Online Object Tracking: A Benchmark
Supplemental Material

Yi Wu
University of California at Merced
ywu29@ucmerced.edu

Jongwoo Lim
Hanyang University
jlim@hanyang.ac.kr

Ming-Hsuan Yang
University of California at Merced
mhyang@ucmerced.edu

We present more evaluation results in this document.

Tracking Speed. Table 1 shows the statistics of the tracking speed of each algorithm in OPE running on a PC with Intel i7 3770 CPU (3.4GHz). The speed of L1APG is slower than [4] as we set the parameters of L1APG to be the default ones of MTT, where the canonical size of template is larger than the default one of L1APG. The implementation of ASLA and SCM are based on IVT and set their parameters for particle filter to be the same.

Spatial Shifts. Figure 1 illustrates eight spatial shifts used in SRE. The amount of shift is 10% of width or height of the ground-truth bounding box.

Sequence Ranking. We rank the difficulty of sequences based on the AUC scores of trackers in SRE. The average of top 5 AUC scores of trackers for each sequence is used for ranking. The performance plots for top 4 most difficult sequences are shown in Figure 2 and Figure 3 (cf. Figure 1 of the manuscript).

Attribute Distribution. The attribute distribution for each subset is shown in Figure 4. The attribute distributions of the whole dataset and OCC subset are shown in Figure 2 of manuscript.

Overall Performance. Figure 5 shows the plots for the overall performance of all the evaluated trackers (cf. Figure 3 of the manuscript).

Attribute-based Performance. The plots for the performance of all the evaluated trackers on the attribute subsets are shown from Figure 6 to Figure 11 (cf. Section 5.2 and Figure 4 of the manuscript).

Initialization with Different Scale. Figure 12 and Figure 13 show the plots of SRE of the trackers initialized with different bounding box sizes (cf. Figure 5 of the manuscript).

Randomness. Numerous tracking algorithms contain components that may be initialized with random seeds, e.g., feature extraction (Haar-like features in MIL), learning algorithm (random forests in CXT), and search mechanism (particle filter). We use the Struck to illustrate this point as it is one of the top performing algorithms. The randomness of Struck lies in its learning algorithm where it needs to randomly select support patterns. We use ten different random seeds to run the SRE, namely, we evaluate Struck for another ten times. More than three million frames are processed in this experiment. Figure 14 shows the success and precision plots for the ten SRE tests. The results indicate the performance of Struck is stable despite it involves some randomness.

Method	FPS-A	F-MAX	S-MAX	F-MIN	S-MIN
CPF [19]	109	284	woman	12.4	fleetface
LOT [18]	0.70	2.0	girl	0.1	fleetface
IVT [20]	33.4	34	woman	32	dudek
ASLA [13]	8.5	9.9	doll	6.4	matrix
SCM [26]	0.51	0.6	carDark	0.4	singer2
L1APG [4]	2.0	3.2	faceocc1	1.1	jogging
MTT [25]	1	1.9	bolt	0.3	basketball
VTD [15]	5.7	9.9	freeman4	3.5	faceocc1
VTS [16]	5.7	9.8	freeman4	3.5	faceocc1
LSK [17]	5.5	18	car4	1.5	football1
ORIA [23]	9.0	27	freeman4	3.2	soccer
DFT [21]	13.2	27	girl	2.8	dudek
KMS [7]	3,159	11,337	freeman3	423	singer1
SMS [5]	19.2	187	crossing	0.3	fleetface
VR-V [6]	109	354	girl	18	singer1
Frag [1]	6.3	15	girl	2.7	walking
OAB [9]	22.4	76	freeman3	4.0	fleetface
SemiT [10]	11.2	46	freeman3	1.9	shaking
BSBT [22]	7.0	39	freeman3	1.6	fleetface
MIL [2]	38.1	53	car4	31	walking
CT [24]	64.4	96	carDark	43	walking
TLD [14]	28.1	138	woman	10	dudek
Struck [11]	20.2	23	singer1	8.3	basketball
CSK [12]	362	1889	freeman3	119	singer2
CXT [8]	15.3	33	subway	3.1	doll

Table 1. Tracking speed. FPS-A: average FPS; F-MAX: maximum FPS; S-MAX: the sequence that achieves maximum FPS; F-MIN: minimum FPS; S-MIN: the sequence that achieves minimum FPS.

Video. Some tracking results are available at: `http://www.youtube.com/user/ID558CVPR2013`. These videos are used to show the sensitivity to initialization of some trackers.

References

[1] A. Adam, E. Rivlin, and I. Shimshoni. Robust Fragments-based Tracking using the Integral Histogram. In *CVPR*, 2006.

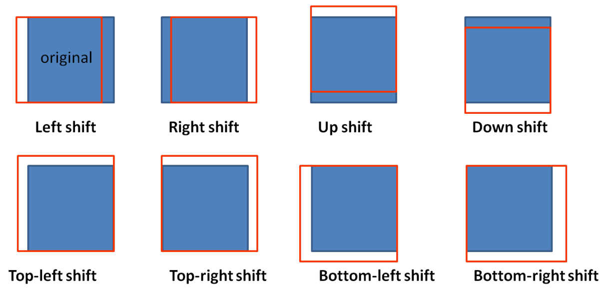

Figure 1. Spatial shifts. The amount of shift is 10% of width or height of the ground-truth bounding box.

[2] B. Babenko, M.-H. Yang, and S. Belongie. Visual Tracking with Online Multiple Instance Learning. In *CVPR*, 2009.

[3] B. Babenko, M.-H. Yang, and S. Belongie. Robust Object Tracking with Online Multiple Instance Learning. *PAMI*, 33(7):1619–1632, 2011.

[4] C. Bao, Y. Wu, H. Ling, and H. Ji. Real Time Robust L1 Tracker Using Accelerated Proximal Gradient Approach. In *CVPR*, 2012.

[5] R. Collins. Mean-shift Blob Tracking through Scale Space. In *CVPR*, 2003.

[6] R. T. Collins, Y. Liu, and M. Leordeanu. Online Selection of Discriminative Tracking Features. *PAMI*, 27(10):1631–1643, 2005.

[7] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-Based Object Tracking. *PAMI*, 25(5):564–577, 2003.

[8] T. B. Dinh, N. Vo, and G. Medioni. Context Tracker: Exploring Supporters and Distracters in Unconstrained Environments. In *CVPR*, 2011.

[9] H. Grabner, M. Grabner, and H. Bischof. Real-Time Tracking via On-line Boosting. In *BMVC*, 2006.

[10] H. Grabner, C. Leistner, and H. Bischof. Semi-supervised On-Line Boosting for Robust Tracking. In *ECCV*, 2008.

[11] S. Hare, A. Saffari, and P. H. S. Torr. Struck: Structured Output Tracking with Kernels. In *ICCV*, 2011.

