Efficient Feature Extraction and Likelihood Fusion for Vehicle Tracking in Low Frame Rate Airborne Video

Kannappan Palaniappan[1], Filiz Bunyak[1], Praveen Kumar[1], Ilker Ersoy[1], Stefan Jaeger [1], Koyeli Ganguli[1], Anoop Haridas[1], Joshua Fraser[1], Raghuveer M. Rao[2], Guna Seetharaman[3]

[1]Dept. of Computer Science, University of Missouri, Columbia, MO 65211, USA

[2]Army Research Laboratory, Adelphi, MD 20783, USA

[3]Air Force Research Laboratory, Rome, NY 13441, USA

Abstract – *Very large format video or wide-area motion imagery (WAMI) acquired by an airborne camera sensor array is characterized by persistent observation over a large field-of-view with high spatial resolution but low frame rates (i.e. one to ten frames per second). Current WAMI sensors have sufficient coverage and resolution to track vehicles for many hours using just a single airborne platform. We have developed an interactive low frame rate tracking system based on a derived rich set of features for vehicle detection using appearance modeling combined with saliency estimation and motion prediction. Instead of applying subspace methods to very high-dimensional feature vectors, we tested the performance of feature fusion to locate the target of interest within the prediction window. Preliminary results show that fusing the feature likelihood maps improves detection but fusing feature maps combined with saliency information actually degrades performance.*

***Index Terms*—Video object tracking, feature fusion, wide-area motion imagery, persistent sensor array**

I. Introduction

New device fabrication technologies and heterogeneous embedded processors have led to the emergence of a new imaging sensor design sweet-spot known as wide-area motion imagery [14]. WAMI sensors consist of an airborne imaging camera array to create a high numerical aperture optical system on a single platform. WAMI also referred to as wide-area persistent airborne video or very large format video poses a new set of challenges in object tracking. The most significant of which include low frame rate sampling, imprecise georegistration, limited spatial resolution, low dynamic range, spatially varying optical transfer function across the effective camera array, parallax effects due to changing pose, geometric occlusions between target and sensor, motion blur, urban scene complexity, and high data volumes. In practical terms the targets are small in size, often have low contrast, have large displacements against a shifting ground-plane, are often occluded by buildings that *wobble* and are embedded among many distractors. Although manually tracking vehicles for many hours can be accomplished with patience, continuous automatic tracking of any and all vehicles in low frame rate WAMI has yet to be demonstrated.

Wide-area video sensor platforms typically follow a continuous circular flightpath in a fixed 3D plane perpendicular to the local ground plane with the orientation of the camera array accurately gimbaled to a fixed point on the ground. The flight pattern combined with the effective field-of-view of the camera array enables persistent coverage of tens of square miles for long periods of time. This spatiotemporal coverage pattern cannot be accomplished using satellite imaging and would be more complex using a collection of distributed airborne narrow-field-of-view video sensor networks. Airborne camera arrays combined with computational photography techniques enable the spatial and temporal integration of multi-camera scene information using the plenoptic function [14].

Figure 1 shows some of the key software modules for our low frame rate aerial imaging vehicle tracking system. This paper focuses on a few aspects of the overall system, namely feature likelihood fusion, efficient computation of features and object saliency. Section II describes the imaging system and associated challenges in vehicle tracking. Section III describes the derived rich set of features used for appearance modeling. Section IV describes fast computation of feature likelihood maps using integral histogram methods. Section V explores the static and dynamic saliency models. The different types of fusion methods are described in Section VI followed by experimental results and conclusions.

II. Imaging Array Characteristics

In this paper we use imagery acquired from an eight-camera array built by Persistent Surveillance Systems. Each camera in the array produces an 11 megapixel 8-bit grayscale image typically 4096×2672 at one to four frames per second. These raw images are georegistered

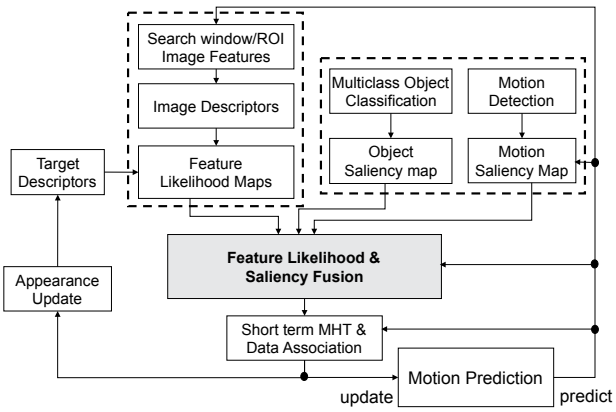

Fig. 1: LOFT wide-area low frame rate video tracking system is based on extracting a derived rich set of feature descriptors to model appearance, combined with feature fusion based on Bayesian likelihood estimation, object saliency computation using an object classifier and motion detector. A Kalman filter model is used for motion prediction and vehicle dynamics.

