Long Term Tracking

Jakob Maležič, 63170191

I. Introduction

Video tracking is the process of locating a moving object over time using a camera. It has a variety of uses, some of which are: human-computer interaction, security and surveillance, video communication and compression, augmented reality, traffic control, medical imaging and video editing [1], [2]. Algorithms for object tracking analyze sequential video frames and output the movement of targets between the frames. A lot of these trackers, work only when the target is always visible. If we want to track objects that can sometimes disappear from the view for brief times, we must use a long term tracker. In this project, we first tested an existing short-term tracker called SiamFC, which extracts the target's template using a deep CNN and then searches for the target based on the correlation response with the template. Then we modified the same tracker to a long-term tracker and tested it on the same set of sequences. We compared the two, reported the results and effects of various parameters on the tracker's performance.

II. Experiments

The tracker returns the new target position and the confidence for this position for each frame. The confidence tells us, how likely it is, that the target appears at the calculated position. To normalize the confidence, we defined it as the maximum correlation response of the frame divided by the correlation response given by the original template. We detected target loss with thresholds for the maximum correlation response. We used two thresholds, one for detecting target loss and another for re-detecting the target. We set the threshold for detecting target loss to 3.5. Re-detection threshold however, we had to dynamically update, because correlation response was always the highest in the beginning of the sequence and without updating the threshold, the threshold would be too high for re-detecting the target later. The threshold was updated every frame and was set to the mean of previous correlation response.

We first ran the short-term SiamFC tracker to get the baseline for comparison with the long-term tracker. For evaluation, we calculated the F-score, which is computed from precision, that tells us the average overlap on frames where predictions are made and recall, which tells us average overlap on frames where target is visible. We then modified the short-term tracker and run it on the same dataset. The results can be seen in table II.

With Gauss sampling method, the tracker reached higher precision, but lower recall and F-score. Precision is probably higher, because the locations for re-detecting are closer to the target, which helps the tracker to find a more precise target location. The recall is lower, because when the target disappears, it can reappear somewhere else, not close to the last seen position.

For re-detecting the target, we had to somehow find the target in the next image. We implemented this, by taking predefined number of templates at random positions and calculating the correlation with the template. The advantage of this approach is that it takes the whole image into the account. This can work well in some cases, where the target can appear anywhere in the image. The disadvantage of this method is that the probability to guess the actual position of the target is small and gets even smaller, if we decrease the number of random templates. Another sampling technique we tried is by taking Gaussian distributed positions around the previously detected position, when the target was still visible. This approach works well for the cases when the target reappears at the similar position where it disappeared. Figure 1 shows an example of re-detection of the target.

Table I
Comparison between the trackers.

sequence	tracker	precision	recall	F-score
car9	short-term	0.64	0.27	0.38
	long-term-gauss	0.60	0.58	0.59
	long-term-random	0.60	0.57	0.59
cat1	short-term	0.84	0.42	0.56
	long-term-gauss	0.81	0.43	0.57
	long-term-random	0.81	0.43	0.57
deer	short-term	0.72	0.20	0.31
	long-term-gauss	0.18	0.20	0.19
	long-term-random	0.18	0.20	0.19
dog	short-term	0.20	0.07	0.11
	long-term-gauss	0.61	0.04	0.07
	long-term-random	0.62	0.03	0.06
person14	short-term	0.67	0.04	0.07
	long-term-gauss	0.46	0.04	0.07
	long-term-random	0.72	0.69	0.7
person20	short-term	0.73	0.73	0.73
	long-term-gauss	0.74	0.74	0.74
	long-term-random	0.74	0.74	0.74
sitcom	short-term	0.54	0.50	0.52
	long-term-gauss	0.47	0.04	0.07
	long-term-random	0.72	0.04	0.07
skiing	short-term	0.55	0.13	0.20
	long-term-gauss	0.63	0.04	0.08
	long-term-random	0.42	0.05	0.09
sup	short-term	0.54	0.48	0.51
	long-term-gauss	0.76	0.66	0.71
	long-term-random	0.54	0.48	0.51
average	short-term	0.61	0.31	0.41
	long-term-gauss	0.71	0.29	0.41
	long-term-random	0.69	0.34	0.45

In table II we can see the number of frames the tracker needed to re-detect the target, based on different number of samples for sequence *car9*. We can see that with more samples, it takes less frames to re-detect the target.

Table II
FPS comparison between sampling methods.

#samples	Random (#frames)	Gauss (#frames)
10	130	80
20	83	61
50	75	56
100	63	54

III. Conclusion

In this project we modified a short-term, CNN-based tracker to work as a long-term tracker. We then tested two different sampling techniques for re-detecting the target. The results for our dataset showed, that random sampling performed better.

Figure 1. Re-detection of the target.

The tracker didn't perform well for some sequences, where the target couldn't be re-detected. This probably happens, because the target template gets extracted at the beginning of the sequence and is never updated. Overall, since the tracker works well on sequences where the target is always visible, the modified tracker can also re-detect the target, which is definitely an improvement.

References

[1] P. Mountney, D. Stoyanov, and G.-Z. Yang, "Three-dimensional tissue deformation recovery and tracking," *IEEE Signal Processing Magazine*, vol. 27, no. 4, pp. 14–24, 2010.

[2] L. Mihaylova, P. Brasnett, N. Canagarajah, and D. Bull, *Advances and Challenges in Multisensor Data and Information*, ser. NATO Security Through Science Series. IOS Press, 2007, vol. 8, pp. 260–268.