Establishing Object Correspondence Across Non-Overlapping Calibrated Cameras

Dileepa Joseph Jayamanne* and Ranga Rodrigo†
Department of Electronic and Telecommunication Engineering,
University of Moratuwa, Sri Lanka
Email: *dileepajj@gmail.com, †ranga@uom.lk

***Abstract*—When establishing object correspondence across non-overlapping cameras, the existing methods combine separate likelihoods of appearance and kinematic features in a Bayesian framework, constructing a joint likelihood to compute the probability of re-detection. A drawback of these methods is not having a proper approach to reduce the search space when localizing an object in a subsequent camera once the kinematic and appearance features are extracted in the current camera. In this work we introduce a novel methodology to condition the location of an object on its appearance and time, without assuming independence between appearance and kinematic features, in contrast to existing work. We characterize the linear movement of objects in the unobserved region with an additive Gaussian noise model. Assuming that the cameras are affine, we transform the noise model onto the image plane of subsequent cameras. We have tested our method with toy car experiments and real-world camera setups and found that the proposed noise model acts as a prior to improving re-detection. It constrains the search space in a subsequent camera, greatly improving the computational efficiency. Our method also has the potential to distinguish between objects similar in appearance, and recover correct labels when they move across cameras.**

I. Introduction

Establishing object correspondence across non-overlapping cameras is an active research field [1]–[8]. However, less attention is given for incorporating a predictive model to approximate when and where a particular object might appear in a subsequent camera, once the appearance and kinematic features of the object are detected in the current camera. The main benefit of such a model is the ability to reduce the search space when associating objects across cameras, and tracking the object of interest over a long distance where fields of view of the cameras only capture a part of the object's full trajectory.

Researchers have adopted different approaches in solving the non-overlapping correspondence problem. Constructing two separate likelihood functions of appearance and kinematic features and combining them in a Bayesian framework to compute the probability of re-detection [1]–[4] is the most common approach used. Object correspondence has also been made using kinematic (space-time) features alone [5], [6]. A drawback of these existing methods is not having a proper method to reduce the search space when looking for the same object in a subsequent camera.

978-1-4799-1740- 2/15/$31.00

A. Related Work

Majority of the research work conducted in object association constructs a joint likelihood by combining the appearance and kinematic likelihoods to achieve high object re-detection [1]–[4]. When generating the probability density function (PDF) for the kinematic likelihood, Shah *et al.* [2]–[4] applied Parzen windows non-parametric density estimation technique on their kinematic features to estimate the space-time PDF at the cost of heavy use of memory.

Different approaches can be considered when modeling kinematic constraints. Matei *et al.* [1] model the kinematic constraints using both road networks and transition time distributions. Huang *et al.* [6], in their system that tracks buses across non-overlapping cameras, only use transition time as their kinematic feature. They determine the transition time distribution and object correspondence using an expectation maximization framework. Sheikh *et al.* [5] model the motion of objects in the unobserved region between any two cameras by a polynomial kinetic model. However, these methods do not consider a proper approach to reduce the search space when searching for the same object in a subsequent camera.

When generating a PDF based on appearance features, one technique is to compute the modified Bhattacharyya distance on the color histograms of the objects appearing across cameras [2]. Another approach is to obtain the appearance similarity by computing brightness transfer function from a given camera to another [3], [4]. Similarly, different appearance features can also be used when generating a PDF based on appearance. Matei *et al.* [1] use color histograms and edge maps to compute an appearance likelihood for each type of appearance feature. When establishing object correspondence in a subsequent camera, these methods also lack a proper approach to reduce search space once the appearance and kinematic features are extracted in the current camera.

In order to resolve this, we characterize the motion of objects with a noise model, and use it together with the appearance likelihood to reduce the search space, and thereby the processing time when establishing correspondence. In our work, the novel approach proposed to condition the location of an object on its appearance and time allows us to incorporate the kinematic features into a noise model and model appearance features through a likelihood function. Consequently, main contributions of our work are

- proposing a novel method to condition the location of

objects on appearance and time,
- characterizing the linear movement of each object in the unobserved region with a noise model,
- justifying the assumption of affine cameras, and identifying the ideal moment to image the noise model onto the image plane through simulation,
- using our noise model as a prior distribution to increase the object re-detection ability, and to reduce search space when associating objects across cameras, and
- distinguishing between objects that are similar in appearance.

