

CS-LoFT: Color and Scale Adaptive Tracking Using Max-Pooling With Bhattacharyya Distance

Noor M. AL-Shakarji*, Filiz Bunyak*, Guna Seetharaman[†] and Kannappan Palaniappan*

* *Department of Computer Science, University of Missouri-Columbia, MO 65211, USA*

[†] *Advanced Computing Concepts, U.S. Naval Research Laboratory, Washington, DC 20375, USA*

Email: * *nmahyd@mail.missouri.edu*, **{punyak, palaniappan}@missouri.edu*, [†]*Gunasekaran.Seetharaman@rl.af.mil*

Abstract—Robust and accurate visual tracking is needed for many computer vision applications from video summarization to visual surveillance. Visual tracking remains to be a challenging task because of factors such as changing object appearance, illumination variations and shadows, partial and full occlusions, camera motion, distractors, and scale changes. Recently our group proposed a Likelihood of Features Tracking (LoFT) system that fuses multiple sources of information about target and its environment to perform robust single object tracking. LoFT has been shown to successfully track objects under different scenarios from full motion video to wide-area motion imagery. In this paper, we extend the LoFT framework with color information and multi-scale feature sets selection to boost its performance. Our extension of color information is incorporated through Color Name (CN) mapping scheme. Multi-scale features computation, and scale selection through frame level max-pooling is also performed to adapt the tracker to scale changes. Experiments on VOT 2015 tracking benchmark, which includes video sequences with significant color and scale variations, show significant performance improvement over baseline LoFT both in accuracy and robustness 11% and 23% respectively.

1. Introduction

Visual object tracking is an integral part of many computer vision applications ranging from video surveillance to biomedical image analysis. Success of these applications depends on robust and accurate tracking under challenging conditions such as changing object appearance and scale, illumination variations, shadows, partial and full occlusions, distractors, and camera motion. Recently our group proposed a recognition based single object tracking system LoFT (Likelihood of Features Tracking) [1], [2], [3] that relies on fusion of multiple sources of information about target and its environment to perform robust single object tracking. Originally LoFT was developed to track objects on wide-area motion imagery (WAMI) acquired by an airborne camera sensor array where a large field-of-view undergoes persistent observation. These sequences involved small support regions for tracked objects, thus did not allow use of detailed appearance description. Because of the large distance between camera platform and the objects, drastic

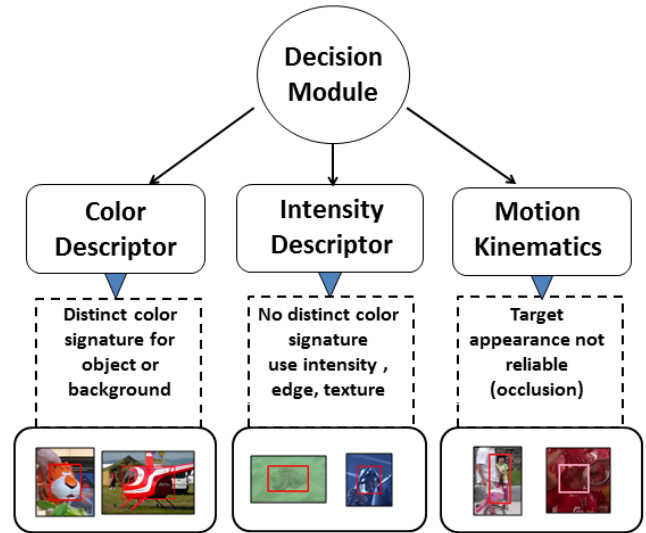


Figure 1. Decision mode operation, with three modes (color, intensity, and kinematics), the tracker switch across these three modes according to the sequence environment.

scale changes did not occur, thus multi-scale processing was not needed. However, full motion video datasets like OTB [4], ALOV [5], and VOT2015 [6], include richer appearance information for tracked objects. These objects also undergo large scale changes due to either object or camera motion, or camera zoom. In this paper we extend the original LoFT framework which have shown to obtain good results for WAMI datasets [1] [2] [3] with (i) color attributes for improved appearance description; (ii) automated feature scale selection for scale change adaptation; and (iii) an operation mode decision unit to switch between CS-LoFT's three different modes of operation (color, intensity, and motion kinematics) based on appearance reliability. Figure 1 illustrates the operation mode decision with some examples.

