Supplemental Material

A Superior Tracking Approach: Building a strong Tracker through Fusion

Christian Bailer[1], Alain Pagani[1], and Didier Stricker[1,2]

[1] German Research Center for Artificial Intelligence, Kaiserslautern, Germany
Christian.Bailer, Alain.Pagani, Didier.Stricker@dfki.de
[2] University of Kaiserslautern, Germany

1 Introduction

In this supplemental material we present and discuss the removal statistics for global removal in Section 2. Heatmaps for SRE and TRE we present in Section 3 and the algorithms for the processing time study we present in Section 4. Furthermore, we show the local stability of our parameters in Section 5. References for algorithms mentioned here can be found in the main paper.

2 Removal Statistics of Global Removal

Here we determine the probabilities that tracking algorithms are removed by our global removal approach. We perform the global removal experiment described in the paper 10 times with different *10 random parts* of the full dataset, each (which are in total 100 experiments). Table 1 shows the probabilities that algorithms are removed in an experiment. The row "Order(OPE)" shows the order in which removal is tested on OPE. As described in the paper, the removal order is equal to the inverse order of the tracking algorithms average performance (w_j). Thus, it it is expectable that algorithms with smaller removal order are in general more likely to be removed as they perform on average worse. However, as Figure 1 shows it is in practice not that simple. There are several algorithms like SMS, VR-V, Frag and LOT that are very removal resistant despite their not so good average performance. On the other hand, there are also good performing algorithms like CSK and VTS that are very likely to be removed. We think this effect originates in a large part from the diversity/originality in behavior that tracking algorithms have among each other. The strengths of algorithms that are often removed are probably already widely covered by other algorithms so that they cannot utilize them in fusion, while their weaknesses will still influence fusion negatively. Removal resistant algorithms on the other hand probably have more original/unique strength that can still be utilized easily by fusion.

To explain this in more detail we define $T^{\setminus x}$ to be the fusion result without the tracking result (or trajectory) T_x. For sequence parts where $T^{\setminus x}$ and T_x are close to each other T_x is obviously not required and there will not be a mentionable difference if it is added. We call this a *covered* sequence part. If the good parts in T_x (where it is correct) are covered, T_x is not able to influence the fusion result positively. However, T_x will still influence the fusion result negatively at frames where it is incorrect. As a consequence this will lead to removal. On the other hand bad algorithms that occasionally produce correct results for uncovered sequence parts will likely not be removed as long as they do not harm fusion too much at frames where they are incorrect. Note that an important property of fusion is to benefit more from correct result than being harmed from wrong results. Thus, fusion can still benefit from tracking algorithms with very bad average performance as long as these can sufficiently utilize their strengths on uncovered sequence parts.

We think the approach described in this section can be an interesting idea for tracker evaluation. It first of all shows if a tracking algorithm has enough diversity to the set of algorithms to be interesting for fusion. However, it is not only interesting for fusion. It also gives an interesting estimate for the originality of the tracking idea used by a tracking algorithm.[3]

Algorithm	ASLA	BSBT	CPF	CSK	CT	CXT	DFT	Frag	IVT	KMS
Order(OPE)	26	6	11	22	4	25	20	12	15	7
OPE	0	75	16	74	90	0	4	5	0	80
TRE	5	42	0	99	93	7	1	1	8	12
SRE	0	9	0	35	97	8	2	0	12	13
Algorithm	L1APG	LOT	LSK	MIL	MS-V	MTT	OAB	ORIA	PD-V	RS-V
Order(OPE)	19	16	21	14	1	18	17	8	5	10
OPE	0	2	20	20	41	0	17	11	21	47
TRE	14	0	2	70	26	68	36	0	0	4
SRE	1	0	0	64	81	24	26	88	2	2
Algorithm	SemiT	SCM	SMS	Struck	TLD	TM-V	VR-V	VTD	VTS	-
Order(OPE)	9	29	2	28	27	13	3	23	24	-
OPE	18	0	8	7	0	8	13	48	29	-
TRE	1	0	5	7	0	64	4	10	82	-
SRE	7	0	21	1	0	7	17	44	95	-

Table 1. Probability in percentage that an algorithm is removed by global removal.

3 Heatmaps for SRE and TRE

Table 3 and 4 show heatmaps for SRE and TRE. Values for full cyan are higher than on OPE (See table descriptions). Table 2 shows the number of sequences where fusion outperforms the best algorithm on the sequence or has at least 95% of the performance of the best algorithm. As can be seen in the table, the

[3] Of course there are also other criteria for originality. CSK can, for example, also be considered as original as it outperforms the algorithms that behave similar to it by far in processing speed.