Next, we provide details about the novel approach.

II. Motion Noise Model and Its Imaging

The probability $P(.)$ of the location of an object conditioned on its appearance and time could be presented as follows:

$$P(L|A,t) \propto P(A|L,t) \times P(L|t) \tag{1}$$

where L is the location of the object at a given time t, and A is the appearance of the object. According to this approach the probability of locating an object conditioned on appearance and time is proportional to the probability of the appearance likelihood conditioned on location and time multiplied by the prior probability of locating an object conditioned on time.

A. Noise Model of an Object Traveling along a Linear Path

When an object is allowed to travel along a linear path it may deviate from its direction slightly from the path while moving forward, and it may undergo slight accelerations and decelerations frequently. We model this by considering that both position and velocity are corrupted by additive Gaussian noise. Thus, the 3-D position vector $\underline{X}'$ at time t_n of an object that was at $\underline{X}$ at time t_{n-1} is modeled as

$$\underline{X}' = \underline{X} + \sum_{t=t_0}^{t_n} \underline{n}_{pt} + \sum_{t=t_0}^{t_n} (\underline{v} + \underline{n}_{vt})\Delta t \tag{2}$$

$$\Delta t = t_n - t_{n-1} \tag{3}$$

where $\underline{n}_{pt}$ represents the Gaussian noise added to the position vector at time t, $\underline{v}$ is the average velocity of the object that travels along the path, and $\underline{n}_{vt}$ is the Gaussian noise added to model the velocity change at time t. Therefore by rearranging terms of Eq. 2 we get the following form:

$$\underline{X}' = X_t + \mathcal{N}(0, \Sigma_t) \tag{4}$$

$$X_t = \underline{X} + \sum_{t=t_0}^{t_n} \underline{v}\Delta t \tag{5}$$

$$\mathcal{N}(0, \Sigma_t) = \sum_{t=t_0}^{t_n} \underline{n}_{pt} + \sum_{t=t_0}^{t_n} \underline{n}_{vt}\Delta t \tag{6}$$

$$\underline{X}' = \mathcal{N}(X_t, \Sigma_t) \tag{7}$$

This noise model could be interpreted as locating an object at a given time t with mean value X_t and covariance matrix Σ_t.

B. Imaging of the Noise Model in Simulation

Based on the noise model, a 3-D point cloud can be generated using the mean value and the covariance matrix of the Gaussian at a given time. The next task is to effectively image the noise model for a given camera. This enables us to construct the posterior in the image plane. If we assume affine camera behavior, it is possible to apply direct transformation equations on the 3-D mean and the 3-D covariance matrix to obtain the respective imaging as follows:

$$E(x) = ME(X) + t, \tag{8}$$

$$\Sigma' = M\Sigma M^T, \tag{9}$$

where Σ is a 3×3 covariance matrix of the 3-D point cloud, Σ' is a 2×2 imaged covariance matrix, $E(X)$ is the expected value of the actual 3-D point cloud and $E(x)$ is the image of the expected value of the 3-D point cloud. M can be easily found by extracting the top left 2×3 matrix from the perspective camera matrix and t_1 and t_2 are given by the top rightmost column vector extracted from the same perspective camera matrix where the last element is scaled to one. Although Eq. 8 is a direct application of imaging a 3-D point from an affine camera [9], imaging the 3-D covariance of the noise model under affine camera assumption is one of our contributions resulting from this work.

C. Simulations to Verify Imaging of the Noise Model under Affine Camera Approximation

Although affine transformation is well known [9] in optical imaging, it is unclear if it is valid when imaging the 3-D covariance matrix of the noise model. We conduct a simulation to verify this. We generate a 3-D Gaussian point cloud with zero mean and a diagonal covariance matrix. We initially place a perspective camera 4 units away from the point cloud and rotate the camera around both x- and z-axis. At each instance we compute the image of the point cloud observed by the perspective camera and its covariance matrix. Then we directly image the 3-D covariance matrix of the point cloud using affine camera approximation and obtain its imaged covariance matrix. We compute the eigenvalues of the above covariance matrices and compare them for all the rotations applied to the camera. Next, we move the camera along the direction of the principal ray away from the point cloud and repeat the same procedure for different camera positions. We observed that the eigenvalues computed using both methods become similar as the average distance to the point cloud increases.