Color information is incorporated to the LoFT framework using our extension of *Color Name (CN)* scheme introduced in [7]. Max-pooling scale selection is done on likelihood maps obtained for local shape index and normalized curvature index features. The operation mode decision unit activates color use when there is adequate color dis-

similarity between the tracked object and its surrounding region. Incorporation of color and scale selection improves the tracker performance particularly for sequences where the tracked objects have distinct color signatures and undergo rapid scale changes.

The rest of this paper is organized as follows. Section 2 describes the original LoFT framework, the features it uses, and the feature fusion process. Section 3 and 4 describe C-LoFT and CS-LoFT, proposed extensions of LoFT with the new color and scale selection features. Experimental results for VOT2015 video sequences are described in Section 5 followed by conclusions in Section 6.

2. Likelihood of Features Tracking (LoFT) Framework

Likelihood of Features Tracking (LoFT) [1], [2], [3] consists of three main modules: (i) target modeling, (ii) likelihood fusion; and (iii) appearance update. *Target modeling* models the appearance of the tracked object using a rich feature set including intensity, edge, shape and texture information. Intensity, gradient magnitude, gradient orientation, shape and curvature indices (derived from eigenvalues of the Hessian matrix) are some of the features used. Figure 2 describes the features that LoFT use. LoFT uses a recognition-based target localization approach. For each feature in the feature set, a likelihood map is estimated by comparing target feature histogram to sliding window feature histograms within the search region. Efficient sliding window histogram computation is performed using integral histogram-based implementation [8].

Likelihood fusion combines likelihood maps from different features into a single likelihood map. Maximum likelihood location in this map is used for target localization. Likelihood fusion enables adaptation of the tracker to dynamic environment changes and target appearance variabilities. Each feature performs differently depending on the target characteristic and environmental situations during tracking. Two weighting schemes are considered, Variance Ratio (VR) [9] for histogram-based features, and Distractor Index [3] for correlation-based features. *Appearance update* enables dynamic appearance adaptation by maintaining and updating a single template by calculating affine changes in the target to handle orientation and scale changes [2]. Multi-scale Laplacian of Gaussian edge detection followed by segmentation is used to correct for drift.

3. C-LoFT: Extension of LoFT with Color Attributes

Most modern trackers ignore color information and rely purely on features derived from grayscale information [10] [11] [12]. Grayscale images are sometimes sufficient to produce reasonably good tracking results for lower computational cost. However rich chromatic information can be very valuable for object tracking. Various methods have been proposed to incorporate color information to trackers

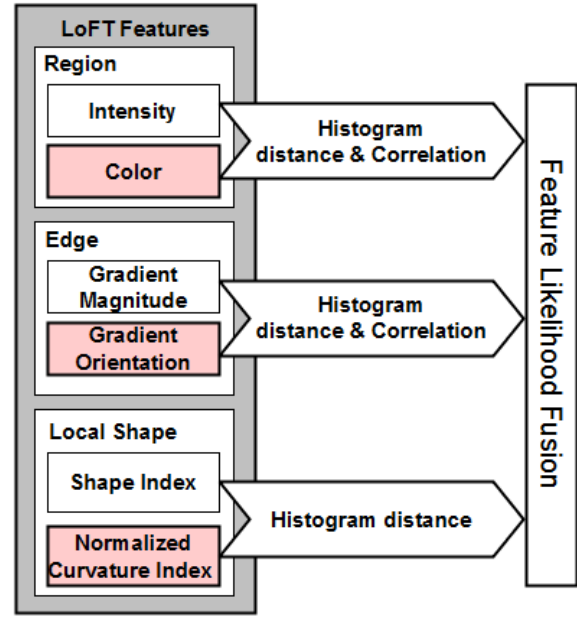


Figure 2. LoFT features, red boxes belong to the extended LoFT that have been added in this paper, C-Loft (color feature) and CS-LoFT (frame level max-pooling best scale selection)

such as simple color space transformations in [13], [14] or color histograms in [15]. More recently, well performing trackers have been further improved with incorporation of color information. In [16] Centinet et al. combined color histograms with Minimum Output Sum of Squared Error (MOSSE) correlation filter [17] as a robust descriptor for object tracking. Danelljan et al. [18] explored the contribution of color in tracking-by-detection frameworks and extended CSK tracker [19] with various color incorporation approaches. Benlefi et al. [20] used color attributes along with motion masks to boost tracker performance. While color introduces rich information for object description, color measurements can vary significantly over an image sequence due to variations in illumination, shadows, and specular reflections. Invariance to these factors has been studied in image classification [19] [14], action recognition [19], and tracking [18].

