 2

[4] A. Barriuso and A. Torralba. Notes on Image Annotation. *arXiv:1210.3448*, oct 2012. 2

[5] A. Berg. Detection and Tracking in Thermal Infrared Imagery, 2016. 2

[6] A. Berg, J. Ahlberg, and M. Felsberg. A Thermal Object Tracking Benchmark. In *12th International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6. IEEE, aug 2015. 2

[7] S. Bianco, G. Ciocca, P. Napoletano, and R. Schettini. An Interactive Tool for Manual, Semi-automatic and Automatic Video Annotation. *Computer Vision and Image Understanding*, 131:88–99, feb 2015. 2

[8] T. A. Biresaw, T. Nawaz, J. Ferryman, and A. I. Dell. ViTBAT: Video Tracking and Behavior Annotation Tool. In *2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 295–301. IEEE, aug 2016. 2

[9] P. Bota, J. Silva, D. Folgado, and H. Gamboa. A Semi-Automatic Annotation Approach for Human Activity Recognition. *Sensors*, 19(3):501, jan 2019. 2

[10] S. Caelles, K.-K. Maninis, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. Van Gool. One-shot Video Object Segmentation. In *CVPR 2017*. IEEE, 2017. 2

[11] L. Castrejon, K. Kundu, R. Urtasun, and S. Fidler. Annotating Object Instances with a Polygon-RNN. *CVPR*, apr 2017. 2

[12] Y. Chen, J. Pont-Tuset, A. Montes, and L. Van Gool. Blazingly Fast Video Object Segmentation with Pixel-Wise Metric Learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1189–1198, 2018. 2

[13] J. Cheng, Y.-H. Tsai, S. Wang, and M.-H. Yang. Segflow: Joint Learning for Video Object Segmentation and Optical Flow. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 686–695. IEEE, 2017. 2

[14] H. Ci, C. Wang, and Y. Wang. Video Object Segmentation by Learning Location-Sensitive Embeddings. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 501–516, 2018. 2

[15] S. Dill, J. A. Tomlin, J. Y. Zien, N. Eiron, D. Gibson, D. Gruhl, R. Guha, A. Jhingran, T. Kanungo, S. Rajagopalan, and A. Tomkins. SemTag and Seeker. In *Proceedings of the twelfth international conference on World Wide Web - WWW '03*, page 178, New York, New York, USA, 2003. ACM Press. 2

[16] D. Doermann and D. Mihalcik. Tools and Techniques for Video Performance Evaluation. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, volume 4, pages 167–170. IEEE Comput. Soc. 1, 2

[17] M. Felsberg, A. Berg, G. Häger, J. Ahlberg, M. Kristan, J. Matas, A. Leonardis, L. Cehovin, 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. IEEE, dec 2015. 2

[18] M. Felsberg, M. Kristan, J. Matas, A. Leonardis, R. Pflugfelder, G. Häger, A. Berg, A. Eldesokey, et al. The Thermal Infrared Visual Object Tracking VOT-TIR2016 Challenge Results. pages 824–849. Springer, Cham, 2016. 2

[19] P. Hu, G. Wang, X. Kong, J. Kuen, and Y.-P. Tan. Motion-Guided Cascaded Refinement Network for Video Object Segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1400–1409, 2018. 2

[20] Y.-T. Hu, J.-B. Huang, and A. G. Schwing. VideoMatch: Matching based Video Object Segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 54–70, 2018. 2

[21] Y. Jin, J. Li, D. Ma, X. Guo, and H. Yu. A Semi-Automatic Annotation Technology for Traffic Scene Image Labeling Based on Deep Learning Preprocessing. In *22017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, pages 315–320. IEEE, jul 2017. 2

[22] J. Johnander, M. Danelljan, E. Brissman, F. S. Khan, and M. Felsberg. A Generative Appearance Model for End-to-end Video Object Segmentation. In *CVPR*, 2019. 1, 3

[23] M. Kabra, A. A. Robie, M. Rivera-Alba, S. Branson, and K. Branson. JAABA: Interactive Machine Learning for Automatic Annotation of Animal Behavior. *Nature Methods*, 10(1):64–67, jan 2013. 2

[24] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Čehovin, T. Vojir, G. Bhat, et al. The Sixth Visual Object Tracking VOT2018 Challenge Results. pages 3–53. Springer, Cham, sep 2019. 2

[25] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Čehovin, T. Vojir, G. Häger, et al. The Visual Object Tracking VOT2017 Challenge Results. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 1949–1972. IEEE, oct 2017. 2

[26] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Čehovin, T. Vojír, G. Häger, et al. The Visual Object Tracking VOT2016 Challenge Results. pages 777–823. Springer, Cham, 2016. 2

