Robust Orientation and Appearance Adaptation for Wide-Area Large Format Video Object Tracking

Rengarajan Pelapur[a], Kannappan Palaniappan[a]
Gunasekaran Seetharaman[b]
[a] Department of Computer Science, University of Missouri-Columbia
[b] Air Force Research Lab(United States)
rengarajanpelapur@mail.missouri.edu, palaniappank@missouri.edu

Abstract

Visual feature-based tracking systems need to adapt to variations in the appearance of an object and in the scene for robust performance. Though these variations may be small for short time steps, they can accumulate over time and deteriorate the quality of the matching process across longer intervals. Tracking in aerial imagery can be challenging as viewing geometry, calibration inaccuracies, complex flight paths and background changes combined with illumination changes, and occlusions can result in rapid appearance change of objects. Balancing appearance adaptation with stability to avoid tracking non-target objects can lead to longer tracks which is an indicator of tracker robustness. The approach described in this paper can handle affine changes such as rotation by explicit orientation estimation, scale changes by using a multiscale Hessian edge detector and drift correction by using segmentation. We propose an appearance update approach that handles the 'drifting' problem using this adaptive scheme within a tracking environment that is comprised of a rich feature set and a motion model.

1. Introduction

The wide-area tracking system, LoFT [1, 2], is an appearance-based single object tracker that uses a set of image features comprised of gradient orientation information using histogram of oriented gradients, gradient magnitude, intensity maps, median binary patterns [3] and shape indices based on eigenvalues of the Hessian matrix. LoFT robustly tracks vehicles in wide-area large format (WALF) video that is airborne imagery characterized by large spatial coverage, high resolution of about 25 cm GSD (Ground Sampling Distance) and low frame rate of a few frames per second. Wide-area large format imagery is also known by several other terms including wide-area aerial surveillance (WAAS), wide-area persistent surveillance (WAPS), Large Volume Streaming Data (LVSD) and wide-area motion imagery (WAMI) [1, 4, 5, 6]. LoFT performs feature fusion by comparing a target appearance model within a search region using feature likelihood maps which estimates the likelihood of each pixel with the search window belonging to part of the target. The tracking system follows the paradigm of object tracking using an initial template or object appearance model. An object is tracked through a sequence of images by extracting an exemplar image patch in the first frame and then matching to the best local image patch or region in subsequent frames using feature operators and feature fusion [1, 2]. Appearance modeling or adapting to changes in object appearance has been handled in different ways in the past, either by using target observations [7, 8], fusing multiple sources of sensor information [9, 10], parts-based deformable templates [11, 12], as a learning problem using on-line boosting [13], or online multiple instance boosting [14], while earlier techniques used offline classifier training [15, 16]. Offline learning requires training on a set of image patches which introduces more complexity as the accuracy then depends on factors such as the amount of variation in the image patches, the training method used, size of the training set and avoiding overtraining. Recently, sparse representation for object classification based tracking methods are actively being investigated [17, 18], but L_1 minimization is computationally expensive, especially when very high dimensional feature vectors are used and appearance adaptation is still challenging even using sparse models.

A typical assumption in appearance-based tracking is that the object's visual representation does not change significantly across a sequence of frames. Al-

978-0-7695-4797-8/12 $26.00 © 2012 IEEE
DOI 10.1109/AVSS.2012.92

though this assumption may hold over short intervals for high frame rate full motion video, it is often not valid across longer time intervals, lower frame rates and abrupt changes in the environment or imaging conditions. One (naive) approach to accommodate appearance changes, is to update the target appearance model on every frame based on the results from the matching algorithm. Updating on every frame can lead to instabilities in representing the target due to variations in scene conditions and partial occlusions which leads to the *drifting* problem, while updating less frequently can result in the model missing appearance changes that are important for continuous tracking. Making this tradeoff is referred to as the stability-plasticity dilemma [19, 13]. The way our proposed approach handles this dilemma is by trying to balance between naive updating on every frame and a no update scheme using a single fixed template, by estimating when there is significant change in the appearance prior to an update. Model updates can occur when there is a rapid change in pose of the object or illumination change thus maintaining *plasticity*, on the other hand when the changes in orientation, scale or illumination is very small then an update is not performed and contributes to model *stability*. This article is aimed at studying the adaptation scheme tradeoffs and their impact on producing longer tracks using the challenging CLIF wide-area imagery dataset [20]. Appearance adaptation also can improve performance on event detection and video indexing tasks [21]. The following sections describe the appearance measurement model, appearance update with drift correction, and experimental evaluation using wide-area motion imagery.