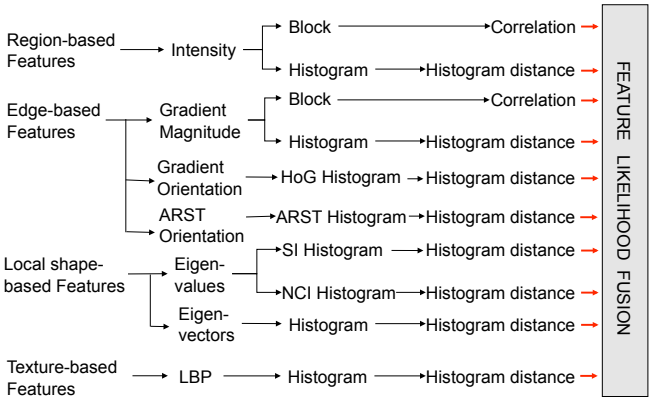

Fig. 2: Rich set of derived feature descriptors and feature fusion used for vehicle appearance modeling and target to search window matching. This is part of the Feature Likelihood and Saliency Fusion block shown in Fig. 1.

to a 16384×16384 image mosaic with a ground sampling distance of about 25cm for the imagery used in this paper. At this spatial resolution and a temporal sampling rate of one hertz the data volume is about one terabyte per hour. More details of the optical characteristics of the camera array imaging system and processing challenges can be found in [14].

Objects moving steadily across the WFOV can persist and stay visible for long durations with intermittent to extended occlusions. Low frame rate sampling leads to large object displacements which leads to a matching and detection based tracking paradigm with kinematics providing a rough region of interest constraint rather a precise guidance of position as is typical at standard video frame rates. Blob segmentation, track initiation, target reacquisition, occlusion handling and pair-wise relations between moving targets are complex vision tasks even in regular airborne or ground-based video [3], [21], [23] that need to be further extended to the WAMI domain to support exploitation of city-wide and region-wide scene activity analysis. Established approaches for image registration, interpolation, segmentation, video stabilization, motion analysis, and structure from motion algorithms [1]–[3], [6], [12], [15], [18], [24] have to be modified and extended to explicitly exploit the persistent viewing geometry in wide-area video. Some applications of (non-persistent) WFOV color images collected using bursty sampling to address the low frame rate sampling for vehicle tracking and traffic pattern analysis are described in [8], [17].

Parallax effects (which are particularly severe in dense urban scenes) along with spatial camera-to-camera registration and georegistration errors prevent direct use of detection algorithms relying on motion information through variations of background subtraction and optical flow analysis. Furthermore, parallax causes temporary occlusions of nearby objects and roads which result in temporary loss of tracked objects. Target appearance changes are observed due to partial occlusions, target pose, and camera viewing angle. Unlike satellite images, visual imagery of objects obtained from orbiting airborne platforms have considerable appearance change due to viewing angle differences over short times especially in the periphery FOV. Low contrast and lack of color conflates vehicles with roads since any mid-intensity or dark-intensity vehicle often matches the mid-gray or dark-asphalt of roads.

III. Object Detection and Feature Descriptors

A wide range of feature and appearance models have been described in the video and target tracking literature. Specific characteristics of wide-area imagery including lack of color, small target size, perspective distortion, and small support maps greatly limit the use of complex appearance models. So we rely on a derived rich set of complementary low level image-based feature descriptors incorporating intensity, edge, texture, and shape information. Motion cues are an important feature that is incorporated through motion saliency estimation (Fig. 1).

The features used in our Low Frame Rate Tracking (LOFT) system can be grouped into four categories: region-based, edge-based, local shape-based, and texture-based (Fig. 2). Block-based similarity measures such as intensity and gradient cross-correlations incorporate spatial information, that histogram/distribution-based similarity measures lack and provide better discrimination power, but are sensitive to pose and viewing orientation. On the other hand histogram-based techniques provide global information about objects and image windows that are tolerant of small changes due to motion, illumination, pose, or viewing direction. We primarily use histogram-based descriptors and similarity measures except for the normalized intensity and gradient magnitude correlation to estimate feature

likelihood maps.

Gradient magnitude normalized cross-correlation and gradient magnitude histograms are edge-based features computed similar to their intensity counterparts. Gradient orientation information is captured using the histogram of oriented gradients (HOG) descriptor which has been successfully used in many recent object and people detection applications [5], [20]. HOG bins the gradient magnitude weighted gradient orientations over an image patch and is a dense version of the popular scale-invariant feature transform (SIFT) descriptor. Robust orientation estimation is important for HOG-like descriptors. Our novel extension uses more accurate orientation estimation based on the adaptive robust structure tensor (ARST-HOG) [11]. Structure tensors are a useful tool for reliably estimating oriented structures within a neighborhood even in the presence of noise. In our preliminary car detection results ARST-HOG outperformed standard HOG.