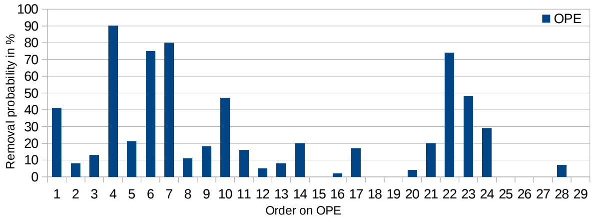


Fig. 1. Removal probabilities of Table 1 on OPE as diagram.

advantage of fusion in these numbers is even bigger for both SRE and TRE than for OPE.

	outperform(>100%)			at least 95%		
	OPE	SRE	TRE	OPE	SRE	TRE
prev work.	3	5	6	12	17	20
Basic	11	15	24	25	28	34
Weighted	15	**20**	**27**	27	28	35
Trajectory	20	**20**	26	33	32	37
Global Removal	18	18	26	**35**	**33**	37
Local Removal	**22**	19	**27**	34	**33**	**39**

Table 2. Table shows on how many sequences different approaches outperform the best tracking algorithm on a dataset or have at least 95% of the performance of the best tracking algorithm. In common there are 51 sequences.

4 Details on Processing Time Evaluation

Processing time evaluation is performed on only 25 of the 29 algorithms as for 4 algorithms there are no processing speeds available. The sets of *fusion selection 1* in Figure 4(a) in the paper are created from right to left with the following algorithms: CSK, CPF, VR, CT, MIL, TLD, IVT, Struck, OAB, CXT, DFT, SMS, SemiT, ASLA, ORIA, VTS, VTD, BSBT, Frag, LSK, KMS, L1APG, MTT, LOT, SCM. The sets of *fusion selection 2* are created from right to left with the following algorithms: CSK, CPF, TLD, Struck, MIL, IVT, VR, CT, CXT, OAB, ASLA, DFT, VTS, VTD, SemiT, LSK, ORIA, Frag, BSBT, SMS, L1APG, KMS, SCM, MTT, LOT.