3.1. Adaptive Color Use with Color Names (CN) Feature

In this paper we extend the original LoFT framework with color information using our extension of *Color Names (CN)* feature [7] (*CN-Ex*). The goal is to boost the performance of the tracker particularly for video sequences where foreground and background have distinct color signatures. Linguistic study described in [21] shows that English language contains eleven basic color terms: *black, blue, brown, grey, green, orange, pink, purple, red, white and yellow*. *Color Names (CN)* feature associates RGB color model with linguistic color labels. We use the mapping provided by [7]

to map RGB color models to 10 color names. The process reduces the 256^3 RGB color space to 10^1 color names space. While *CN* includes mapping for black, gray, and white. We add three additional bins for low saturation regions (low saturation & low intensity, low saturation & medium intensity, low saturation & high intensity), to prevent their switch from one color name to another with slight illumination changes, shadow etc.

Color information is used in the tracking process only when tracked object and its surroundings have distinct color features. C-LoFT uses an adaptive tracking scheme and uses color, intensity, or motion kinematics depending on availability and reliability of these features. We use Bhattacharyya distance [22] to measure color similarity between target and the surrounding area. Let H_{obj} and H_{bg} be 13 – bin color names histograms for tracked object and surrounding background respectively. Discrete probability densities $p(i)$ for the object, and $q(i)$ for the background, are computed by normalizing each histogram by corresponding number of elements in it:

$$p(i) = \frac{H_{obj}(i)}{n_{obj}} \quad (1)$$

$$q(i) = \frac{H_{bg}(i)}{n_{bg}} \quad (2)$$

Bhattacharyya distance between object versus background color distributions is computed as:

$$S = \sum_{i=1}^{13} \sqrt{p(i)q(i)} \quad (3)$$

Inclusion (or exclusion) of color feature is determined for each sequence based on the value of S . Figure 3 shows the process of calculating S for two difference sequence. Low values of S correspond to discriminant color information (dissimilar object versus background color distribution). Figure 4 illustrates the incorporation of color information in C-LoFT.

4. Scale Space Tracking with CS-LoFT

When tracked objects move along the camera axis, their sizes in the video change. Many trackers such as CSK [19], or MOSSE [17] track the objects of interest at a single scale and ignore possible changes of size. This implies poor performance in sequences with significant scale variations. Incorporation of multi-scale templates as in DSST (Discriminative Scale-Space Tracking) [23] improves tracking performance through scale changes.

CS-LoFT extends LoFT with multi-scale processing. Scale changes not only alter size of the tracked objects but also affect scale of their texture. LoFT features that are most sensitive to scale changes are local shape features *local shape index* (Eq.6) and *normalized curvature index* (Eq.7) [3] derived from eigenvalues of Hessian matrix.

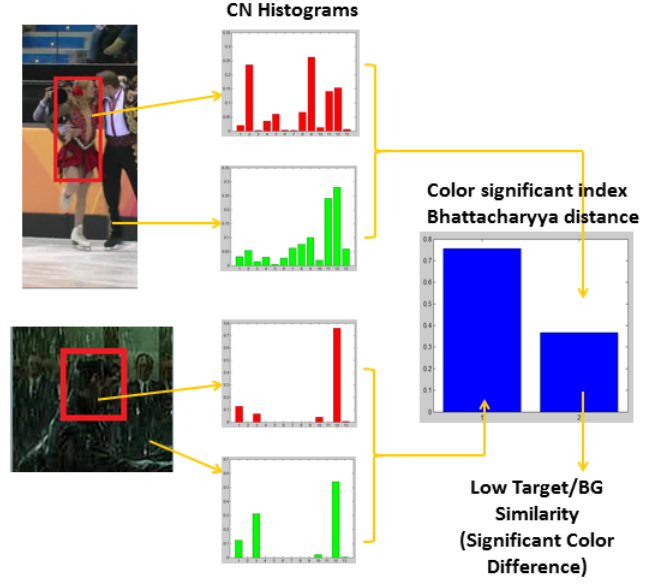


Figure 3. Two examples of different sequences show the difference values of S depends on the similarity between target and surrounded region.