2. Feature-based Appearance Model

We consider that the object to be tracked is defined by its initial appearance and associated set of feature descriptors which are provided as input to the tracking system. We denote the given sequence of images as I_k where k is the frame number. The image patch or region, centered at (x, y), representing the target object in the initial frame I_0, is denoted as $T_0(x, y)$. Let us denote a new template T_1 as the region localizing the target in I_1, and the variation between T_1 and T_0 as V_1. Appearance variations can be modeled by changes in shape and texture [22]. For our adaptive model, we assume that the variations in shape and texture can be modeled by an affine transformation matrix A_k, that includes translation, rotation, scale and shear changes between time $k - 1$ and k. When the variation in appearance V_k is significant then an appearance update is performed. The following subsections describe the appearance modeling approach that we use along with the criteria for updating followed by a description of the update strategies.

2.1. Extracting an Approximate Template

We assume that the tracking position localization is satisfactory so that we can extract an image patch with the same parameters as the initialized template model or the previously updated appearance model. The localization using the fused feature conditional likelihood map of the target is given by [2],

$$(x_{\mathrm{k}+1}, y_{\mathrm{k}+1}) = \arg\max_{x,y} \; L(I_{\mathrm{k}+1}(x, y), T_\mathrm{k}) \tag{1}$$

where $L(I_{\mathrm{k}+1}, T_\mathrm{k})$ is the likelihood matching function, $(x_{\mathrm{k}+1}, y_{\mathrm{k}+1})$ the estimated target position at time $k+1$, and T_{k+1} the associated target template. The two target appearance models, T_k and $T_{\mathrm{k}+1}$, with coordinate locations are used in the update module.

2.2. Multiscale Edge Detection

Diminishing the influence of background pixels is important as the accuracy of orientation estimation depends on a reliable edge map. A multiscale edge detection approach is used based on [23] which developed a vessel enhancement filter based on an eigenvalue analysis of the Hessian matrix across scale space. The Hessian matrix uses second-order image derivatives,

$$H(I) = \sigma^2 \begin{bmatrix} I_{xx} \; I_{xy} \\ I_{xy} \; I_{yy} \end{bmatrix} \tag{2}$$

where σ^2 is used to achieve invariance under image rescaling [24, 25]. Let λ_1, λ_2 be the eigenvalues and $\mathbf{e}_1$, $\mathbf{e}_2$ the corresponding eigenvectors of the Hessian matrix with $|\lambda_1| \geq |\lambda_2|$. The Hessian is computed after convolving the iamge with a 2D isotropic Gaussian,

$$G(x, y, \sigma_i) = (1/\sqrt{2\pi}\sigma_i) exp(-(x^2 + y^2)/(2\sigma_i^2)) \tag{3}$$

where σ_i is the standard deviation of the Gaussian distribution for the i^{th} scale. We use a suitable range of scales based on target size and use the Frobenius norm of the Hessian matrix at the maximum scale as a measure of the second order structureness:

$$\begin{aligned} I_{k,H}(x, y) &= ||H||_F \\ &= \sqrt{(\max_\sigma \lambda_1(x, y, \sigma))^2 + (\max_\sigma \lambda_2(x, y, \sigma))^2} \end{aligned} \tag{4}$$

The best scale at each pixel, $\sigma_{max}(x, y)$, is determined by the maximum response of each eigenvalue. A range of smaller scales is selected in order to supress blob-like responses which after thresholding will produce an enhanced edge map while small scale responses in the background are diminished using the second order structureness [25].