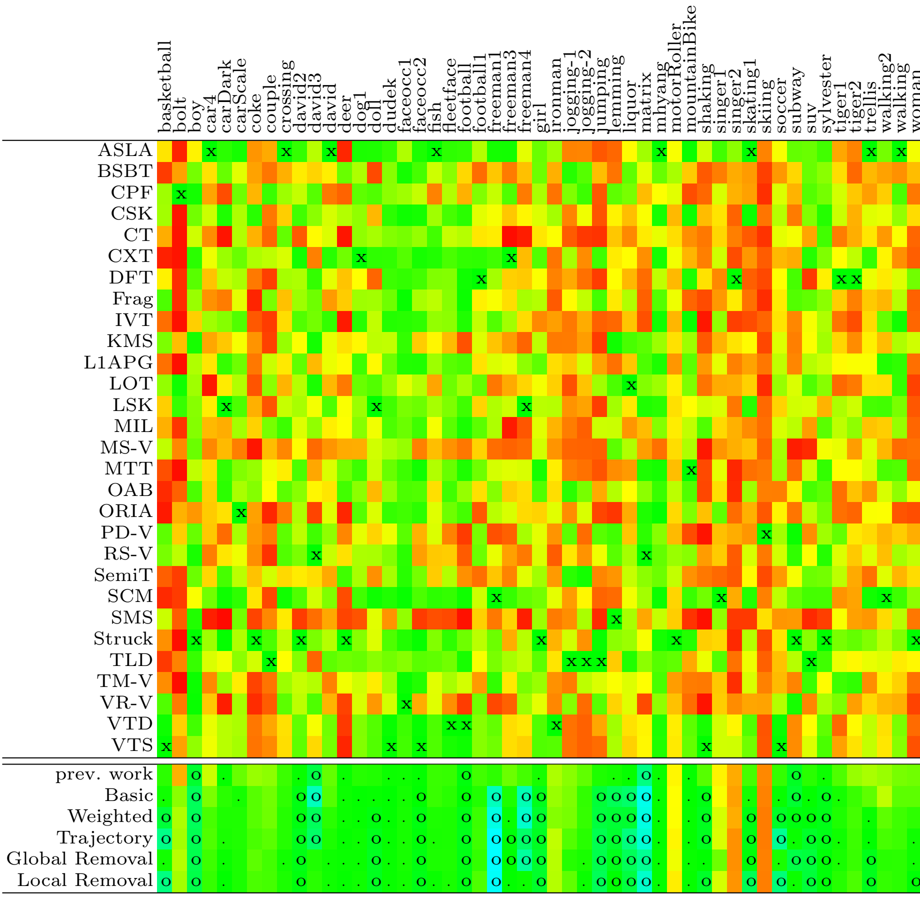


Table 3. Comparison of tracking results and fusion results of SRE. The heatmap is normalized so that the best tracking result on a sequence is green. Red is the worst possible result. Cyan means that the fusion result is up to **18%** (full cyan) better than the best tracking result. "x" marks the best tracking algorithm for a sequence and "o" fusion results that outperform the best algorithm. Heatmap is calculated by success score (see paper for details). Best viewed in color.

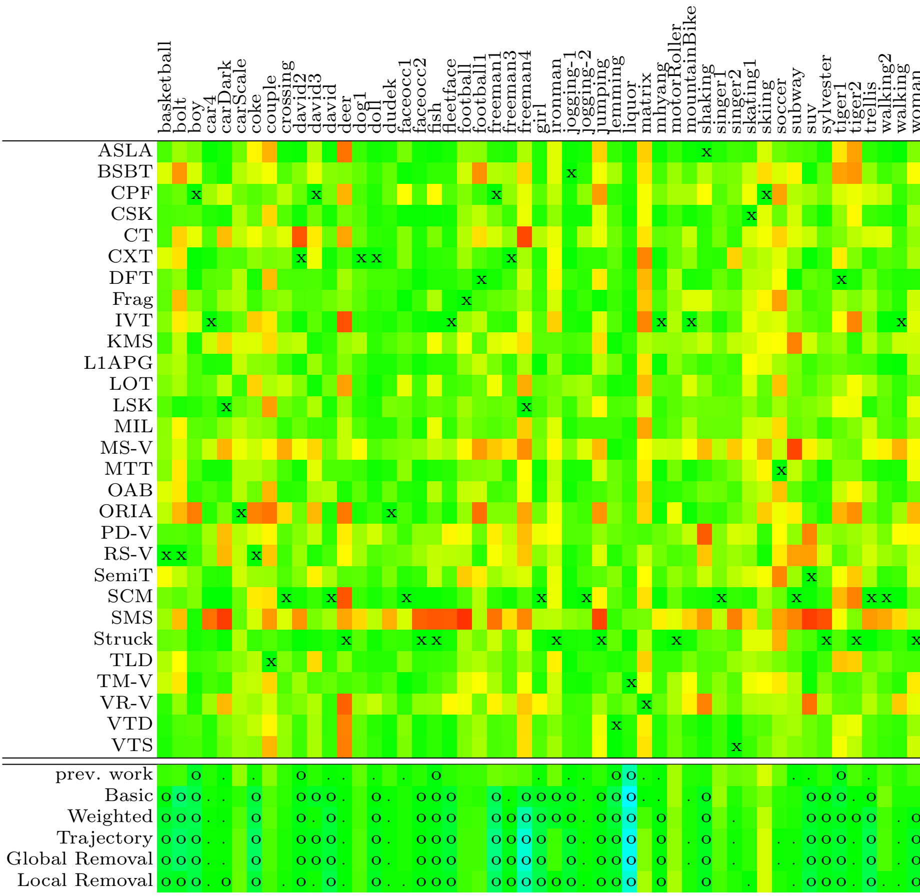


Table 4. Comparison of tracking results and fusion results of TRE. The heatmap is normalized so that the best tracking result on a sequence is green. Red is the worst possible result. Cyan means that the fusion result is up to **33%** (full cyan) better than the best tracking result. "x" marks the best tracking algorithm for a sequence and "o" fusion results that outperform the best algorithm. Heatmap is calculated by success score (see paper for details). Best viewed in color.

5 Stability of Parameters

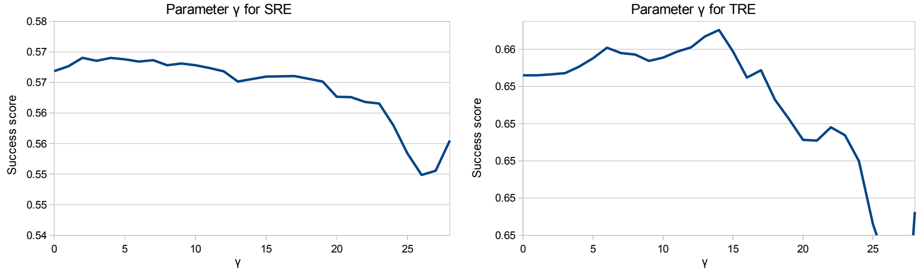
Here we show the behavior of our parameters on one dataset (OPE). Figure 3 shows what happens if our optimal parameters on OPE are changed. All subfigures are created with our local removal approach (based on the trajectory optimization approach for fusion) that uses all 4 parameters.