[12] F. Henriques, R. Caseiro, P. Martins, and J. Batista. Exploiting the Circulant Structure of Tracking-by-Detection with Kernels. In *ECCV*, 2012.

[13] X. Jia, H. Lu, and M.-H. Yang. Visual Tracking via Adaptive Structural Local Sparse Appearance Model. In *CVPR*, 2012.

[14] Z. Kalal, J. Matas, and K. Mikolajczyk. P-N Learning: Bootstrapping Binary Classifiers by Structural Constraints. In *CVPR*, 2010.

[15] J. Kwon and K. M. Lee. Visual Tracking Decomposition. In *CVPR*, 2010.

[16] J. Kwon and K. M. Lee. Tracking by Sampling Trackers. In *ICCV*, 2011.

[17] B. Liu, J. Huang, L. Yang, and C. Kulikowsk. Robust Tracking using Local Sparse Appearance Model and K-Selection. In *CVPR*, 2011.

[18] S. Oron, A. Bar-Hillel, D. Levi, and S. Avidan. Locally Orderless Tracking. In *CVPR*, 2012.

[19] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet. Color-Based Probabilistic Tracking. In *ECCV*, 2002.

[20] D. Ross, J. Lim, R.-S. Lin, and M.-H. Yang. Incremental Learning for Robust Visual Tracking. *IJCV*, 77(1):125–141, 2008.

[21] L. Sevilla-Lara and E. Learned-Miller. Distribution Fields for Tracking. In *CVPR*, 2012.

[22] S. Stalder, H. Grabner, and L. van Gool. Beyond Semi-Supervised Tracking: Tracking Should Be as Simple as Detection, but not Simpler than Recognition. In *ICCV Workshop*, 2009.

[23] Y. Wu, B. Shen, and H. Ling. Online Robust Image Alignment via Iterative Convex Optimization. In *CVPR*, 2012.

[24] K. Zhang, L. Zhang, and M.-H. Yang. Real-time Compressive Tracking. In *ECCV*, 2012.

[25] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja. Robust Visual Tracking via Multi-task Sparse Learning. In *CVPR*, 2012.

[26] W. Zhong, H. Lu, and M.-H. Yang. Robust Object Tracking via Sparsity-based Collaborative Model. In *CVPR*, 2012.

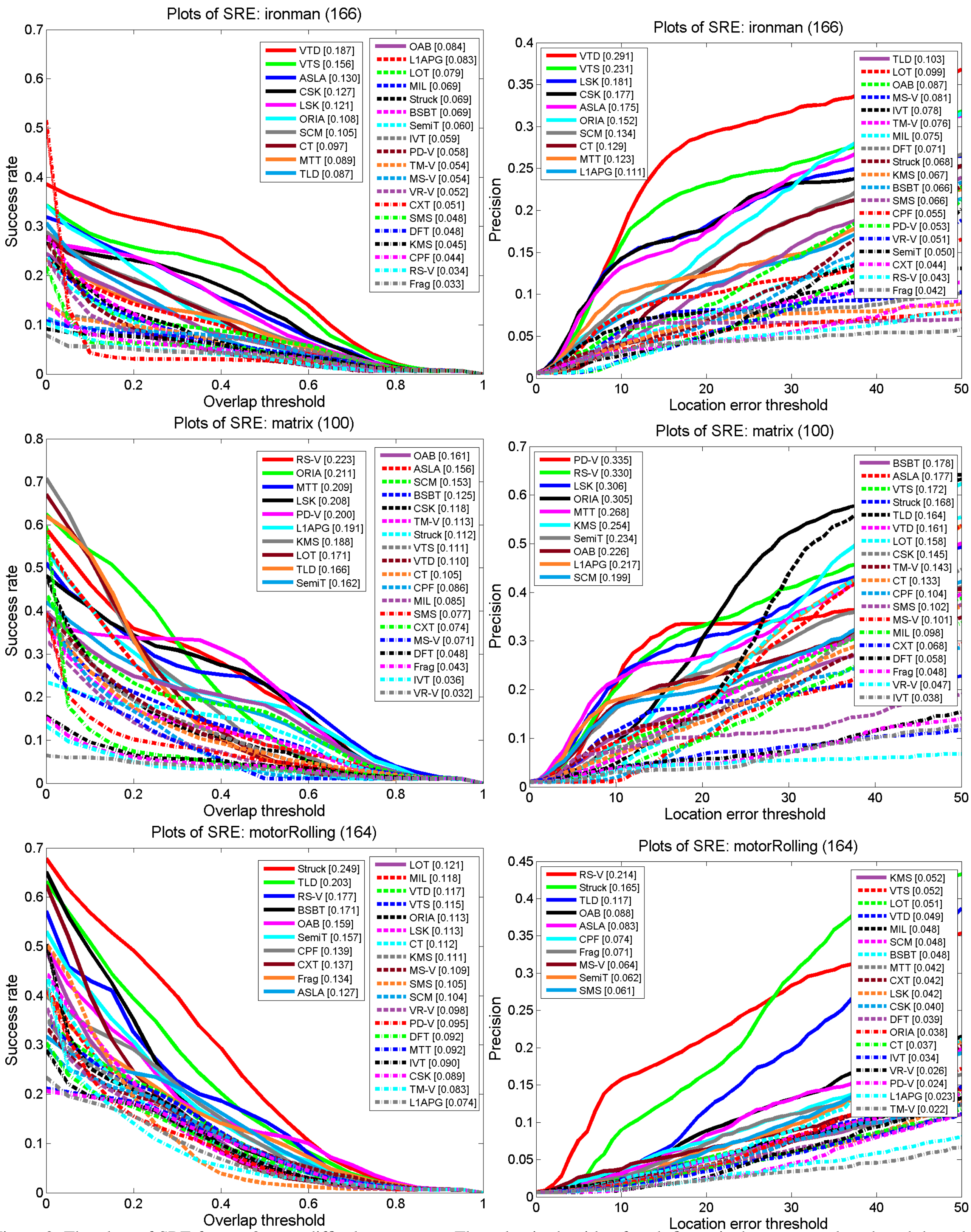

Figure 2. The plots of SRE for top 3 most difficult sequences. The value in the title of each figure is the sequence length and the values in the legend are the performance scores. All the tested trackers perform poorly on these challenging sequences. One reason is many factors coupled in these sequences to make them more challenging for tracking. Another is some attributes affect the appearance of target dramatically, such as the illumination variations in *ironman* and *matrix*. Although in [16], the authors have demonstrated promising results of VTS on these two most difficult sequences, under our SRE they cannot handle these challenging cases robustly.