2.3. Orientation Estimation

We propose a robust way to estimate the orientation of vehicle objects based on the Radon transform assuming that a reliable edge map is available using the second order structureness image $I_{k,H}$ from Eq. 4. The Radon transform computes a projection of the image as a sum of line integrals accumulating pixel intensities along rays at a set of angles. The Radon transform line integral, $R(\rho, \theta)$, maps the image to a new ρ and θ space, where a ray is given as, $\rho = x \cos\theta + y \sin\theta$. The Radon transform of the *thresholded* structureness feature map, $\tilde{I}_{k,H}$, at time k is defined as,

$$R_k(\rho, \theta) = \iint \tilde{I}_{k,H}(x, y)\delta(\rho - x\cos\theta - y\sin\theta)dx\, dy \tag{5}$$

where $\delta(\cdot)$ is the Dirac delta sampling function. Given the Radon transform projections we can calculate the variance of each profile (ρ varies while θ remains fixed),

$$V_k(\theta_F) \equiv Var(R_k(\rho, \theta = \theta_F)) \tag{6}$$

The second derivative properties of the profile variance function, $V_k(\theta)$, is quite robust as an estimate of local structure orientation [26]. We look for peaks in the profile variance function by searching for a strong minimum in the second derivative of the variance. An example of vehicle orientation estimation using the proposed Radon transform method is shown in Fig. 1 and its subsequent impact on target localization accuracy by reducing the impact of appearance drift is illustrated in Fig. 2.

2.4. Appearance Update Confidence Value

A confidence value is used during the decision process to adaptively determine when an appearance update should take place to switch to the new template model $T_{\mathrm{k+i}}$ and replace T_{k}. The confidence value is adaptively estimated by selecting and associating the highest peaks in the Radon transform function (Eq. 5) across two time steps. Let us characterize the set of strongest peaks (strength and angle θ ignoring ρ) at two time points by lists $\mathbf{p}_{(k+i)}$ and $\mathbf{p}_k$. Algorithm 1 is used to establish peak-to-peak correspondences among the set of strongest peaks in the transform domain.

```
Algorithm 1 Radon Transform Peak Association
Input: $\mathbf{p}_{k+i}$, $\mathbf{p}_k$  // List of peaks at two time steps with
    a transform value and angle in degrees for each peak
Output: $\mathbf{D}$  // List of distances
for $r = 1 \rightarrow sizeof(\mathbf{p}_k)$ do
    $\mathbf{D}[r] \leftarrow kNN(\mathbf{p}_{k+i}[:], \mathbf{p}_k[r])$
end for
```

Using the list of distances $\mathbf{D}$ from Algorithm 1 we sort and find the best association of each peak in $\mathbf{p}_k$ with a single peak in $\mathbf{p}_{(k+i)}$. The resulting set of peak matching distances characterizes appearance confidence as,

$$Conf = \begin{cases} 1 - D_{min}/Th_1, & \text{if } D_{min} < Th_1 \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

where D_{min} is the minimum value in the distance array $\mathbf{D}$ and represents the best association among the set of strongest peaks that in the ideal case would be zero yielding the highest confidence value of one.

2.5. Ellipse-Fitting to Estimate Position

Once we have estimated the vehicle orientation and have high confidence in the appearance model we can localize the vehicle position more accurately. We find the largest connected component [27, 28] using the contours in the second-order structureness images $I_{\mathrm{k,H}}(x, y)$ and $I_{\mathrm{k+i,H}}(x, y)$ from Eq 4. The assumption so far is that we have a good enough edge map to describe the object shape and that we can extract the object template using the largest connected component within the local neighborhood. After fitting an ellipse to the largest connected component we estimate the new bounding box $B_{\mathrm{k+i}}$, which is desirable at this stage since we initially assumed that the previous template and the new instance of the object have similar shape parameters which may no longer hold true. The effectiveness of this step is demonstrated in Fig. 2.

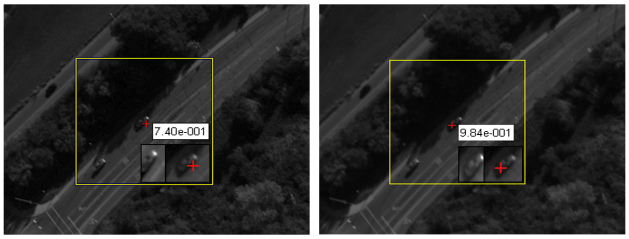

Figure 2. Improved appearance model leads to more accurate position estimate for car C1_2_0 and a longer track. The left image has a weak appearance model and leads to early loss of track. The yellow square is the 150 x 150 pixel search window; red plus sign is target localization.