[27] C. Li, X. Liang, Y. Lu, N. Zhao, and J. Tang. RGB-T Object Tracking:Benchmark and Baseline. may 2018. 2, 6

[28] X. Li and C. Change Loy. Video Object Segmentation with Joint Re-identification and Attention-Aware Mask Propagation. In *The European Conference on Computer Vision (ECCV)*, September 2018. 2

[29] J. Liu, S. Zhang, S. Wang, and D. N. Metaxas. Multispectral Deep Neural Networks for Pedestrian Detection. *arxiv:1611.02644*, nov 2016. 2

[30] J. Ma, Y. Ma, and C. Li. Infrared and Visible Image Fusion Methods and Applications: A Survey. *Information Fusion*, 45:153–178, jan 2019. 2

[31] S. Manen, M. Gygli, D. Dai, and L. Van Gool. PathTrack: Fast Trajectory Annotation With Path Supervision, 2017. 2

[32] K. Maninis, S. Caelles, Y. Chen, J. Pont-Tuset, L. Leal-Taixe, D. Cremers, and L. Van Gool. Video Object Segmentation Without Temporal Information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018. 2

[33] OpenCV. CVAT. {https://github.com/opencv/cvat}, 2019. 1, 2

[34] D. P. Papadopoulos, J. R. R. Uijlings, F. Keller, and V. Ferrari. Extreme Clicking for Efficient Object Annotation. aug 2017. 2

[35] F. Perazzi, A. Khoreva, R. Benenson, B. Schiele, and A. Sorkine-Hornung. Learning Video Object Segmentation from Static Images. In *Computer Vision and Pattern Recognition*, volume 2, 2017. 2

[36] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, and L. Van Gool. The 2017 DAVIS Challenge on Video Object Segmentation. *arXiv:1704.00675*, 2017. 3

[37] X. Qin, S. He, Z. Zhang, M. Dehghan, and M. Jagersand. ByLabel: A Boundary Based Semi-Automatic Image Annotation Tool. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1804–1813. IEEE, mar 2018. 2

[38] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, dec 2015. 1, 2

[39] J. Shin Yoon, F. Rameau, J. Kim, S. Lee, S. Shin, I. So Kweon, and S. Korea. Pixel-Level Matching for Video Object Segmentation using Convolutional Neural Networks. Technical report. 2

[40] M. Talha and R. Stolkin. Particle Filter Tracking of Camouflaged Targets by Adaptive Fusion of Thermal and Visible Spectra Camera Data. *IEEE Sensors Journal*, 14(1):159–166, jan 2014. 2

[41] P. Voigtlaender, M. Krause, A. Osep, J. Luiten, B. B. G. Sekar, A. Geiger, and B. Leibe. MOTS: Multi-Object Tracking and Segmentation. *CoRR*, abs/1902.03604, 2019. 2

[42] P. Voigtlaender and B. Leibe. Online Adaptation of Convolutional Neural Networks for Video Object Segmentation. In *British Machine Vision Conference 2017, BMVC 2017, London, UK, September 4-7, 2017*, 2017. 2

[43] T. Vojir and J. Matas. Pixel-Wise Object Segmentations for the VOT 2016 Dataset. Research Report CTU–CMP–2017–01, Center for Machine Perception, K13133 FEE Czech Technical University, Prague, Czech Republic, January 2017. 7

[44] B.-L. Wang, C.-T. King, and H.-K. Chu. A Semi-Automatic Video Labeling Tool for Autonomous Driving Based on Multi-Object Detector and Tracker. In *2018 Sixth International Symposium on Computing and Networking (CANDAR)*, pages 201–206. IEEE, nov 2018. 2

[45] S. Wug Oh, J.-Y. Lee, K. Sunkavalli, and S. Joo Kim. Fast Video Object Segmentation by Reference-Guided Mask Propagation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7376–7385, 2018. 2

[46] N. Xu, L. Yang, Y. Fan, J. Yang, D. Yue, Y. Liang, B. L. Price, S. Cohen, and T. S. Huang. YouTube-VOS: Sequence-to-Sequence Video Object Segmentation. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part V*, pages 603–619, 2018. 3

[47] L. Yang, Y. Wang, X. Xiong, J. Yang, and A. K. Katsaggelos. Efficient Video Object Segmentation via Network Modulation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2

[48] J. Yuen, B. Russell, Ce Liu, and A. Torralba. LabelMe Video: Building a Video Database with Human Annotations. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1451–1458. IEEE, sep 2009. 1, 2

[49] J. Zhao and S.-c. S. Cheung. Human Segmentation by Geometrically Fusing Visible-light and Thermal Imageries. *Multimedia Tools and Applications*, 73(1):61–89, nov 2014. 2