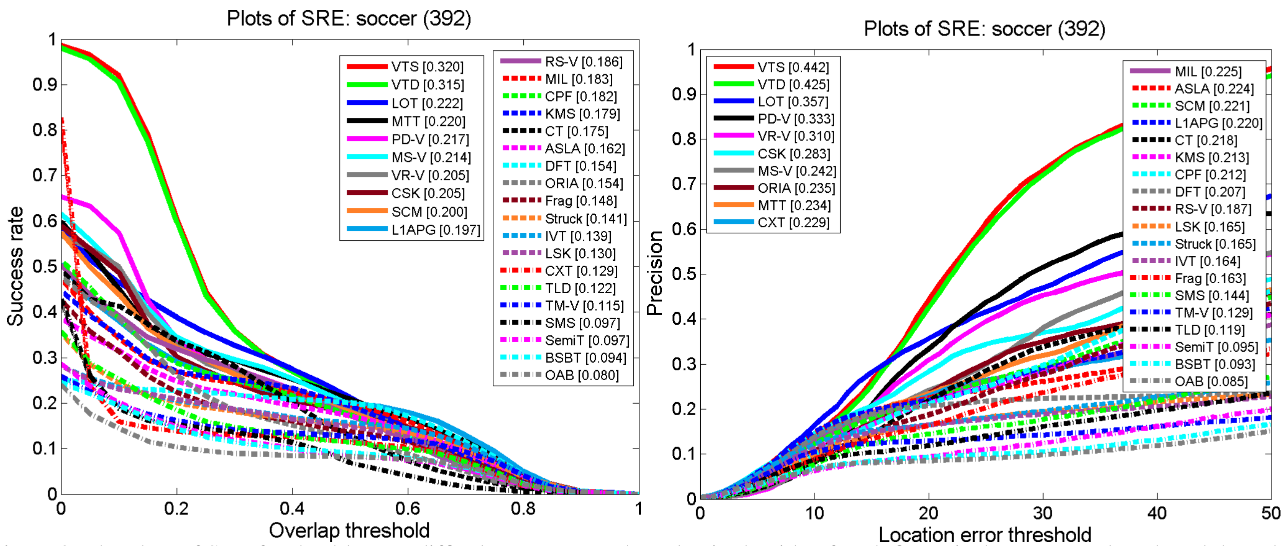

Figure 3. The plots of SRE for the 4th most difficult sequences. The value in the title of each figure is the sequence length and the values in the legend are the performance scores.

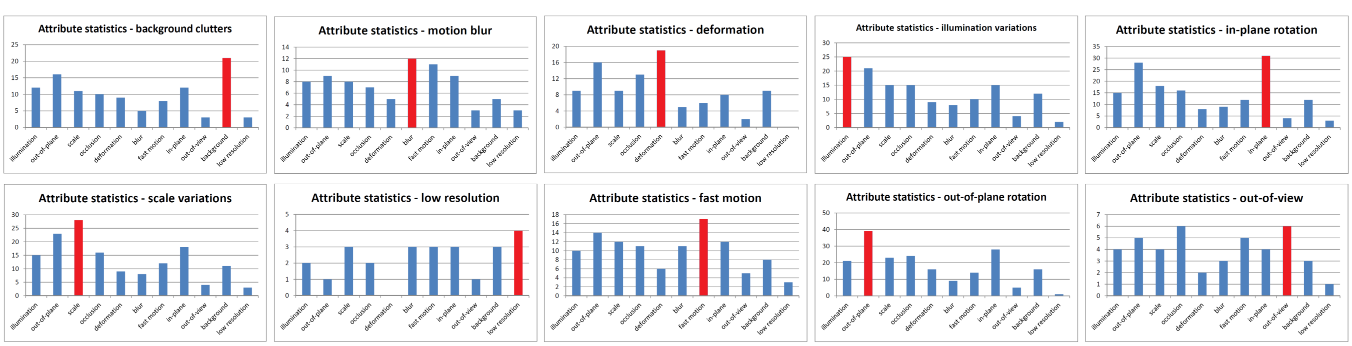

Figure 4. Attribute distribution. In the attribute distribution of each subset, the dominate attribute is marked as red.