3. Appearance Update Strategy

Appearance update strategies have a significant influence on the overall quality of the tracker by striving for high precision while increasing recall which is a challenging tradeoff to achieve [19]. We maintain several update strategies and select between them based on context and cues about the environment as well as target behavior. The update strategy presented here

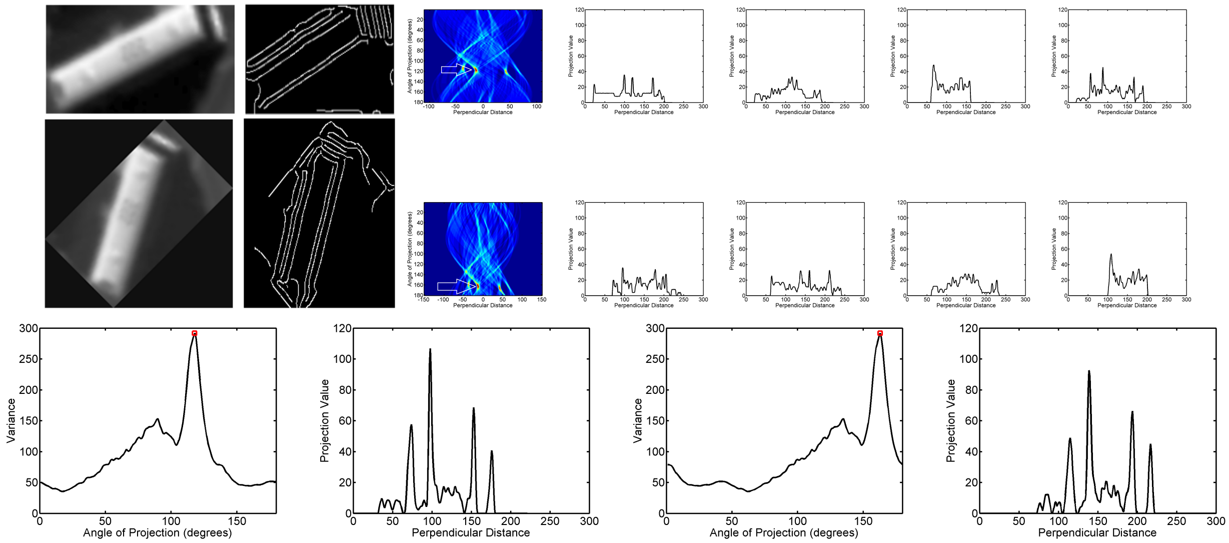

Figure 1. Vehicle orientation estimation using the Radon transform. Top row shows original template C5_3_7, thresholded second-order multiscale structureness map, resulting Radon transform, Radon transform profiles for $theta = 1, 45, 90, 135$. Middle row is for template C5_3_7 rotated by 45°. Bottom row: Max peak in variance vs angle (wrt horizontal axis) plot, shown in red, occur at 118° and 163°, for the two templates from which the change in orientation is estimated as 45°; corresponding transform profiles show the similarity in peak structure used for matching.

tries to address the *stability and plasticity* dilemma by employing the use of information such as amount of change in the pose and scale of the object, any illumination changes, and time interval since the last update. Using the likelihood-based Eq. 1, one update condition,

$$T_{k+1} = \begin{cases} \text{Extract}(I_{k+1}, (x_{\text{k+1}}, y_{\text{k+1}})), \\ T_k, \quad \text{if angleDiff} \notin [A_L, A_U] \end{cases} \tag{8}$$

checks *angleDiff*, the difference in orientation between current and previous vehicle templates compared to a user specified angle range. The function Extract(·) returns an image patch centered at the given coordinates. Alg. 2 covers a more extensive set of tests. Fig. 3 shows a case where active appearance updating improves performance by incorporating changes in the template model to maintain the track without drifting.

4. Evaluation Using CLIF

The impact of using adaptive appearance updating on LoFT performance was assessed using the challenging CLIF dataset [20]. We used the same 14 sequences as in previous papers [29, 2] which provide more details about the targets and sequences. We point out that the original CLIF sequences contain duplicated frames in some cameras to compensate for missing frames which adversely affects Kalman filter predictions of target position. LoFT v1.3F tracking performance in our tests used the same modules for feature fusion (i.e. Hessian_new) and Kalman filter motion prediction (offset) as described in [1, 2] without background subtraction, local registration (since the data was registered off-line), vehicle classification or MHT. We use precision and recall scores as a metric where precision measures the degree of repeatability of a video-tracking result [30],