Local shape-based features are measured using the eigenvalues of the Hessian matrix $\mathcal{H}$, of the intensity field $I(x, y)$, that describes the second order structure of local intensity variations around each image point,

$$\mathcal{H}(x,y) = J(\nabla I) = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix}, I_{xy} = \frac{\partial^2 I}{\partial x \partial y}. \quad (1)$$

Two measures of local shape are the shape index (Eq. 2) and the normalized curvature index (Eq. 3) features derived from the eigenvalues, $\lambda_1 \geq \lambda_2$, of $\mathcal{H}$,

$$\theta(x,y) = tan^{-1}\frac{\lambda_2(x,y)}{\lambda_1(x,y)} \equiv \text{ATAN2}(\lambda_{min}, \lambda_{max}), \quad (2)$$

$$\phi(x,y) = tan^{-1}\frac{(\lambda_1(x,y)^2 + \lambda_2(x,y)^2)^{1/2}}{(1 + I(x,y))}. \quad (3)$$

In the experiments results an unsigned ordering of the eigenvalues was used. A third shape measure is the magnitude weighted histogram of the Hessian eigenvector orientations. This descriptor is similar to HOG but with orientations corresponding to Hessian (Jacobian of image gradient) eigenvectors instead of $\nabla I(x, y)$.

Textures are easy to recognize but hard to define. Texture analysis approaches include features of co-occurrence matrices, spatial filtering, random field models and texton pattern modeling. A simple texture measure that combines statistical and structural models of texture is based on the local binary pattern (LBP) histogram [13]. The LBP characterizes the quantized local intensity variability and various extensions to LBP have been proposed including the median binary pattern (MBP) [7]. We use the uniform rotation-invariant LBP consisting of 18 unique patterns.

Feature likelihood maps are computed using sliding window histogram differencing. Local maxima in the feature likelihood maps that exceed a threshold are considered as high probability target locations.

IV. Efficient Feature Extraction

Increasing the number of features or dimensionality of the descriptor space typically improves vehicle detection and tracking processes, but at the expense of increasing computation time considerably. Efficient computation of both features and (dis)similarity measures become critical for approaches that use fusion of multiple features. Our system uses the following approach to achieve speed: 1) efficient file I/O, 2) separable filters, 3) fast correlation computation, and 4) fast sliding window histogram computation.

Considering that the typical size of our input images are $16K \times 16K$, file I/O is crucial for efficient feature estimation. Rather than maintaining the full image in memory, we use direct access to a much smaller region of interest (ROI) using our specific pyramid file format with efficient out-of-core access mechanisms. Feature computation is accelerated using separable filters for gradient and Hessian matrix estimation. Fast normalized cross correlation using integral images [10] is used to compute intensity and gradient magnitude correlations.

In order to dramatically speed up the calculation of sliding window histograms needed for the feature likelihood maps we use the integral histogram method which improves performance by a factor of 60,000 or more for 200×200 pixel gray level search windows [16]. The integral histogram is a recursive propagation method that works in Cartesian spaces. In 2D the integral histogram $H(x, y)$, corresponds to the histogram of $R([0, x], [0, y])$, the region between the origin $(0, 0)$ and location $p(x, y)$. Integral histogram can be efficiently generated by propagation using the update equation,

$$\begin{aligned} H(x,y) = H(x-1,y) + H(x,y-1) \\ - H(x-1,y-1) + Q(I(x,y)) \end{aligned} \quad (4)$$

with $H(0, 0) = 0$. $Q(\cdot)$ is a histogram length vector with one non-zero element used to increment the appropriate bin in the integral histogram H,

$$Q(I(x,y)) = \begin{cases} 0 & I(x,y) \neq b^* \\ w(x,y) & I(x,y) = b^* \end{cases}$$

where b^* is the bin associated with $I(x, y)$, $w(x, y)$ is equal to 1 for regular histograms, and equal to a weight in weighted histograms such as the gradient magnitude weighted HOG. Regional histograms corresponding to sliding window, $R(]x1, x2],]y1, y2])$, with sides parallel to the image coordinates can be computed from the integral histogram using three additions:

$$\begin{aligned} h_R(]x1,x2],]y1,y2]) = & H(x_2,y_2) - H(x_2,y_1) \\ & - H(x_1,y_2) + H(x_1,y_1) \end{aligned} \quad (5)$$

where h_R is the histogram of the rectanglular region R and H is integral histogram (Fig. 3).

V. Static and Dynamic Saliency

Static saliency or object saliency refers to focusing interest in those image regions likely to contain targets of interest using an object classification approach. For classification, we use a (binary) support vector machine

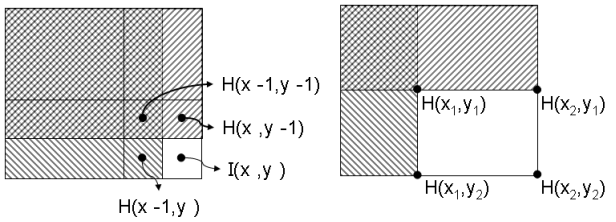

Fig. 3: Efficient histogram computation using integral histograms showing: (a) integral histogram generation through propagation (left), b) regional histogram computation using integral histograms (right).

(SVM) [19] to discriminate between vehicles and non-vehicles. To train the SVM classifier, we manually cropped about 250 image chips from a typical urban scene in the Philadelphia wide-area imagery. Each image chip is about the size of a vehicle but may be of different sizes depending on the shape and orientation of the vehicle. Each training image chip contains a vehicle along with some background pixels and each non-vehicle image chip is based on a random sampling of typical road surfaces and surrounding scene pixels, where the latter can contain street segments together with parts of buildings or sidewalks and other non-vehicle objects. Figure 4 shows a few typical image chips from the classifier training dataset. The image chip examples illustrate the complexity of the classification problem. The vehicles are of small size, often with very low contrast compared to the background road surface and without discernible shape detail especially for dark vehicles. For each of the image chips, we compute feature maps as described in Section III, resulting in a 78-dimensional feature vector to describe each image chip. The classification of each image chip is performed by a linear support vector machine trained using these 78-D labeled feature sets and achieves a recognition rate of about 90% based on 5-fold cross-validation.