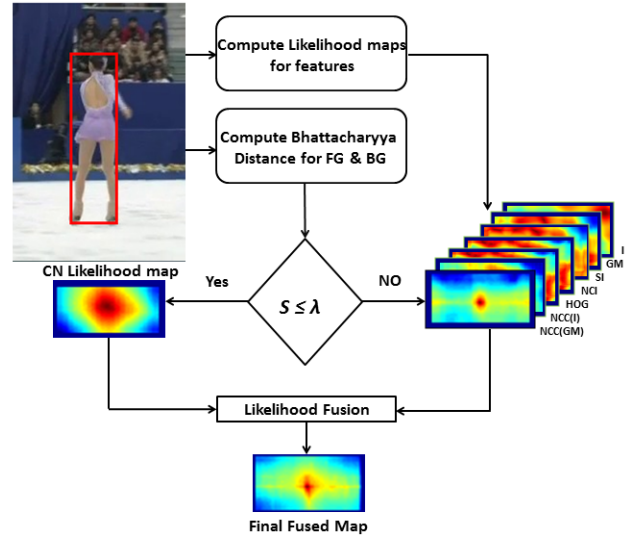


Figure 4. Incorporation of color information in C-LoFT. In which the likelihood map of I represents: Intensity histogram, GM : Gradient magnitude histogram, SI : Shape index histogram, NCI : Normalize curvature index histogram, HOG : Histogram of gradient, $NCC(I)$: Normalized cross correlation for intensity, and $NCC(GM)$: Normalized cross correlation for Gradient

Hessian matrix H describes the second order structure of local intensity variations around each image point $L(x, y)$,

$$H(x, y; \sigma) = \begin{bmatrix} L_{xx}(x, y; \sigma) & L_{xy}(x, y; \sigma) \\ L_{xy}(x, y; \sigma) & L_{yy}(x, y; \sigma) \end{bmatrix} \quad (4)$$

where $L_{xy} = \frac{\partial^2 L}{\partial x \partial y}$, $L(x, y; \sigma)$ refers to the Gaussian

smoothed intensity image

$$L(x, y; \sigma) = g(x, y; \sigma) * I(x, y) \quad (5)$$

g is the Gaussian kernel, and σ is the scale. Local shape index (Eq.6) and normalized curvature index (Eq.7) are derived from the eigenvalues $\lambda_1 \geq \lambda_2$ of Hessian matrix H .

$$\phi(x, y) = \tan^{-1} \frac{\lambda_2(x, y)}{\lambda_1(x, y)} \quad (6)$$

$$\theta(x, y) = \tan^{-1} \frac{((\lambda_1(x, y)^2) + (\lambda_2(x, y)^2))^{\frac{1}{2}}}{1 + L(x, y)} \quad (7)$$

4.1. Frame Level Max-Pooling Scale Selection

We propose a novel frame level scale selection as described in Algorithm 1. First local shape (Eq. 6) and normalized curvature (Eq. 7) indices are computed at multiple scales. Sliding window template matching is performed between search window and tracked object template; and likelihood maps $\mathcal{L}_{\sigma_i}$ are obtained for each scale σ_i . Then, pixelwise maximum likelihood is computed as:

$$\mathcal{L}_{max}(x, y) = \max_{\sigma_i}(\mathcal{L}_{\sigma_i}(x, y)) \quad (8)$$

Finally, best scale σ^* is then selected as the scale that minimizes the mean square error between the pixelwise maximum likelihood $\mathcal{L}_{max}$ and each $\mathcal{L}_{\sigma_i}$.

$$\sigma^* = \underset{i}{\operatorname{argmin}}(\sum_{x, y} (\mathcal{L}_{max} - \mathcal{L}_{\sigma_i})) \quad (9)$$

The described frame level max-pooling scale selection approach is applied to shape index and normalized curvature index separately. Figure 5 describes frame level max-pooling scale selection approach. Likelihood maps corresponding to selected scales are fused with likelihood maps for the remaining features. Proposed frame level scale selection ensures (i) scale coherence in the selected likelihood map (all pixels correspond to the same scale); and (ii) robustness against local outliers.

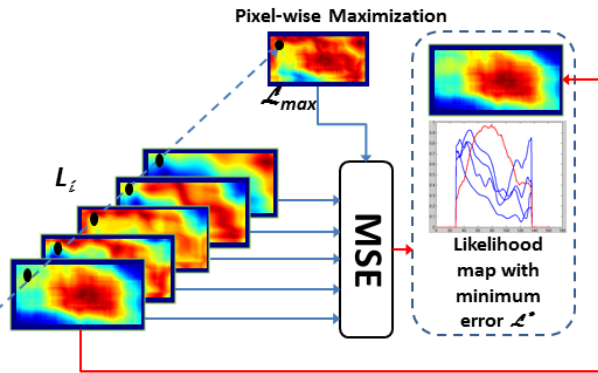


Figure 5. Frame level max-pooling scale selection approach.