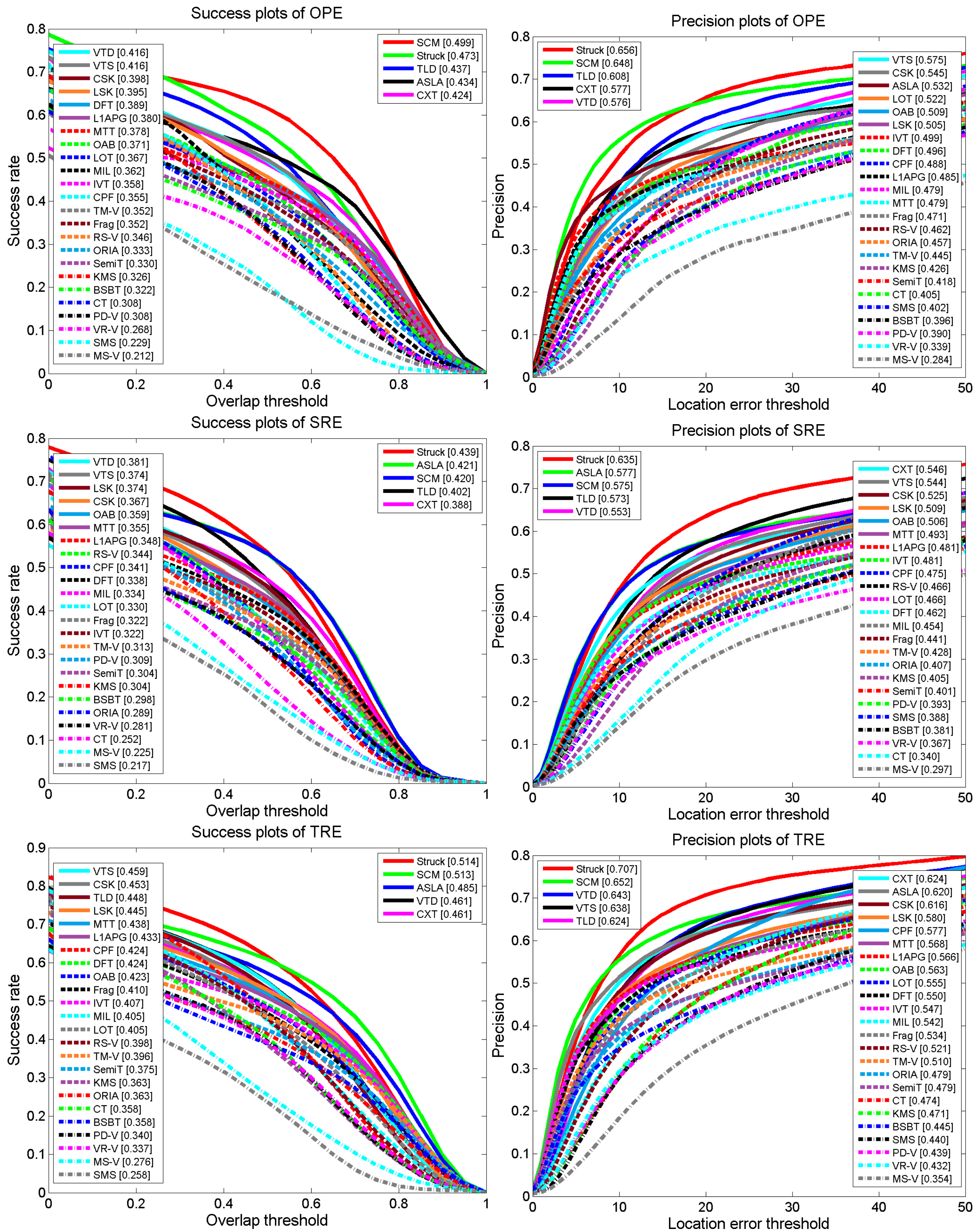

Figure 5. The plots of OPE, SRE, TRE. For each figure, the legend of top 5 ranked trackers is shown separately and the performance score for each tracker is also shown in the legend.

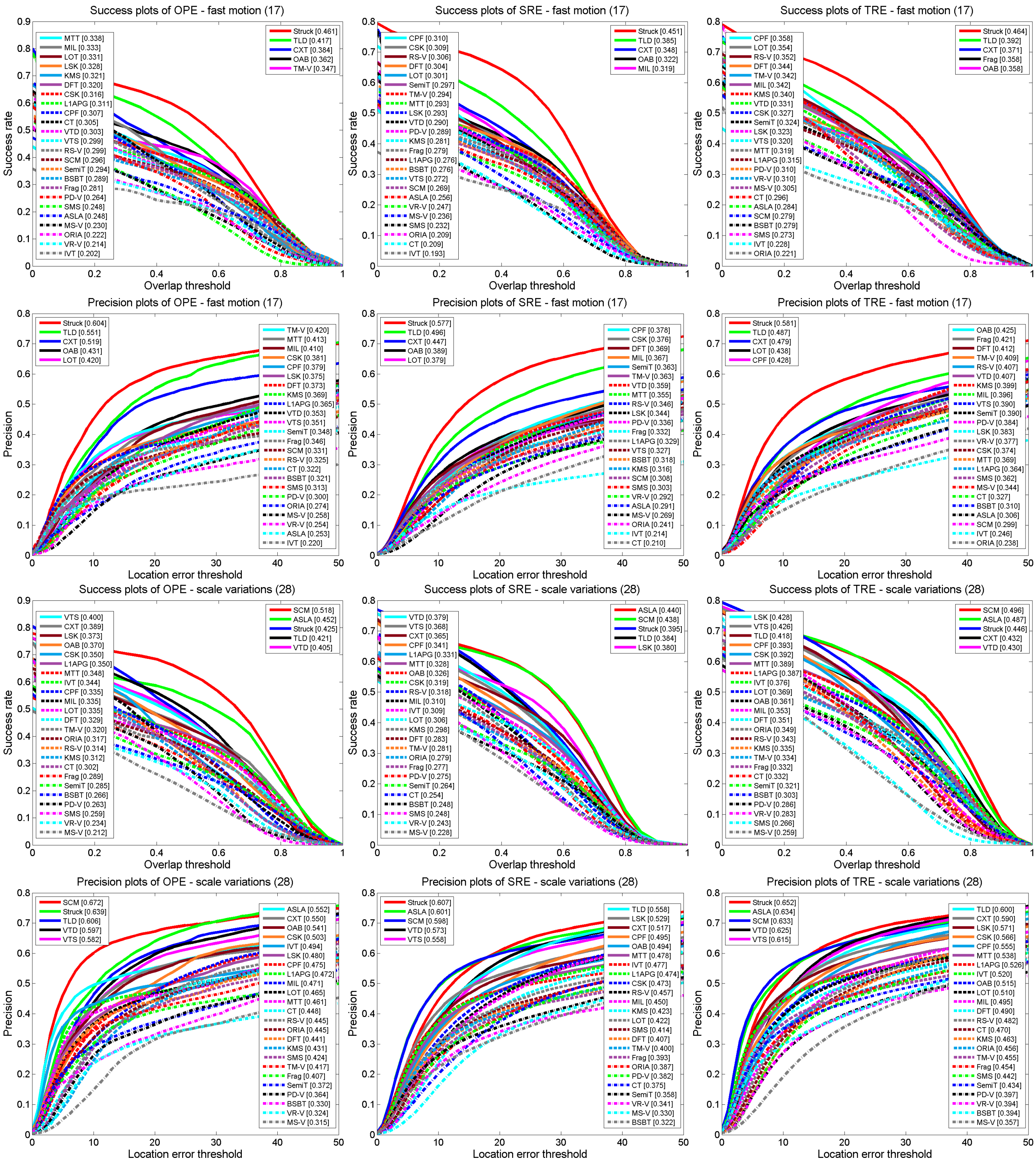

Figure 6. The plots for SV and FM sub-datasets. The value appears in the title is the number of sequences in that sub-dataset.