To generate the saliency map from the classifier output, we compute a confidence value along with each class label. The confidence values of the saliency map indicate how likely it is that a specific pixel belongs to a vehicle

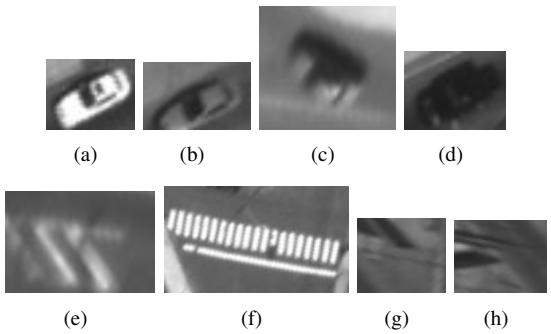

Fig. 4: Typical image chips of vehicles (a-c) and non-vehicles (d-g) used to train the SVM classifier.

object. In our case, the classifier confidence value is the distance of a feature vector to the separating hyperplane of the trained support vector machine. We normalize all confidence values using a simple linear mapping that we compute using the training data. In future work, we plan on using a more generalized technique for computing confidence values.

Dynamic saliency or motion saliency refers to the use of local background estimation to detect changes that can be attributed to moving targets. Since local background modeling requires precise alignment of temporally adjacent frames we use a fast frequency domain-based registration algorithm for local image stabilization. Predicting the location of the search window at the next time step is a key requirement for reliable object tracking especially in urban scenes with dense traffic and complex backgrounds. Accurate prediction requires a good dynamical model of the vehicle behavior and a smooth trajectory which requires image-to-image stabilization to remove the frame-to-frame jitter. Even though the Kalman filter can do limited smoothing of the trajectory, it is not sufficient for low frame rate WAMI due to large georegistration errors from limited accuracy of the inertial measurement unit (IMU), IMU shot noise and non-linear affects due to multiple camera seams and other artifacts. To improve the Kalman prediction, we stabilize two consecutive frames in a local neighborhood (512×512) using Fourier transform-based registration. Fourier methods were selected for this application due to a combination of speed with reasonable quality under different noise conditions, occlusion and timevarying illumination.

The FFT-based approach to registration estimates the inter-frame translation using the Fourier shift theorem. In the ideal case when two identical images differ only by a translation, their Fourier transform has the same magnitude but with an additional phase-shift. The cross-power spectrum can be used to estimate this phase difference,

$$R(u,v) = \frac{\mathcal{F}(I_t)\mathcal{F}^*(I_{t+1})}{|\mathcal{F}(I_t)\mathcal{F}^*(I_{t+1})|} = e^{2\pi i(\frac{u}{M}\Delta x + \frac{v}{N}\Delta y)}, \quad (6)$$

where I_t and I_{t+1} are consecutive images that differ by a translation of $(\Delta x, \Delta y)$, $\mathcal{F}(I)$ is the Fourier transform of I, $\mathcal{F}^*(I)$ its complex conjugate, M and N are image dimensions and R is the normalized cross-power spectrum. Again in the ideal case the inverse Fourier transform of R will have a peak at $(\Delta x, \Delta y)$. Once translation of the local region around the object is estimated, the image pair can be locally stabilized and the Kalman predictor applied to the stabilized (smooth) trajectory. Each new frame is stabilized with respect to the previous frame and a cumulative translation estimated and applied to the new centroid in order to place the tracked object in the coordinate system of the *first* frame in the sequence. This ensures that the prediction filter always operates on a stabilized target trajectory.

VI. Vehicle Detection Using Information Fusion

Since we use multiple feature cues to improve detection and tracking performance, a systematic approach to fuse heterogeneous information from multiple features is required. There are many potential approaches to robust feature fusion some of which are discussed in [9] . We fuse the saliency and feature likelihood map information using the Variance Ratio method [4], [22].

1) Feature Likelihood Fusion: The result from the previous stage contains all possible cars among which we have to detect the target car. We generate feature likelihood maps for each feature as described in § III. Each feature performs differently depending on the car characteristics, scene illumination and many other environment factors that often change with time. Therefore, rather than using a fixed set of weights for features, we adaptively determine the weights of each feature likelihood map L by comparing the foreground and background separability of the current and previous frame likelihood maps. The local region surrounding the previous object position is such that there are as many background pixels as object pixels.