Algorithm 1 Frame level max-pooling scale selection

Input : I_{SW}, I_T : search window and object template

Output : $\sigma^*, \mathcal{L}^*$: best scale and corresponding likelihood map

```

1: for each scale  $i$  do
2:    $L(x, y; \sigma_i) \leftarrow g(x, y; \sigma_i) * I(x, y)$ 
3:   Compute Hessian matrix  $H(x, y; \sigma)$  Eq. 4 and its
     eigenvalues  $\lambda_1, \lambda_2$ 
4:   Compute local shape features  $\phi(x, y)$  Eq. 6,  $\theta(x, y)$ 
     Eq. 7
5:    $\mathcal{L}_{\sigma_i} \leftarrow$  sliding window template matching between
     search window & object template
6: end for
7: Pixelwise maximum likelihood:
    $\mathcal{L}_{max}(x, y) \leftarrow \max_{\sigma_i}(\mathcal{L}_{\sigma_i}(x, y))$ 
8: Best frame level scale:
    $\sigma^* \leftarrow \underset{i}{\operatorname{argmin}}(\sum_{x, y} (\mathcal{L}_{max} - \mathcal{L}_{\sigma_i}))$ 

```

5. Experimental Results

We have evaluated the proposed CS-LoFT tracker on VOT2015 benchmark dataset [6], comparing against the original LoFT tracker. VOT challenge performance measures *accuracy* and *robustness* were used for evaluation. *Accuracy* measures how well the bounding box predicted by the tracker overlaps with the ground truth bounding box. *Robustness* measures how many times the tracker loses the target during tracking. A failure is indicated when the overlap measure becomes zero [6]. Figures 6 and 7 shows intermediate results related to color and scale processing in CS-LoFT. CS-LoFT switches to color tracking mode only when the tracked objects and their surroundings have different color features (Figure 6a), relies on intensity otherwise (Figure 6b). Figure 7 shows how much the proposed frame level max-pooling scale selection method preserves spatial context while pixel-wise max pooling over scales generates distractors. Table 1 and Figure 8 show detailed evaluation of LoFT and CS-LoFT on 60 video sequences of VOT2015 dataset. Table 1 groups the sequences into three sections according to their *robustness* measurement on CS-LoFT as improved, same, and worst. Figure 9 compares the performance of CN, our extended CN, LoFT, and the proposed CS-LoFT trackers in terms of robustness.

CS-LoFT improves the performance of the original LoFT tracker in terms of both *accuracy* and *robustness* by 11% and 23% respectively. Performance is particularly improved for the sequences that have significant color distractor and rapid scale changes.

6. Conclusions

We presented a recognition based object tracking framework that extends Likelihood of Features Tracker (LoFT) with color attributes, scale selection, and operation mode selection. Addition of color attributes improves object and