$$\text{Precision} = \frac{|TP|}{|TP| + |FP|}. \tag{9}$$

Algorithm 2 Appearance Update Decision

```
Input: $Angle(T_{\text{t+i}}), Angle(T_{\text{t}})$
  $threshAngleU, threshAngleL$
   //Allowed Upper and Lower angles in degrees
  $threshAngleC$ //Appearance confidence threshold
  $initTurn$ //Flag to denote initialized on a turn
  $updateAppStatus$ //Check frame count since last update
  $appConfidence$ //Object appearance confidence
Output: $updateAppearance$ //Appearance update flag
  $angleDiff \leftarrow |Angle(T_{\text{t+i}}) - Angle(T_{\text{t}})| \,\mathbf{mod}\, 180$
  $updateAppearance \leftarrow 0$
  if $initTurn$ AND $appConfidence \geq threshAngleC$ OR $updateStatus$ AND $appConfidence \geq threshAngleC$ then
    $updateAppearance \leftarrow 1$
    return
  end if
  if $angleDiff \geq threshAngleL$ AND $angleDiff \leq threshAngleU$ AND $appConfidence \geq threshAngleC$ then
    $updateAppearance \leftarrow 1$
  end if
```

Recall is the ratio of the correctly tracked frames to ground-truth frames,

$$\text{Recall} = \frac{|TP|}{|TP| + |FN|}. \tag{10}$$

For a frame to be marked as correctly tracked for precision-recall metrics, the center was required to be within 20 pixels of the ground-truth center. We also use the Missing Frame Rate (MFR) metric [29], with a 1% overlap between the tracking bounding box and the ground-truth box as the match criteria; note that MFR has no explicit penalty for the size of the box. Since the two match criteria are different, $MFR \approx 1 - \mathit{Recall}$. Mean distance is defined as the distance between the last tracked point in a track generated by a tracker to the corresponding point in ground-truth. Note that this metric does not penalize the tracker for the remaining ground-truth. Table 1 shows the tracking results with and without the update scheme. It can be seen that the tracker using appearance updating consistently outperforms the one without the update scheme, with average precision and recall being 3% and 6% higher respectively. The average MFR is about 5% higher with appearance updating.

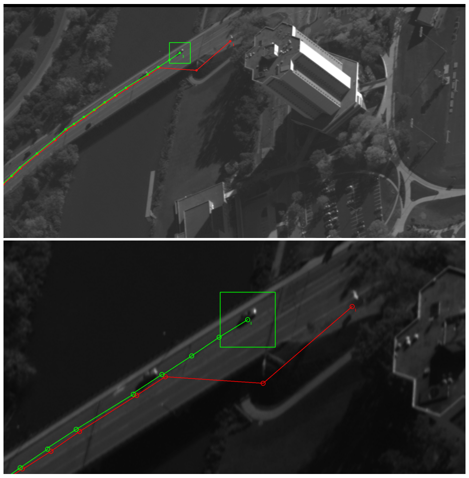

Figure 3. Appearance updating improves performance. The red track shows the LoFT trajectory without using appearance updating and the green track with updating. Although C1_2_0 has minimal occlusions and does not have any turns, its appearance over time accumulates small changes that causes the tracker to drift. Bottom image is a zoomed view.

5. Conclusion

We presented a novel active appearance balanced update strategy using a Radon transform-based estimation of vehicle orientation and visual changes. The proposed approach reduces the drifting problem leading to measurable improvement in tracking performance on challenging aerial WAMI sequences. Using additional LoFT modules, incorporating multiple templates and on-line learning strategies offer further directions to improve overall performance towards longer tracks of higher purity.

Acknowledgments

Raphael Viguier, Mahdieh Poostchi and Filiz Bunyak assisted with validating and discussing the CLIF experiments. This research was partially supported by U.S. Air Force Research Laboratory (AFRL) under agreement AFRL FA8750-11-C-0091. Approved for public release (case 88ABW-2012-3914). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies of AFRL or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints.

References

[1] K. Palaniappan and *et al.*, "Efficient feature extraction and likelihood fusion for vehicle tracking in low frame rate airborne video," in *13th Int. Conf. Information Fusion*, 2010, pp. 1–8.

[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," in *15th Int. Conf. Information Fusion*, 2012.