The assumption here is that the target detection in the previous frame was accurate and that conditions do not change significantly between two successive frames. Our strategy for determining the adapting weights is based on a Bayesian formulation leading to the Variance Ratio method (VR) [4], [22] which weights features according to their power to discriminate between the foreground (tracked object) and surrounding background,

$$\text{VR}(L_i; p, q) \equiv \frac{var(L_i, (p+q)/2)}{[var(L_i, p) + var(L_i, q)]}, \qquad (7)$$

where L_i is the feature likelihood map, p represents the class of object pixels and q represents the class of background pixels. The numerator of VR is the between class variance i.e, the variance of L_i over both object and background pixels. A higher value of this quantity means that both object and background values are more spread out, which is desirable. The denominator contains the within class variance because we prefer features that minimize these variances i.e., we choose features that tightly cluster background and foreground pixels. Higher values of VR mean more discriminative power for that feature in distinguishing between the appearance of the object and the background. The weights w_i are approximated using the normalized VR,

$$w_i \approx \frac{VR(L_i; p, q)}{\sum_{j=1}^{N} VR(L_i; p, q)}. \qquad (8)$$

Since the VR for each feature is compared in a relative manner, we need to ensure that feature likelihood maps, L_i, are normalized to fit within the range 0 to 1.

We found that template correlation features give more peak-like response in contrast to the smooth response of histogram matching features. This is because template matching using normalized cross-correlation uses local sums to normalize the cross-correlation, giving high likelihoods within a small local region around the center of the object window and low likelihood values for the rest of the object window. This make it difficult to weight the template-based correlation features using the VR so we use a different approach to estimate their weights for fusion. We determine the number of local/regional maxima within 90% of the maximum probability which gives the number of matched peaks, m, and the correlation feature weights are updated as, $w_i \approx (M_i \sum_{j=1}^{m} M_i)^{-1}$. If there are more number of distracters for the feature, it will produce a larger number of peaks and reduce the corresponding weight in the fusion process. We first combine the histogram based features using the weights based on Eq. 8 and then combine the resultant feature likelihood map with the correlation based features using the modified weighing scheme.

2) Saliency Fusion: Information from object and motion saliency maps are used to reduce the likelihood of non-car and static regions (like roads and buildings) from the search window. The motion saliency map is fused by multiplying the motion probability with the fused feature probability map. However, if the predicted velocity of the target car is less than a threshold then the motion map is not reliable as is the case with parked vehicles, slowing vehicles, bunched up vehicles, traffic congestion, *etc.* In this case, the motion map is not used at all. The object saliency map is used as a binary mask by thresholding the confidence provided by the car/no-car classifier.

VII. Experimental Results

We tested our low frame rate vehicle tracking algorithm using very large format video acquired by the PSS platform over the Philadelphia urban region. The flight altitude was between 4000 ft and 4300 ft with an average car size of 35×42 pixels. Only the feature-based target vehicle detection performance using five manually tracked cars is reported. Data characteristics for each vehicle are summarized in Table I. Quantitative evaluation of each individual feature in the rich image-based feature set is compared to the performance using different combination of feature fusion methods.

Figure 5 shows the probability maps obtained using each feature separately and after fusion for a sample white car which is very distinctive from the other cars within the search window shown in Fig. 5(a). It can be observed from the probability maps of the features that certain features are more discriminative than others. Fig. 5(k) shows the weights computed using VR of the histogram features to estimate the fused probability map Fig. 5(j). The normalized intensity and gradient correlation also perform well in this case, as they give distinct peaks near the car and thus receive higher weights as shown in Fig. 5(o). The last row shows the motion and object saliency maps and how their combination helps to filter out non-moving/parked cars (Fig. 5(q)) and non-car regions (Fig. 5(s)).

	Car-1	**Car-2**	**Car-3**	**Car-4**	**Car-5**
#Frames	254:(700-954)	138:(700-838)	118:(700-818)	266:(700-966)	98:(700-798)
Intensity Color	White	White	White	Gray-White(736-809)-Gray	White
Occlusions	41:(735-750), (803-811), (919-934)	1:(837)partial	1:(818)partial	17:(793),(908-911), (923-926),(934-939, 942, 945) partial	19:(779-797)
Turns	3:(911,919,935)	2:(810,822)	2:(711,788)	1:(914)	0
Stationary	-	11:(734-744)	42:(747-788)	41:(863-903)	All frames
Distortion/blur	106:(790-895)	51:(769-819)	-	112:(763-874)	56:(742-797)
Seams on car	3	-	1	5	-

TABLE I: Test data set characteristics (number between brackets are start and end frame numbers).