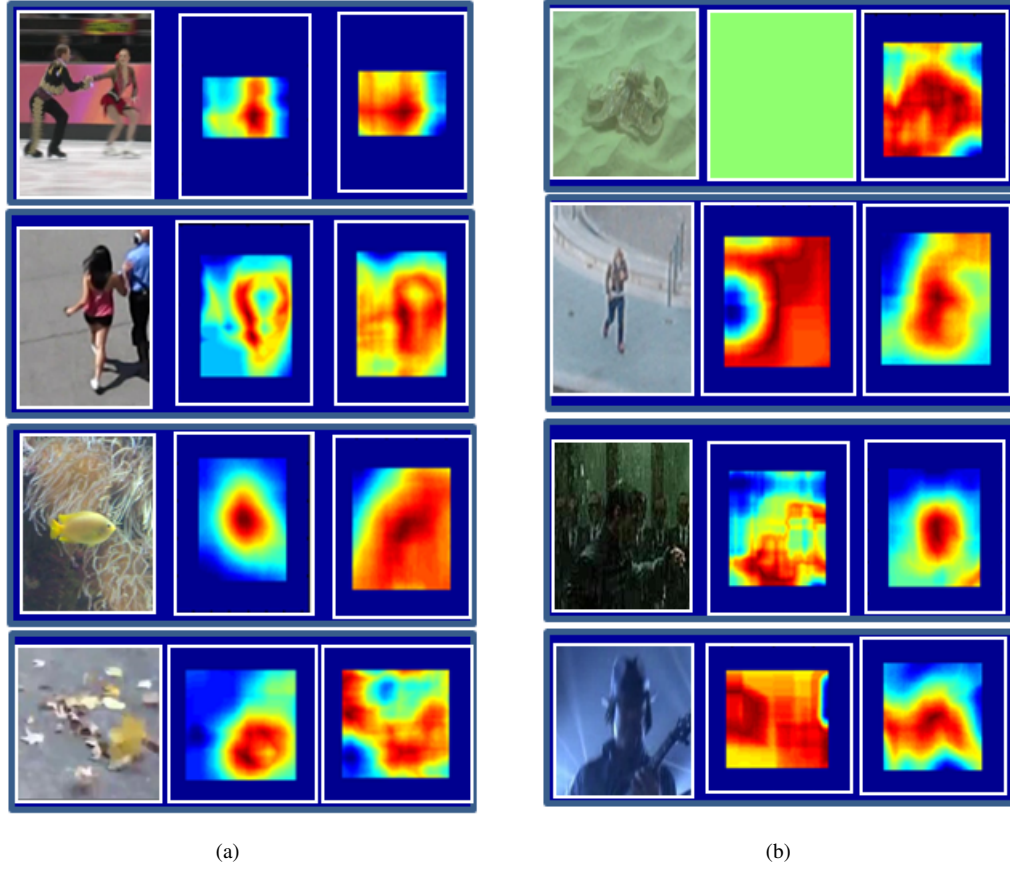


Figure 6. Color versus intensity. (a) Tracked objects and their surroundings have different color features; (b) Tracked objects and their surroundings have similar color but different intensity features. Columns left to right: original image, likelihood map obtained using color names (CN) feature, likelihood map obtained using intensity feature.

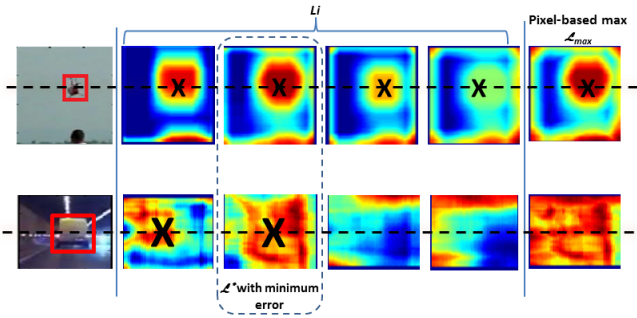


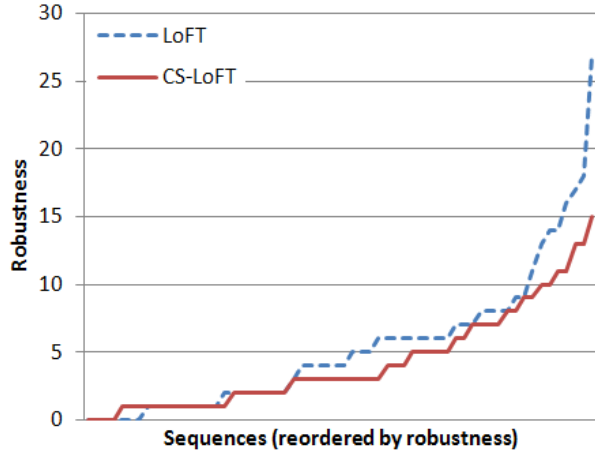
Figure 7. Frame level max-pooling best scale selection versus pixel-based max-pooling. The third column: the frame level L^* over all scales, while the sixth column: L_{max} pixel-based max-pooling result.

background appearance description. Color information is incorporated through CN-Ex, our extension of Color Name (CN) scheme. CN-Ex decouples encoding of low saturation and high saturation colors for more robust description and matching through illumination changes and shadows. Automated feature scale selection adapts the tracker to scale changes caused by object motion, camera motion, or

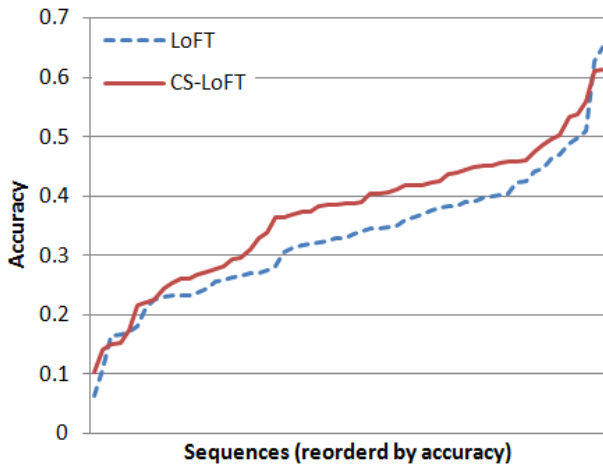
zoom. The proposed scale selection method ensures scale coherence in the likelihood map and robustness against local outliers. Operation mode decision unit selects CS-LoFTs three different modes of operation (color, intensity, and motion kinematics) based on appearance reliability. Experimental results show improved performance particularly for the sequences that have significant color distractors and rapid scene changes. As future work, we are planning to adapt our holistic object description framework to part-based description and tracking.

