Exercise 5: Long-Term Tracking

Nikolay Vasilev

I. Introduction

Object tracking is a fundamental aspect of computer vision, which involves detecting and following specific object within a video sequence. Accurate and robust object tracking plays a main role in a lot of use cases, such as: security and surveillance, robotics, augmented reality, etc.

Well known example of object tracking is the **SiamFC** [1] tracker, which has gained significant attention due to its exceptional performance in real-time tracking scenarios. It uses deep learning techniques to achieve high accuracy and efficiency in tracking objects within video sequences.

In this paper, we focus not only on the evaluation of the *short-term* SiamFC tracker, but we also propose an implementation of a *long-term* version of the tracker. We will compare and analyse the performance of both trackers in terms of Precision, Recall, and F-score metrics on a chosen long-term sequence called *car9*. Additionally, we will try to determine optimal confidence threshold for initiating and terminating re-detection processes. We will also investigate the impact of different numbers of randomly sampled regions during re-detection on the tracker's ability to re-detect the target within fewer frames, and compare different sampling strategies, such as random sampling and Gaussian sampling around the last confident position.

II. Experiments

We will start our experiment by setting up and running the short-term SiamFC tracker. As we can see in Table I, we have pretty average Precision score, but relatively low Recall. This means that the SiamFC tracker, in its short-term tracking configuration, can accurately localize the target in considerable number of cases, but at the same time have problems with effectively tracking the target in a lot of cases.

Table I
Evaluation of short-term SiamFC tracker on the long-term video sequence $car9$.

Precision	Recall	F-score
0.64	0.27	0.38

By implementing the long-term SiamFC tracker, we expect to improve the object re-detection, adaptability to appearance changes, extended tracking duration and in general Recall performance. But before we can evaluate our implementation of the long-term SiamFC tracker, we have to figure out, which parameters give us best results.

In Table II is presented the impact of the $failure\_threshold$ variable on the performance of the long-term SiamFC, which is evaluated on the long-term video sequence $car9$. As we can see, we are again using the metrics Precision, Recall and F-score, but we also have additional metric that help us find out, how many frames are needed for the tracker to re-detect the target. The value of the parameter don't really seem to have an impact on the Precision, Recall and F-score, but it changes the number of frames needed to re-detect the target. We see that the value 2 gives us less frames. Of course having less frames is not always good, because it may also mean that the target was not accurately detected, which is also the case here, since the threshold is too small. The value 3.5 takes a bit more frames, but also gives us better performance, which is more important for our tracker.

Table II
Impact of the $failure\_threshold$ on the performance of the long-term SiamFC tracker on the long-term video sequence $car9$.

Parameter	Precision	Recall	F-score	#Frames
$failure\_threshold = 2$	0.60	0.59	0.59	23
$failure\_threshold = 3.5$	0.60	0.59	0.60	27
$failure\_threshold = 5$	0.60	0.59	0.60	35

Now, when we took a look at the $failure\_threshold$, it is only logically to investigate the re-detection threshold. In our implementation, we do not have a fixed value for the re-detection threshold, but we adapt it each frame the target is visible, depending on the mean value of all previous correlations, the following way:

$$redet\_threshold = mean(correlations) - redet\_factor$$

The parameter $redet\_factor$ has an impact on the $redet\_threshold$, which is why it is the one we are interested in. It becomes clear from Table III that also this parameter does not have a great impact on the values of the Precision, Recall and F-score, but more on the number of frames needed to re-detect the target. The least frames needed to re-detect the target are for $redet\_factor = 1.7$. This value doesn't negatively impact the other metrics, which makes it the optimal value for this parameter.

Table III
Impact of the $redet\_factor$ on the performance of the long-term SiamFC tracker on the long-term video sequence $car9$.

Parameter	Precision	Recall	F-score	#Frames
$redet\_factor = 0.2$	0.60	0.58	0.59	81
$redet\_factor = 0.5$	0.60	0.59	0.60	42
$redet\_factor = 1$	0.60	0.59	0.60	36
$redet\_factor = 1.2$	0.60	0.59	0.60	28
$redet\_factor = 1.5$	0.60	0.59	0.60	29
$redet\_factor = 1.7$	0.60	0.59	0.60	27
$redet\_factor = 2$	0.60	0.59	0.60	28

We have to take into considerations also the impact of the parameter $redet\_samples$ on the performance of the long-term SiamFC tracker. In table IV, we see that there are two values that give us pretty good results. Both 20 and 60 re-detection samples improve our tracker so it re-detect the target in 27 frames, but there is very little difference in there F-scores, which makes us think that $redet\_samples = 20$ is the optimal value for this parameter.