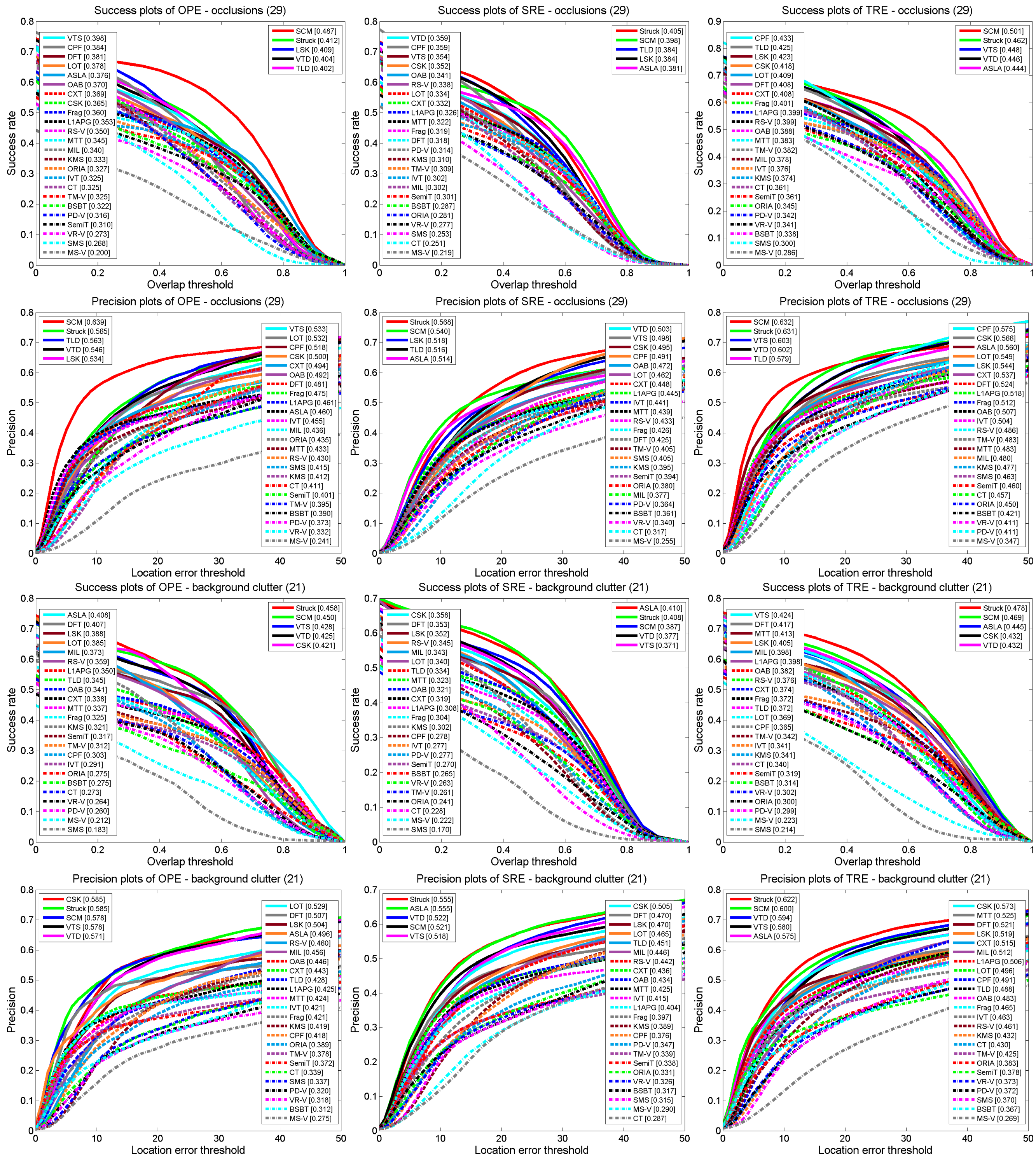

Figure 7. The plots for OCC and BC sub-datasets. The value appears in the title is the number of sequences in that sub-dataset.

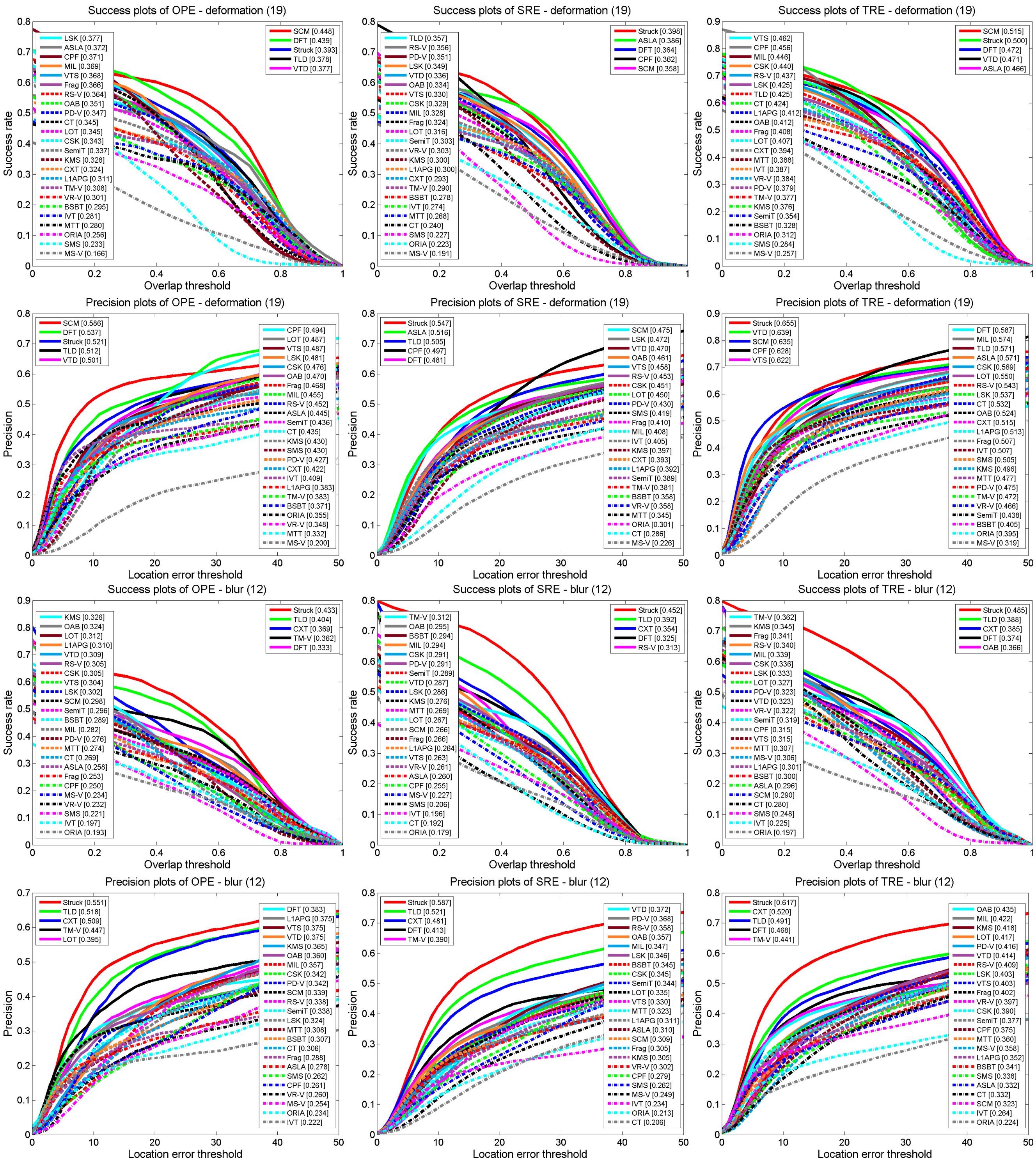

Figure 8. The plots for DEF and MB sub-datasets. The value appears in the title is the number of sequences in that sub-dataset.