[3] A. Hafiane, G. Seetharaman, K. Palaniappan, and B. Zavidovique, "Rotationally invariant hashing of median patterns for texture classification," *Lecture Notes in Computer Science*, vol. 5112, pp. 619–629, 2008.

[4] K. Palaniappan, R.M. Rao, and G. Seetharaman, "Wide-area persistent airborne video: Architecture and challenges," *Distributed Video Sensor Networks*, pp. 349–371, 2011.

[5] A. Haridas, R. Pelapur, J. Fraser, F. Bunyak, and K. Palaniappan, "Visualization of automated and manual trajectories in wide-area motion imagery," in *Int. Conf. Information Visualisation*, 2011, pp. 288–293.

[6] E.P. Blasch, P.B. Deignan, S.L. Dockstader, M. Pellechia, K. Palaniappan, and G. Seetharaman, "Contemporary concerns in geographical/geospatial information systems (GIS) processing," *IEEE National Aerospace and Electronics Conference (NAECON)*, pp. 183–190, 2011.

[7] T.F. Cootes, G.J. Edwards, and C.J. Taylor, "Active appearance models," *IEEE Trans. PAMI*, vol. 23, no. 6, pp. 681 –685, June 2001.

[8] I. Matthews and S. Baker, "Active appearance models revisited," *Int. J. Computer Vision*, vol. 60, no. 2, pp. 135–164, 2004.

Image	GT	Track Length		Correctly Tracked		False Alarms		Precision		Recall		Mean Distance		MFR	
Sequence		w update	w/o update	w update	w/o update	w update	w/o update	w update	w/o update	w update	w/o update	w update	w/o update	w update	w/o update
C0_3_0	50	49	49	12	2	37	47	0.244	0.040	0.240	0.040	282	146	0.800	0.960
C1_2_0	27	25	25	25	22	0	3	1.000	0.880	0.925	0.814	13	36	0.074	0.241
C1_4_0	50	48	22	18	16	30	6	0.375	0.723	0.360	0.320	441	338	0.640	0.756
C1_4_6	50	48	48	26	27	22	21	0.541	0.562	0.520	0.540	49	48	0.480	0.480
C2_4_1	49	34	28	6	4	28	24	0.176	0.142	0.120	0.080	968	1032	0.920	0.944
C3_3_4	27	3	3	2	2	1	1	0.666	0.666	0.074	0.074	513	513	0.920	0.960
C4_1_0	18	25	29	3	2	22	27	0.120	0.068	0.166	0.111	168	236	0.880	0.888
C4_3_0	20	18	18	18	12	0	6	1.000	0.666	0.900	0.600	10	42	0.200	0.400
C4_4_1	30	28	28	28	28	0	0	1.000	1.000	0.933	0.933	5	12	0.034	0.066
C4_4_4	13	15	15	14	13	1	2	0.933	0.866	1.000	1.000	13	12	0.000	0.000
C5_1_4	24	7	6	2	2	5	4	0.285	0.333	0.083	0.083	1195	1304	0.916	0.952
C5_2_0	49	4	4	2	2	2	2	0.500	0.500	0.040	0.040	11	14	0.950	0.978
C5_3_7	27	26	26	26	20	0	6	1.000	0.769	0.962	0.740	220	676	0.037	0.071
C5_4_1	19	46	46	19	19	27	27	0.413	0.413	1.000	1.000	245	220	0.000	0.000
Average	-	**26.8**	24.7	**14.3**	12.4	12.5	**12.3**	**0.589**	0.557	**0.528**	0.466	**428**	460	**0.489**	0.539
Overall	-	**-**	-	**-**	-	-	**-**	**0.534**	0.511	**0.444**	0.347	**-**	-	**0.574**	0.597

Table 1. LoFT performance on 14 CLIF sequences with and without appearance updates showing Correctly Tracked frames (higher is better), False Alarms, Precision, Recall, Mean Distance error (smaller is better), and Missing Frame Rate (MFR). GT updated and MFR calculation corrected from [2], C_5_4_1 GT skips duplicate frame 3.

[9] F. Bunyak, K. Palaniappan, S. K. Nath, and G. Seetharaman, “Geodesic active contour based fusion of visible and infrared video for persistent object tracking,” in *8th IEEE Workshop Applications of Computer Vision (WACV 2007)*, Austin, TX, Feb. 2007, p. Online.