Now, when we have finished our evaluation on the long-term SiamFC tracker using random sampling for re-detection, we can use the optimal parameters and check, if there is any change in the metrics, when we use Gaussian sampling for the re-detection of the target on the long-term video sequence $car9$.

Table IV
Impact of the *redet_samples* on the performance of the long-term SiamFC tracker on the long-term video sequence *car*9.

Parameter	Precision	Recall	F-score	#Frames
$redet\_samples = 20$	0.60	0.59	0.60	27
$redet\_samples = 40$	0.60	0.59	0.60	39
$redet\_samples = 60$	0.60	0.59	0.59	27
$redet\_samples = 80$	0.60	0.59	0.60	39
$redet\_samples = 100$	0.60	0.59	0.60	28

The results in Table V show that there is no difference between the performance of the long-term SiamFC tracker, which is using the random sampling strategy, and the one using the Gaussian sampling strategy. Of course, we must keep in mind that we are only evaluating the tracker on a single long-term video sequence, because of the lack of GPU. It is possible that in other video sequences the strategies would have different performances. In theory, the tracker using random sampling will give better performance on video sequences, where the target shows fast and erratic movements or frequently disappears. In other hand the Gaussian sampling around the last confident position would be more suitable for video sequences, where the target's motion is relatively smooth and predictable. Evaluating the long-term SiamFC tracker on more long-term video sequences will be able to give us more information.

Table V
Impact of different sampling strategies on the performance of the long-term SiamFC tracker on the long-term video sequence *car*9.

Sampling strategy	Precision	Recall	F-score	#Frames
Gaussian	0.60	0.59	0.60	27
Random	0.60	0.59	0.60	27

If we now compare the short-term with the long-term SiamFC tracker, we can see that even though the short-term version has a little higher Precision, the long-term has higher Recall and therefore higher F-score. In Figure 1 it can be seen very clear what happens in the moment, when the target disappears and reappears in the video sequence. As we see on the first row of images, the long-term version of the tracker successfully continuous the tracking of the target, even when the target is not visible. In the same time on the second row of images, since the short-term version of the tracker does not re-detect the target, the tracker loses track of the target in the moment, when it disappears and therefore cannot track it anymore afterwards.

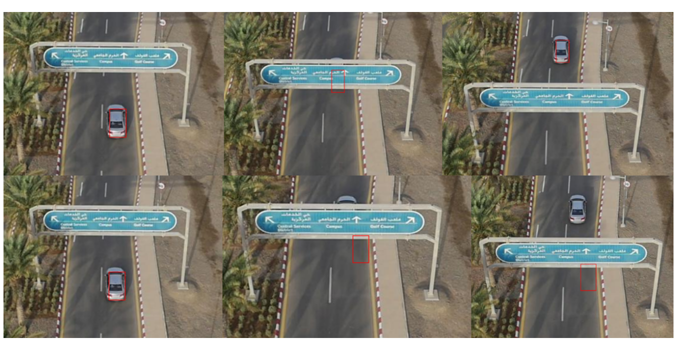

Figure 1. Example of the short-term SiamFC (2nd row) and the long-term SiamFC (1st row) on the long-term video sequence *car*9. The images are taken in the moment of disappearance of the target.

III. Conclusion

This paper has explored the performance of the SiamFC tracker in both short-term and long-term tracking scenarios. The evaluation of the short-term version of the tracker showed its ability to accurately localize the target and its limitations in successfully tracking it. To overcome the limitations of the short-term tracker, we extended it to a long-term by implementing mechanisms for target re-detection. Its evaluation showed an improved performance compared to the short-term version, addressing the challenges of target disappearance and occlusion.

By evaluating the SiamFC tracker and its extension to a long-term tracker we defined its strengths and limitations. Since the tracker was evaluated only on one long-term video sequence, the information we found is not much and not so reliable. This data's validity can be proved by evaluating both versions of the tracker on a whole set of long-term video sequences and therefore calculate average of the Precision, Recall, F-score and needed frames to re-detect the target. This could also help us make new discoveries, which could be helpful in future researches.

References

[1] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. S. Torr, "Fully-convolutional siamese networks for object tracking," in *Computer Vision – ECCV 2016 Workshops*, G. Hua and H. Jégou, Eds. Cham: Springer International Publishing, 2016, pp. 850–865.