As can be seen in Figure 3(a) local changes in σ will not change the success score much. In the range of 0.02 - 0.05 all values are a good choice. With a too small σ the curve breaks as this leads to a "*the winner takes all*" strategy. The winner is the tracking algorithm with the highest weight. In Figure 4 we show how σ influences our basic approach that does not use weights. Here the curve does not drop that fast. However, for very small σ it will still drop because of other effects. Too big σ lead to oversmoothing which is also not perfect.

Fusion is also stable against local changes in α as can be seen in Figure 3(b). Too small values lead to overrating of scale i.e. the tracking box position has nearly no influence anymore on the fusion result. Too big values overate position and scale has nearly no influence anymore on the fusion result. Note that scale still has a lot of influence with $alpha = 10$. Without influence the success score will drop to 0.614.

There is also a stable plateau for β in Figure 3(c). Too small values will reduce the influence of trajectory optimization. Too big values can avoid algorithm changes. However, with $\beta = 50$ most algorithm changes seem to be still possible.

γ is the most instable of our parameters. Nevertheless, there is a tendency of increasing success score from the left up to the global maximum. This is also true for TRE and SRE (Figure 2). However, the global maximum of SRE is already at 2. We think this comes from the inaccurate initialization of SRE. On OPE and SRE we can also still benefit from removal even if we remove a few more algorithms than γ at the global maximum.



(a) (b)

Fig. 2. γ for SRE and TRE for our local removal approach based on trajectory optimization.

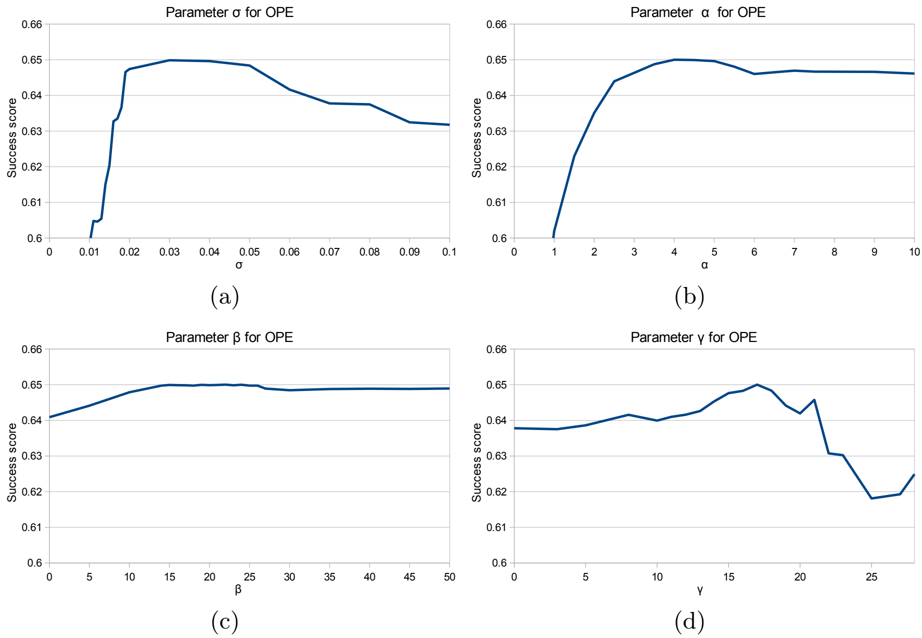


Fig. 3. Figures show how the result changes if our parameters are changed on OPE (with local removal approach, based on trajectory optimization).

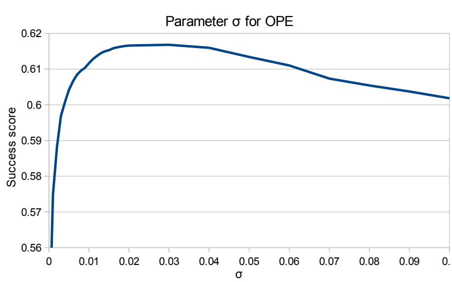


Fig. 4. α for OPE for our basic approach.