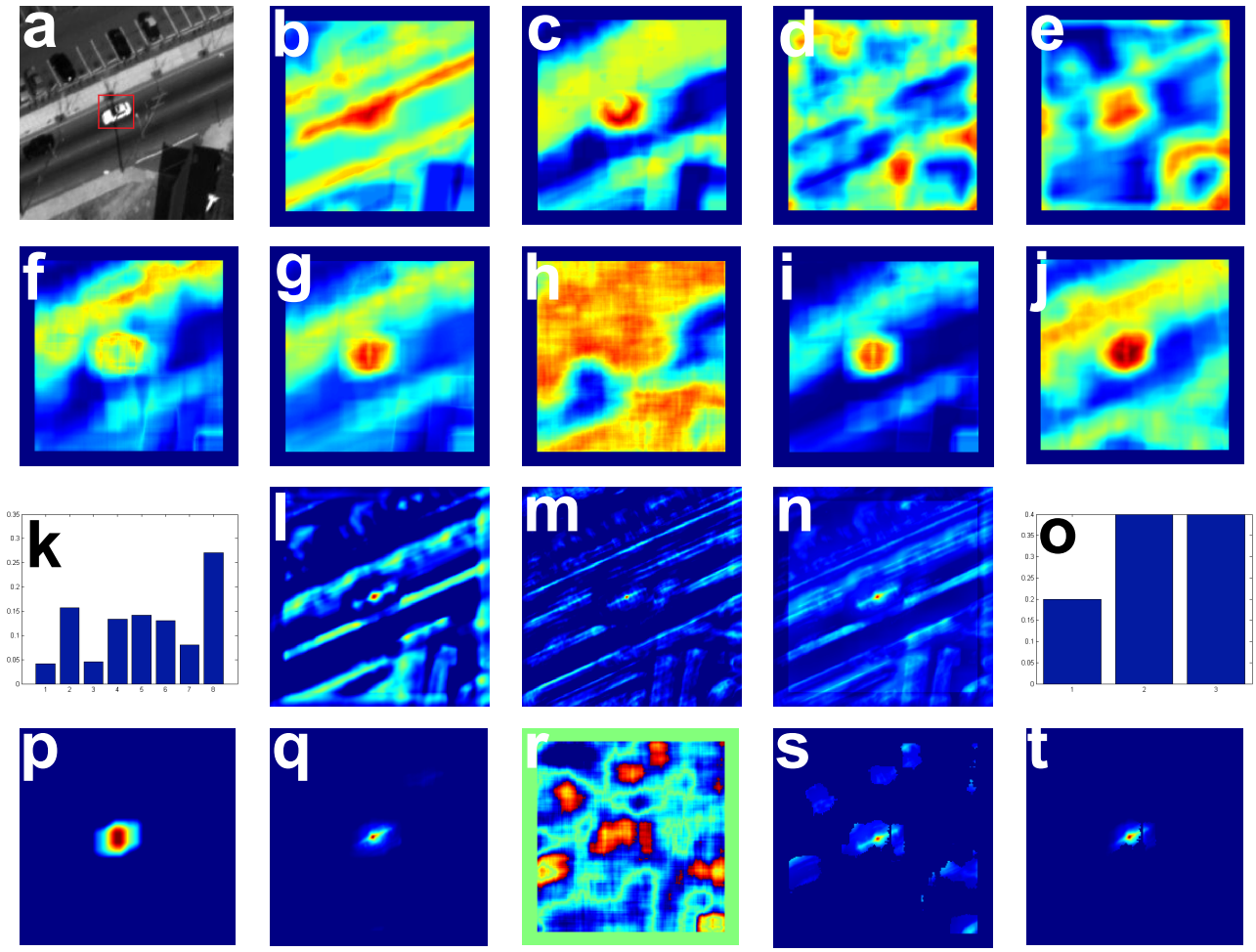

Fig. 5: Probability maps obtained from different features and after fusion for a white car in one specific frame and for the selected search window shown: a) ROI/search window with target car highlighted, b)intensity histogram, c) gradient magnitude histogram, d)shape index, e) normalized curvature index, f) histogram of Hessian eigenvector orientations, g) histogram of gradient orientations (HoG), h) LBP histogram, i) HOG on gradients from adaptive robust structure tensor, j) fused probability map from the histogram features, k) weights for the histogram features (in order b to i) to get j, l) intensity normalized cross-correlation, m) gradient magnitude normalized cross correlation, n) fused probability map from all features, o) weights for j,l and m to get n, p) motion saliency, q) fused probability map with motion, r) object saliency (object classification), s) fused probability map with classification, t) fused probability map using all features and saliency information.