References

- [1] R. Pelapur, K. Palaniappan, and G. Seetharaman, "Robust orientation and appearance adaptation for wide-area large format video object tracking," *IEEE Int. Conf. on Advanced Video and Signal-Based Surveillance*, pp. 337–342, 2012.
- [2] R. Pelapur, S. Candemir, F. Bunyak, M. Poostchi, G. Seetharaman, and K. Palaniappan, "Persistent target tracking using likelihood fusion in wide-area and full motion video sequences," *Int. Conf. on Information Fusion*, pp. 2420–2427, 2012.
- [3] K. Palaniappan, F. Bunyak, P. Kumar, I. Ersoy, S. Jaeger, K. Ganguli, A. Haridas, J. Fraser, R. M. Rao, and G. Seetharaman, "Efficient feature extraction and likelihood fusion for vehicle tracking in low



(a)



(b)

Figure 8. Performance evaluation on VOT2015 dataset: (a) robustness, (b) accuracy. Sequences reordered by robustness and accuracy values respectively.

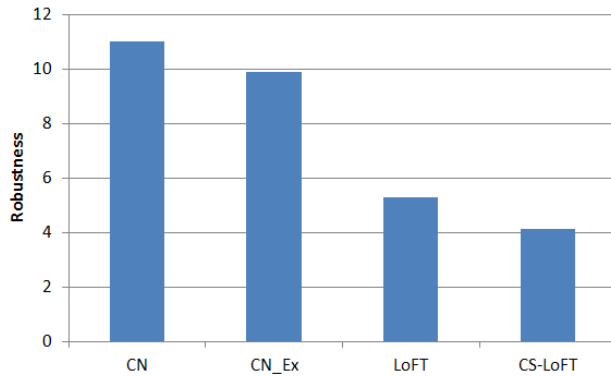


Figure 9. Robustness comparison between CN, extended CN, LoFT, and CS-LoFT trackers.

TABLE 1. LOFT AND CS-LOFT ROBUSTNESS AND ACCURACY COMPARISON ORDERED FORM MOST SEQUENCES THAT IMPROVED TO LESS IMPROVED IN ROBUSTNESS MEASUREMENT.

No.	Sequences	LoFT	CS-LoFT	LoFT	CS-LoFT
		Robustness		Accuracy	
1	iceskater1	27	10	0.35	0.40
2	book	14	4	0.23	0.26
3	soccer1	17	7	0.45	0.49
4	iceskater2	14	5	0.35	0.41
5	bolt1	18	13	0.23	0.24
6	bolt2	6	3	0.23	0.25
7	hand	13	10	0.33	0.39
8	handball1	6	3	0.34	0.39
9	handball2	8	5	0.34	0.39
10	pedestrian1	6	3	0.38	0.44
11	ball1	6	4	0.11	0.14
12	dinosaur	4	2	0.26	0.28
13	fernando	4	2	0.26	0.29
14	fish4	5	3	0.27	0.34
15	gymnastics1	16	14	0.32	0.37
16	leaves	4	2	0.35	0.41
17	shaking	6	4	0.40	0.46
18	singer2	4	2	0.42	0.46
19	bag	1	0	0.06	0.10
20	fish2	6	5	0.27	0.31
21	gymnastics2	7	6	0.32	0.38
22	gymnastics4	6	5	0.33	0.38
23	motocross1	6	5	0.37	0.42
24	motocross2	4	3	0.37	0.42
25	rabbit	8	7	0.39	0.45
26	soldier	2	1	0.47	0.50
27	sphere	2	1	0.49	0.53
28	tunnel	2	1	0.63	0.61
29	ball2	2	2	0.16	0.15
30	birds2	1	1	0.18	0.22
31	blanket	2	2	0.21	0.22
32	crossing	0	0	0.26	0.28
33	fish1	4	4	0.26	0.30
34	godfather	1	1	0.31	0.37
35	graduate	11	11	0.32	0.37
36	gymnastics3	5	5	0.32	0.38
37	helicopter	1	1	0.34	0.40
38	nature	7	7	0.38	0.42
39	racing	0	0	0.40	0.45
40	road	7	7	0.40	0.45
41	sheep	2	2	0.40	0.46
42	singer3	1	1	0.44	0.47
43	soccer2	5	5	0.46	0.50
44	tiger	8	8	0.50	0.54
45	traffic	1	1	0.51	0.56
46	wiper	1	1	0.65	0.61
47	birds1	9	10	0.17	0.17
48	bmh	0	1	0.22	0.22
49	car1	0	1	0.24	0.27
50	car2	0	1	0.24	0.27
51	girl	8	9	0.28	0.36
52	glove	2	3	0.31	0.36
53	marching	2	3	0.36	0.42
54	octopus	0	1	0.38	0.44
55	pedestrian2	1	2	0.39	0.44
56	singer1	0	1	0.42	0.46
57	butterfly	3	5	0.23	0.26
58	fish3	1	3	0.27	0.33
59	matrix	9	11	0.36	0.42
60	basketball	6	9	0.17	0.15
Avg.		5.37	4.15	0.33	0.37