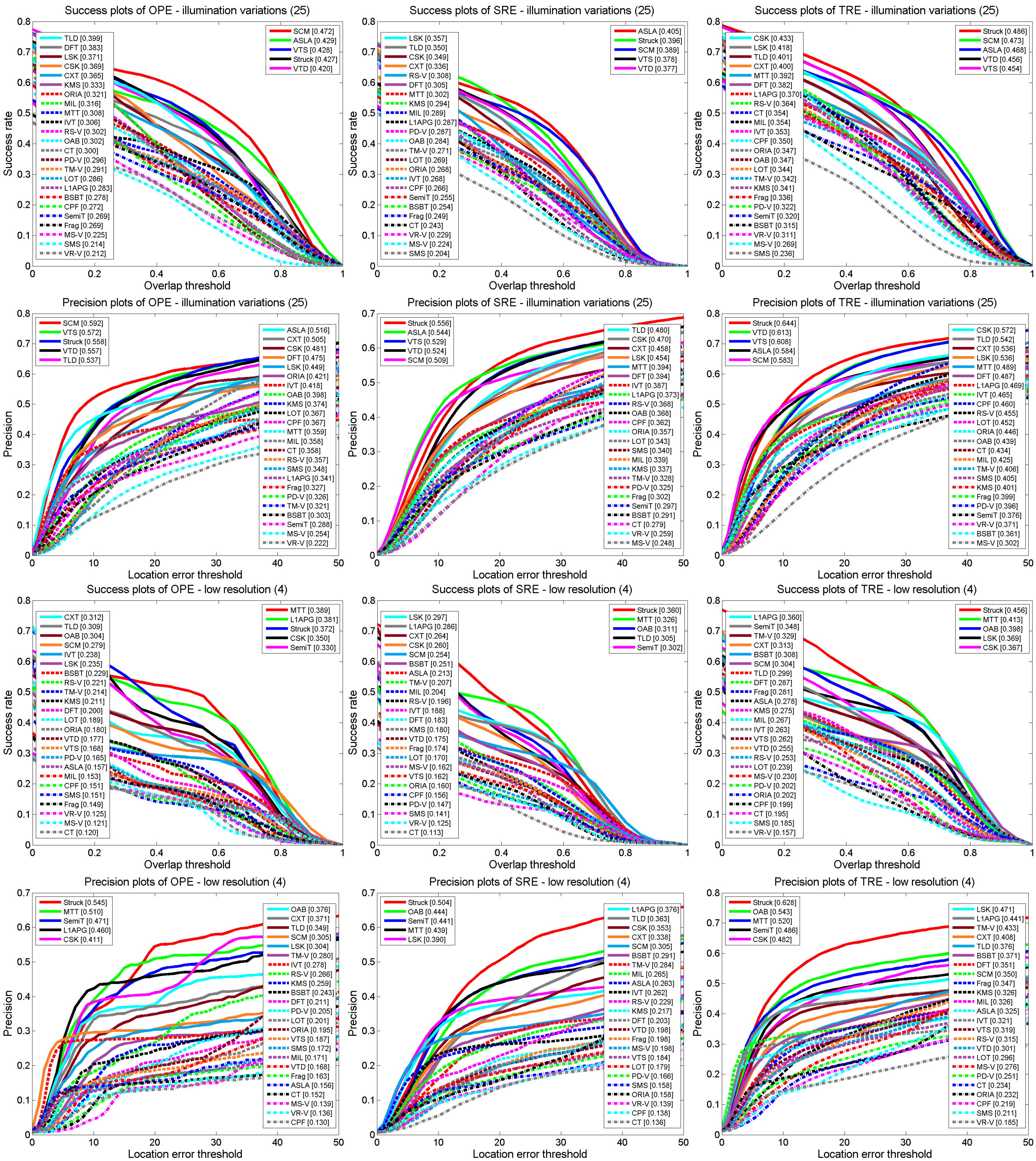

Figure 9. The plots for IV and LR sub-datasets. The value appears in the title is the number of sequences in that sub-dataset.

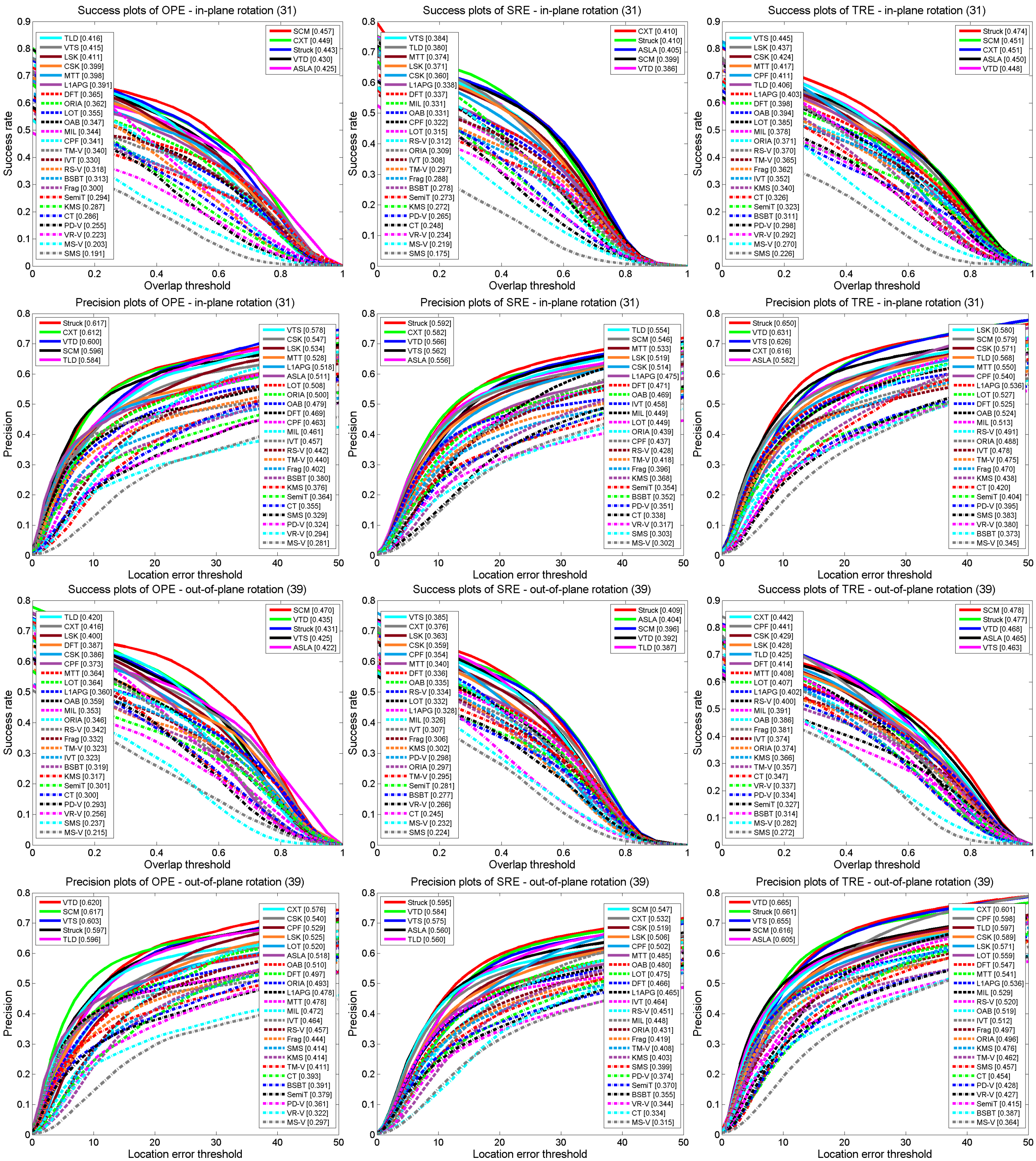

Figure 10. The plots for IPR and OPR sub-datasets. The value appears in the title is the number of sequences in that sub-dataset.