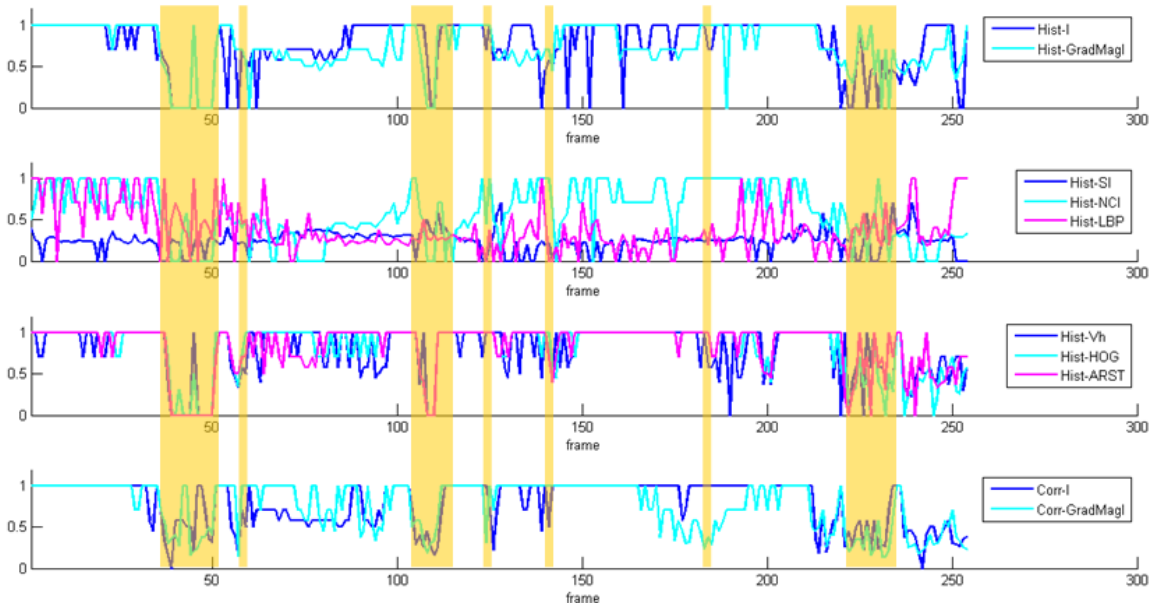

Fig. 6: Vehicle detection performance for each feature using Car-1 test data. Vertical axis is inverse square root of rank and yellow shading along the horizontal time axis marks frames when the target vehicle is occluded.

Vehicle detection performance using the rich feature set described in §III was evaluated using five manually tracked vehicles. Detection performance is measured as $(\sqrt{Rank_{car}})^{-1}$where $Rank_{car}$ is the rank of the local peak in the probability map corresponding to the target vehicle with peaks ranked in decreasing order of probability. In the ideal case the highest Rank 1 peak corresponding to the most likely location of the vehicle being tracked would be centered on the target vehicle. As the peak rank increases numerically, higher values correspond to lower confidence peaks being associated with the target vehicle. Decreasing performance converges towards zero and is identically zero when none of the local peaks within the search window correspond to the target vehicle. Detection results for the Car-1 test sequence is shown in Figure 6.

The overall classifier-based detection rate averaged across all frames for each of the test vehicles is as follows: Car-1 at 93.7%, Car-2 at 100.0%, Car-3 at 85.6%, Car-4 at 60.2% and Car-5 at 66.4%. Car-2 is always detected at each time step within the search window (centered on the manual ground truth), while Car-4 and Car-5 are much harder to detect. The reason that these two vehicles have lower recall rates is due to perspective induced geometric distortions in shape and orientation that increases classifier confusion. If we remove frames with significant perspective shape changes, then the hit rate for Car-5 improves to 78.6%.

Figure 7 shows the aggregate performance of different features and fused results in detecting the target car over all five test cases. It can be seen that fusion using all the features gives the best recall with significant improvement over all other features. However, fusion with motion and classification saliency lowers the overall recall. In the case of motion saliency there were cars of similar shape and

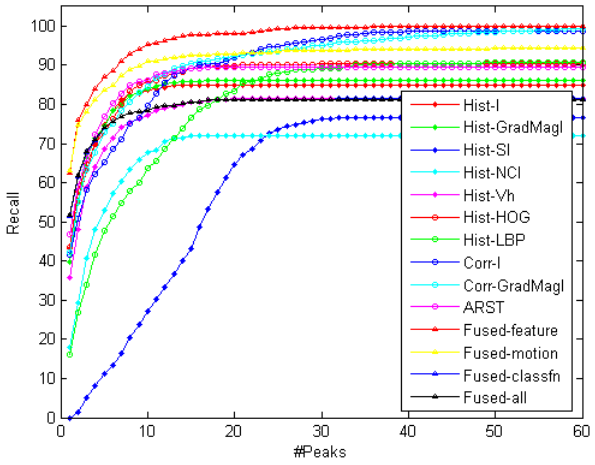

Fig. 7: Average recall versus number of peaks selected for each feature vector and fusion method using all frames averaged across the five test cases.

intensity moving at a velocity in which they appear to replace each other in successive frames due to the low temporal sampling rate and consequently motion is not detected at the target car position. In the case of classifier saliency, first, the classifier is trained on a general set of vehicles and not fine-tuned to a specific vehicle, like some of the other features. So some of the features may be capable of detecting the vehicle even when the classifier does not. Second, the classifier may correctly produce peaks for each car in the frame, but the weights of these peaks may be lower than the maximum peaks in the frame. In Fig. 7 the magnitude of the (classifier) peaks are not

incorporated in the evaluation.

VIII. Conclusions

Automatic tracking of objects in low frame rate WAMI is very challenging due to the sensor geometry, low temporal sampling rate, spatially varying optics, continually varying relative pose between the sensor and the scene, geometric occlusions, urban scene complexity, and scalable computing requirements to manage large data volumes. We have shown that a detect-and-track paradigm using a rich feature set of object appearance descriptors combined with feature fusion can be used for automatic tracking in low frame rate wide-area video. Fusing maximum feature likelihood maps in a Bayesian framework achieves the best results in terms of recall rate averaged across the length of the track. Combining information from motion estimation and vehicle classification decreases over recall rate for a number of reasons. In dense traffic background subtraction models reduce the successful detection of motion when similar vehicles replace each other at low temporal sampling rates. Using vehicle specific descriptors and detecting local peaks that may be of low probability enables the feature-only fusion mechanism to outperform general vehicle versus non-vehicle classifier performance in terms of recall rate within the search window. Future wide-area camera array imaging sensors will provide increased sampling rates up to 10 fr/sec, color and infrared channels with variable resolution to improve the detectability and continuous tracking of targets.