Fig. 1 shows how the camera is rotated about x-axis and moved away from the point cloud along the principal ray of the camera. Fig. 2 shows how the first eigenvalue of the covariance matrix imaged under affine approximation converges to the perspective imaging when the camera is rotated about the x-axis with different average distances. We also obtained similar results for the second eigenvalue as well as for eigenvalues obtained when the camera is rotated about the z-axis with different average distances. Therefore, when the camera is far

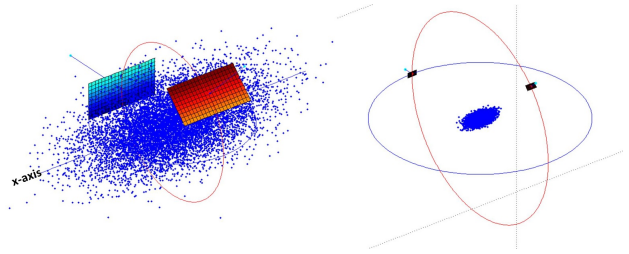

Fig. 1. Imaging the covariance matrix of the point cloud from the perspective camera and its affine approximation. The planes show the camera image planes, and the dots represent the randomly generated points with a standard deviation along the x-axis equal to 2 units. Left: The camera is rotated about the x-axis with an average distance of 4 units. Right: The camera is moved along the principal ray and rotated about the x-axis with an average distance of 24 units. This simulation confirms that when the camera is approximately 12 times further than the standard deviation, imaging the 3-D covariance matrix under the affine approximation is accurate. Best viewed in color.

away from the point cloud, the affine approximation converges to the perspective imaging. In other words when the scene depth is very small compared to the average distance and the point cloud is close to the principal ray of the camera, the covariance matrix of the point cloud can be directly imaged using Eq. 9.

III. Methodology

In order to achieve geometrically constrained object tracking in non-overlapping cameras, our system models the probability of the re-detection location in a subsequent camera constrained on the appearance of the object and time of re-detection. We characterize the linear movement of objects in the unobserved region with an additive Gaussian noise model. We use this as a prior and model the appearance likelihood of objects across cameras by appearance features. The resulting posterior allows us to constrain the search space for re-detection, thus greatly improving the computational efficiency. Fig. 3 depicts our methodology.

Our system comprises several modules: background subtraction, local object tracking within the camera, appearance feature extraction per object, speed estimation, initializing 3-D noise model, transforming the noise model onto the subsequent camera, and object color histogram matching. We use a color based technique when subtracting the background to estimate the foreground region that corresponds to the motion of objects [10]. We track objects within the field of view of the camera from point of entry to exit. We extract a color histogram per object. We identify the time of entry, the time of exit and the distance traveled within the field of view by an object to compute its average speed. When an object is about to enter into the unobserved region, we initiate its noise model to characterize its motion. Then we transform its noise model onto the image plane and apply histogram matching to establish object correspondence across non-overlapping cameras. Next, we provide details of our appearance likelihood.