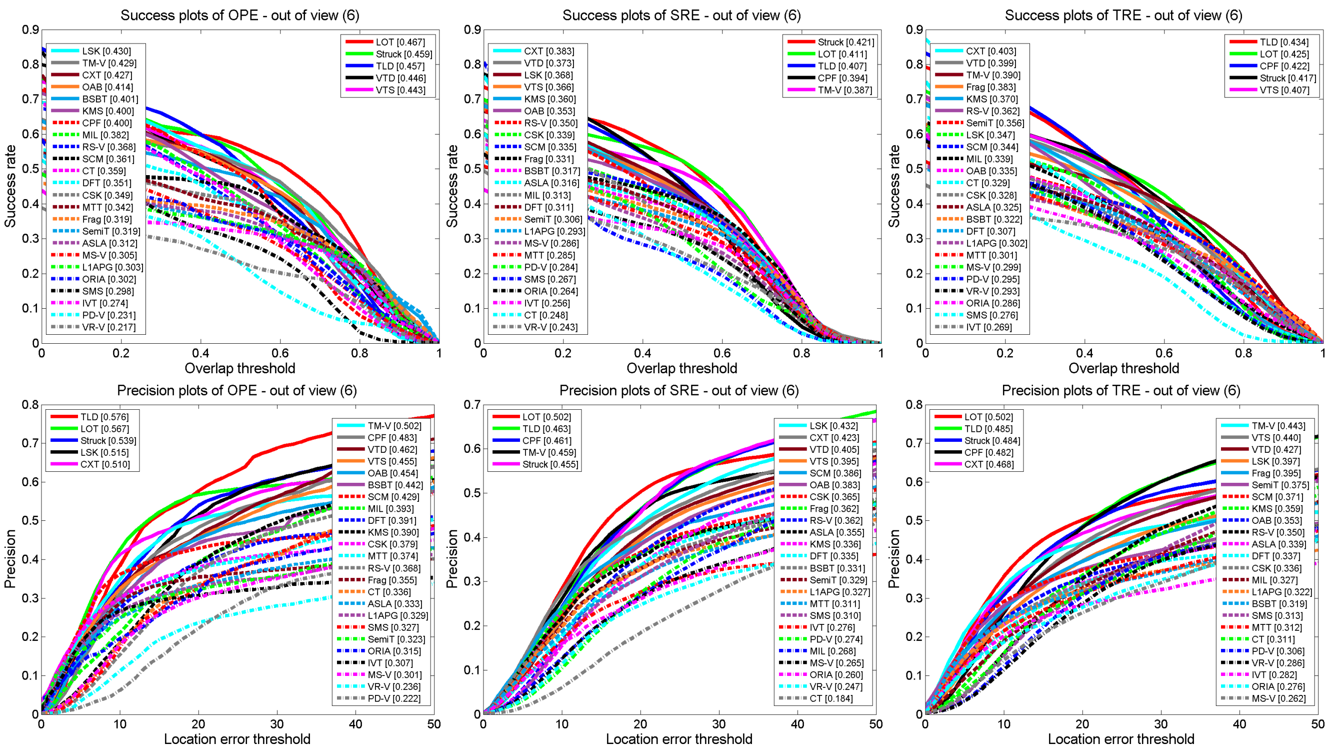

Figure 11. The plots for OV sub-dataset. The value appears in the title is the number of sequences in that sub-dataset.