IX. Acknowledgments

This research was partially supported by grants from the Leonard Wood Institute (LWI 181223) in cooperation with the U.S. Army Research Laboratory (ARL) under Cooperative Agreement Number W911NF-07-2-0062, and the U.S. Air Force Research Laboratory (AFRL) under agreements FA8750-09-2-0198, FA8750-10-1-0182. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied of LWI, ARL, AFRL or the U.S. Government. This document has been cleared for public release under case number 88ABW-2010-2780. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

References

[1] A. Bovik, editor. *The Essential Guide to Video Processing*. Academic Press, Elsevier, 2nd edition, 2008.

[2] F. Bunyak and K. Palaniappan. Efficient segmentation using feature-based graph partitioning active contours. In *12th IEEE Int. Conf. Computer Vision*, pages 873–880, Kyoto, Japan, 2009.

[3] F. Bunyak, K. Palaniappan, S. K. Nath, and G. Seetharaman. Fux tensor constrained geodesic active contours with sensor fusion for persistent object tracking. *J. Multimedia*, 2(4):20–33, August 2007.

[4] R. Collins, Y. Liu, and M. Leordeanu. Online selection of discriminative tracking features. *IEEE Trans. Pattern Anal. Mach. Intel.*, 27(10):1631–1643, Oct. 2005.

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Conf. Comp. Vision Patt. Recog.*, volume 1, pages 886–893, 2005.

[6] A. Hafiane, K. Palaniappan, and G. Seetharaman. UAV-video registration using block-based features. In *IEEE Int. Geoscience and Remote Sensing Symposium*, volume II, pages 1104–1107, 2008.

[7] A. Hafiane, G. Seetharaman, K. Palaniappan, and B. Zavidovique. Rotationally invariant hashing of median patterns for texture classification. *Lecture Notes in Computer Science (ICIAR)*, 5112:619–619, 2008.

[8] S. Hinz, D. Lenhart, and J. Leitloff. Detection and tracking of vehicles in low framerate aerial image sequences. In *Proc. Workshop on High-Resolution Earth Imaging for Geo-Spatial Information*, page CD, Hannover, Germany, 2007.

[9] S. J. Julier, J. K. Uhlmann, J. Walters, R. Mittu, and K. Palaniappan. The challenge of scalable and distributed fusion of disparate sources of information. In *SPIE Defense and Security Symposium-Multisensor, Multisource Information Fusion*, volume 6242. 2006.

[10] J. Lewis. Fast normalized cross-correlation. In *Vision Interface*, volume 10, pages 120–123, 1995.

[11] S. Nath and K. Palaniappan. Adaptive robust structure tensors for orientation estimation and image segmentation. *Lecture Notes in Computer Science (ISVC)*, 3804:445–453, 2005.

[12] S. K. Nath and K. Palaniappan. Fast graph partitioning active contours for image segmentation using histograms. *EURASIP Journal on Image and Video Processing*, page 9p, 2009.

[13] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Patt. Anal. Mach. Intell.*, 24(7):971–987, 2002.

[14] K. Palaniappan, R. Rao, and G. Seetharaman. Wide-area persistent airborne video: Architecture and challenges. In B. Banhu, D. V. Ravishankar, A. K. Roy-Chowdhury, D. Terzopoulos, and H. Aghajan, editors, *Distributed Video Sensor Networks: Research Challenges and Future Directions*. Springer, 2010.

[15] K. Palaniappan, J. Uhlmann, and D. Li. Extensor based image interpolation. In *IEEE Int. Conf. Image Processing*, volume 2, pages 945–948, 2003.

[16] F. Porikli. Integral histogram: A fast way to extract histograms in cartesian spaces. In *IEEE Conf. Comp. Vision Pattern Recog.*, volume 1, page 829, 2005.

[17] D. Rosenbaum, F. Kurz, U. Thomas, S. Suri, and P. Reinartz. Towards automatic near real-time traffic monitoring with an airborne wide angle camera system. *European Transp. Res. Rev.*, 1(1):11–21, 2009.

[18] G. Seetharaman, G. Gasperas, and K. Palaniappan. A piecewise affine model for image registration in 3-D motion analysis. In *IEEE Int. Conf. Image Processing*, pages 561–564, 2000.

[19] V. Vapnik. *The nature of statistical learning theory*. Springer Verlag, 2000.

[20] X. Wang, T. X. Han, and S. Yan. An HOG-LBP human detector with partial occlusion handling. In *IEEE Int. Conf. Comp. Vision*, Kyoto, 2009.

[21] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM Computer Surveys*, 38(4):A13, 2006.

[22] Z. Yin, F. Porikli, and R. Collins. Likelihood map fusion for visual object tracking. In *IEEE Workshop Appl. Comput. Vis.*, pages 1–7, 2008.

[23] Z. Yue, D. Guarino, and R. Chellappa. Moving object verification in airborne video sequences. *IEEE Trans. Circuits and Systems for Video Technology*, 19(1):77–89, 2009.

[24] L. Zhou, C. Kambhamettu, D. Goldgof, K. Palaniappan, and A. F. Hasler. Tracking non-rigid motion and structure from 2D satellite cloud images without correspondences. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 23(11):1330–1336, Nov. 2001.