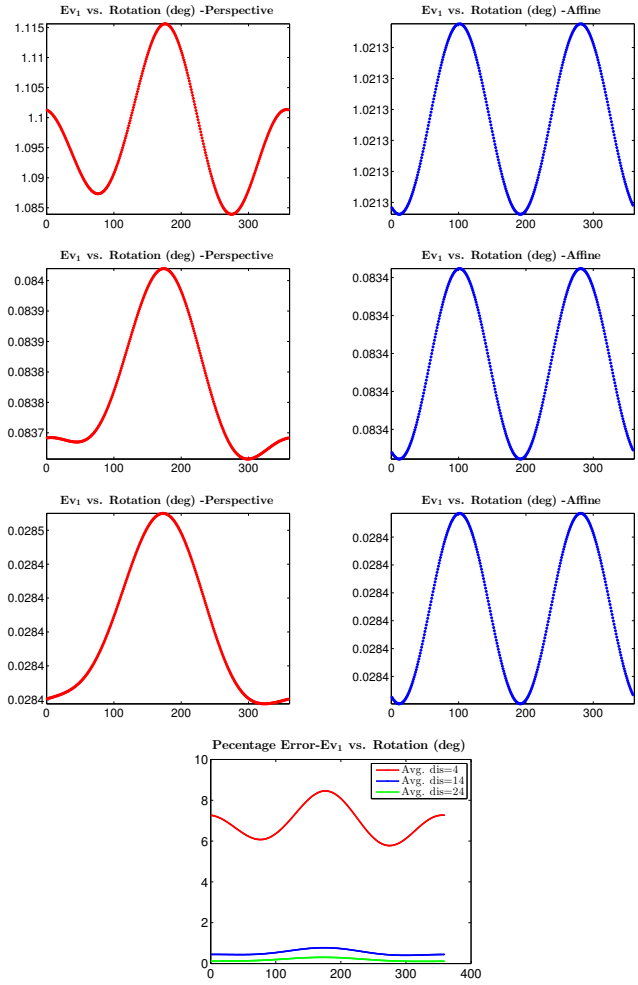

Fig. 2. Comparison of Ev_1 of the covariance matrix obtained after imaging the point cloud by perspective and affine approximation with x-axis rotation. In row 1, 2 and 3, camera is placed with an average distance of 4, 14, and 24 units away from the point cloud respectively. Best viewed in color.

A. Appearance Likelihood Function

When establishing object correspondence across non-overlapping cameras, as proposed in Section II, conditioning the location of an object on its appearance and time requires us to construct an appearance likelihood conditioned on location and time. We use color histograms [2] to model the appearance likelihood of objects detected by non-overlapping cameras. We compute the modified Bhattacharyya distance as a similarity measure of two color histograms. The modified Bhattacharyya distance between two histograms k and q of n bins is given by

$$D(k,q) = \sqrt{1 - \sum_{i=1}^{n} \sqrt{k_i q_i}}. \tag{10}$$

We parameterize the appearance likelihood by fitting a Gaussian distribution on the histogram obtained by computing the modified Bhattacharyya distances on average color histograms of matching objects detected by two non-overlapping cameras. We condition the appearance likelihood on location

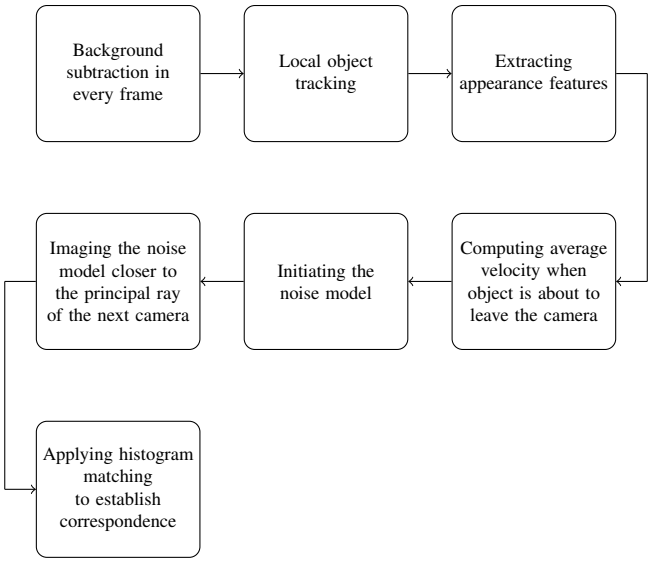

Fig. 3. Block diagram of our method used in establishing object correspondence across non-overlapping cameras.