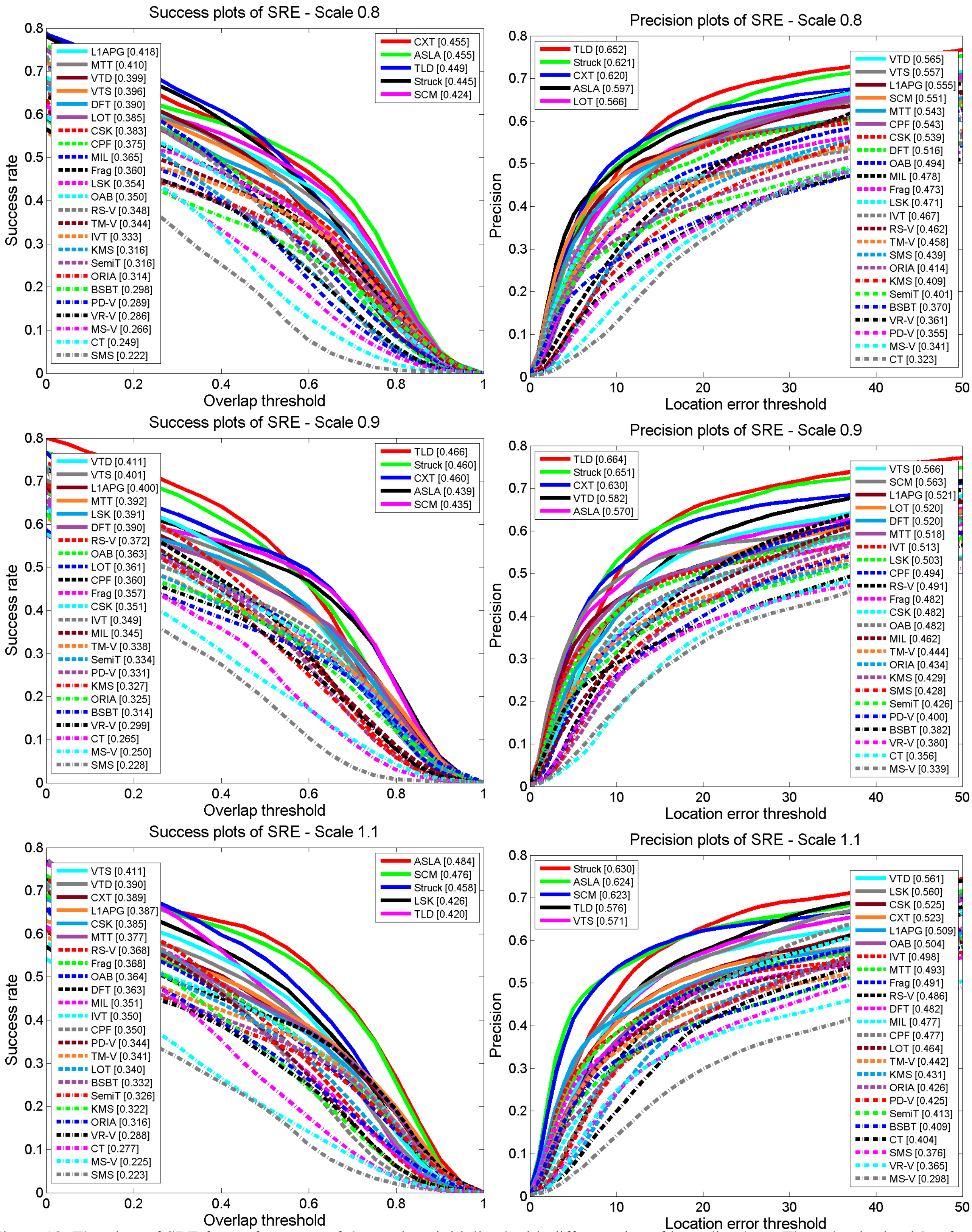

Figure 12. The plots of SRE for performance of the trackers initialized with different size of bounding box. The value in the title of each figure is the scale factor and the values in the legend have the same meanings as Figure 5.

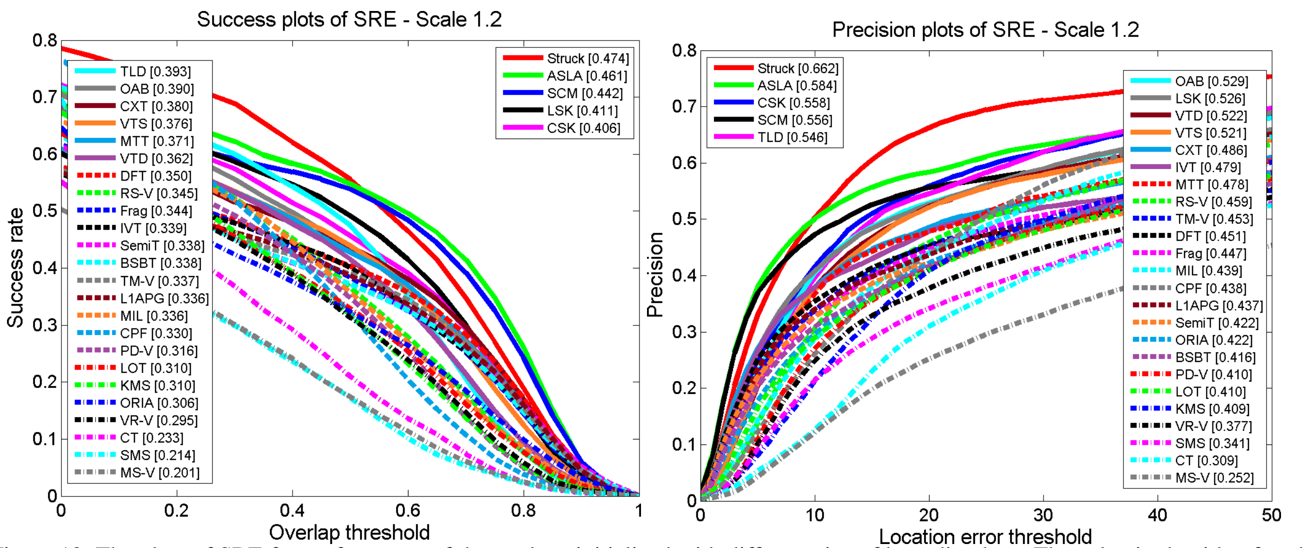

Figure 13. The plots of SRE for performance of the trackers initialized with different size of bounding box. The value in the title of each figure is the scale factor and the values in the legend have the same meanings as Figure 5.

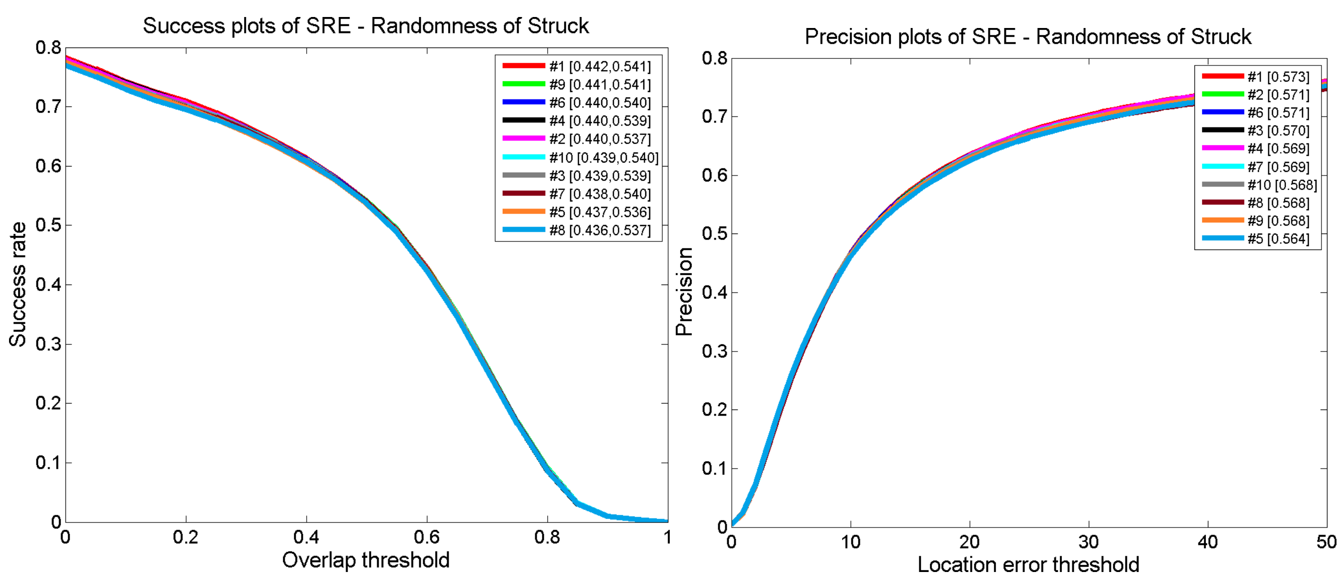

Figure 14. Randomness of Struck. Each plot corresponds to one random SRE test of Struck. In the legend of success plots, the first value in the square bracket is AUC score and the last one is the success rate value in threshold 0.5. In precision plots, the value in the square bracket is the precision value at location threshold of 20 pixels as what is done in [3].