frame rate airborne video,” *Int. Conf. on Information Fusion*, pp. 1–8, 2010.

[4] Y. Wu, J. Lim, and M.-H. Yang, “Object tracking benchmark,” *IEEE*

Trans. on Pattern Analysis and Machine Intelligence, vol. 37, no. 9, pp. 1834–1848, 2015.

- [5] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: An experimental survey," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 37, no. 7, 2014.
- [6] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, L. Cehovin, G. Fernandez, T. Vojir, G. Hager, G. Nebehay, and R. Pflugfelder, "The visual object tracking vot2015 challenge results," *IEEE Int. Conf. on Computer Vision Workshops*, pp. 1–23, 2015.
- [7] J. Van De Weijer, C. Schmid, J. Verbeek, and D. Larlus, "Learning color names for real-world applications," *IEEE Trans. on Image Processing*, vol. 18, no. 7, pp. 1512–1523, 2009.
- [8] F. Porikli, "Integral histogram: a fast way to extract histograms in cartesian spaces," *IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 1, pp. 829–836, 2005.
- [9] R. T. Collins, Y. Liu, and M. Leordeanu, "Online selection of discriminative tracking features," *IEEE trans. on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1631–1643, 2005.
- [10] S. Hare, A. Saffari, and P. H. Torr, "Struck: Structured output tracking with kernels," *IEEE Int. Conf. on Computer Vision*, pp. 263–270, 2011.
- [11] K. Zhang, L. Zhang, and M.-H. Yang, "Real-time compressive tracking," *European Conf. on Computer Vision*, pp. 864–877, 2012.
- [12] T. B. Dinh, N. Vo, and G. Medioni, "Context tracker: Exploring supporters and distracters in unconstrained environments," *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1177–1184, 2011.
- [13] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," *European Conf. on Computer Vision*, pp. 661–675, 2002.
- [14] K. Nummiaro, E. Koller-Meier, and L. Van Gool, "An adaptive color-based particle filter," *Image and Vision Computing*, vol. 21, no. 1, pp. 99–110, 2003.
- [15] Z. Zivkovic and B. Krose, "An em-like algorithm for color-histogram-based object tracking," *IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 1, pp. 1–798, 2004.
- [16] M. Centir, P. Fragneto, D. Denaro, B. Rossi, and C. Marchisio, "A combined color-correlation visual model for object tracking using particle filters," *Int. Symp. on Image and Signal Processing and Analysis*, pp. 153–158, 2013.
- [17] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2544–2550, 2010.
- [18] M. Danelljan, F. Shahbaz Khan, M. Felsberg, and J. Van de Weijer, "Adaptive color attributes for real-time visual tracking," *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1090–1097, 2014.
- [19] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," *European conf. on Computer Vision*, pp. 702–715, 2012.
- [20] T. Benlefi, R. Liu, B. Dai, and M. B. Boubekeur, "Combining color and vibe foreground mask for better tracking in video sequences," *Int. Cong. on Image and Signal Processing*, pp. 18–22, 2015.
- [21] B. Berlin and P. Kay, "Basic color terms: their universality and evolution," *University of California Press*, 1969.
- [22] A. Bhattacharyya, "On a measure of divergence between two multinomial populations," *Sankhy: The Indian Journal of Statistics (1933-1960)*, vol. 7, no. 4, pp. 401–406, 1946.
- [23] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," *British Machine Vision Conf.*, 2014.