[10] F. Bunyak, K. Palaniappan, S. K. Nath, and G. Seetharaman, “Flux tensor constrained geodesic active contours with sensor fusion for persistent object tracking,” *J. Multimedia*, vol. 2, no. 4, pp. 20–33, August 2007.

[11] I. Ersoy, K. Palaniappan, R. Rao, and G. Seetharaman, “Tracking in persistent wide-area motion imagery,” in *Proc. SPIE Conf. Geospatial InfoFusion II (Defense, Security and Sensing: Sensor Data and Information Exploitation)*, 2012, vol. 8396.

[12] I. Ersoy, K. Palaniappan, and G. Seetharaman, “Visual tracking with robust target localization,” in *IEEE Int. Conf. Image Processing*, 2012.

[13] H. Grabner, M. Grabner, and H. Bischof, “Real-time tracking via on-line boosting,” in *Proc. BMVC*, 2006, vol. 1, pp. 47–56.

[14] B. Babenko, M. Yang, and S. Belongie, “Robust object tracking with online multiple instance learning,” *IEEE Trans. PAMI*, vol. 33, no. 8, pp. 1619–1631, 2011.

[15] D.A. Ross, J. Lim, R.S. Lin, and M.H. Yang, “Incremental learning for robust visual tracking,” *Int. J. Computer Vision*, vol. 77, no. 1, pp. 125–141, 2008.

[16] M.J. Black and A.D. Jepson, “Eigentracking: Robust matching and tracking of articulated objects using a view-based representation,” *Int. J. Computer Vision*, vol. 26, no. 1, pp. 63–84, 1998.

[17] X. Mei and H. Ling, “Robust visual tracking and vehicle classification via sparse representation,” in *IEEE Trans. PAMI*, 2011, pp. 2259–2272.

[18] B. Liu, L. Yang, J. Huang, P. Meer, L. Gong, and C. Kulikowski, “Robust and fast collaborative tracking with two stage sparse optimization,” *Proc. European Conf. Computer Vision*, pp. 624–637, 2010.

[19] S. Grossberg, “Competitive learning: From interactive activation to adaptive resonance,” *Cognitive science*, vol. 11, no. 1, pp. 23–63, 1987.

[20] Air Force Research Laboratory, “Columbus Large Image Format (CLIF) 2007 dataset,” *https://www.sdms.afrl.af.mil/datasets/clif2007/.*

[21] M. H. Kolekar, K. Palaniappan, S. Sengupta, and G. Seetharaman, “Semantic concept mining based on hierarchical event detection for soccer video indexing,” *J. Multimedia*, vol. 4, no. 5, pp. 298–312, October 2009.

[22] T.F. Cootes and P. Kittipanya-Ngam, “Comparing variations on the active appearance model algorithm,” in *Proc. BMVC*, 2002, pp. 837–846.

[23] A.F. Frangi, D. Rueckert, and J.S. Duncan, “Three-dimensional cardiovascular image analysis,” *IEEE Trans. Medical Imaging*, vol. 21, no. 9, pp. 1005 –1010, 2002.

[24] T. Lindeberg and D. Fagerström, “Scale-space with casual time direction,” *Proc. European Conf. Computer Vision*, pp. 229–240, 1996.

[25] T. Pock, R. Beichel, and H. Bischof, “A novel robust tube detection filter for 3d centerline extraction,” *Image Analysis*, pp. 55–94, 2005.

[26] K. Jafari-Khouzani and H. Soltanian-Zadeh, “Radon transform orientation estimation for rotation invariant texture analysis,” *IEEE Trans. PAMI*, vol. 27, no. 6, pp. 1004 –1008, 2005.

[27] M. B. Dillencourt, H. Samet, and M. Tamminen, “A general approach to connected-component labeling for arbitrary image representations,” *J. ACM*, vol. 39, no. 2, pp. 253–280, 1992.

[28] P. Kumar, K. Palaniappan, A. Mittal, and G. Seetharaman, “Parallel blob extraction using the multi-core Cell processor,” *Lecture Notes in Computer Science*, vol. 5807, pp. 320–332, 2009.

[29] H. Ling and *et al.*, “Evaluation of visual tracking in extremely low frame rate wide area motion imagery,” *Int. Conf. Information Fusion*, pp. 1–8, 2011.

[30] E. Maggio and A. Cavallaro, *Video Tracking: Theory and Practice*, Wiley, 2011.