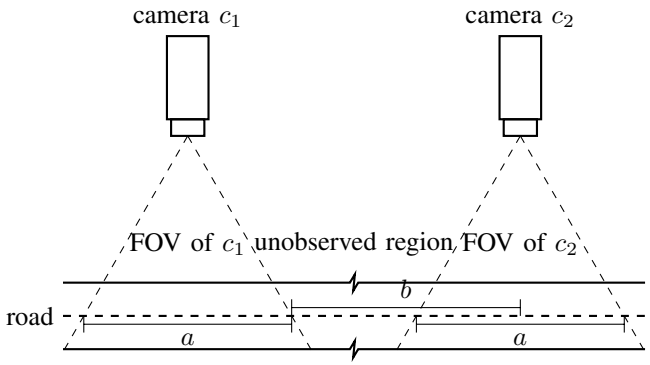

Fig. 4. The two cameras are related via a translation. Fields of view of the two cameras equal a, and b is the distance to the world origin.

and time by extracting the color histograms at a particular location at a particular time per object. Since the noise model characterizes the linear movement of objects in the unobserved region and prevents processing frames of a subsequent camera during the time when object travels in the unobserved region, efficient object re-detection could be achieved by combining it with histogram matching.

B. Validating the Noise Model on Experimental Data

We designed an experiment (Exp. 1) to test our approach described in Section II in an indoor environment. We mounted two web-cams with an unobserved region focusing on the side view of the objects. We connected the two web-cams to the same PC and extracted frames with the time-stamp. Then we drove a remote controlled car along the path. Fig. 4 shows the camera setup. The parameters of the camera setup are given in Table I.

We ran two local trackers on the set of frames obtained by the two non-overlapping cameras. Then we extracted color histogram per object in the first camera. Once the point of entry and the point of exit of an object is identified, we computed its average speed. Then we immediately initiated its noise model and imaged the noise model in the second camera close to the principal ray, and identified the object blob to which it is related. Once the color histogram per object is extracted in the second camera, we computed color histogram similarity to recover the correct object label previously assigned. We used a checker board pattern as the calibration object and used Matlab camera calibration tool box [11] to calibrate the non-overlapping cameras.

TABLE I
PARAMETERS OF CAMERA SETUPS USED IN EXPERIMENTS

Experiment	View	a	b
Exp. 1 (toy)	Side view	520 mm	1590 mm
Exp. 2 (toy)	Side view	840 mm	2095 mm
Exp. 3 (real)	Side view	6300 mm	12000 mm

We designed another experiment (Exp. 2) where we drove two similar-color, similar-type toy cars and attempted distinguishing them. We used small toy cars so that two or more would fit entirely in the FOVs of the non-overlapping cameras. Then we conducted the same experiment as described in Section III-B in outdoors to validate our approach described in Section II on real data (Exp. 3). The results are discussed in the Section IV.

C. Local Tracker

We use an appearance based local tracker with background subtraction for object tracking. Although background could be modeled using different techniques [12]–[17], in this work we stick to very basic background modeling since we do not deal with situations where there is intense object traffic. Therefore, we use the most recent object-less frame as the background corresponding to the current frame in consideration. Any object-less frame can be defined as a frame where no foreground region is found. We filter out the foreground region using a percentage color change compared to the background and apply background subtraction. However, when we applied our method on a highway scenario, we modeled the background using a background model [13]. Details of this highway application are given in Section IV.

We detect SURF [18] points only on the extracted foreground region and applied a feature validation technique to filter features detected in the background region. Once the features are detected, we match those in the next frame and the corresponding background frame to validate. We consider a feature point as valid feature point if it can be matched once in the next frame and does not have a match in the corresponding background frame. Feature validation not only help us to validate features but also to validate estimated foreground region thereby eliminating the false estimations. Although different feature clustering algorithms are available [19], [20], we cluster the validated SURF points based on the number of foreground object blobs. Then we track the clusters based on proximity and cluster trajectory.

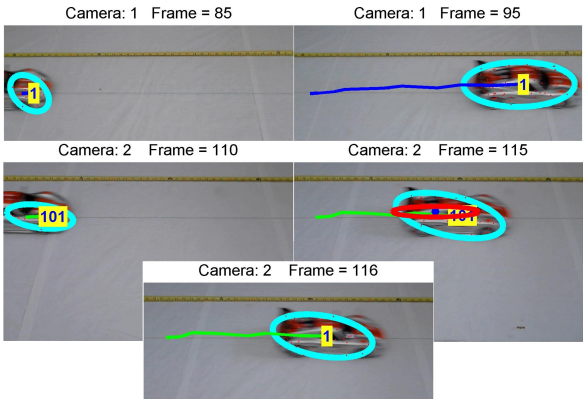

Fig. 5. Row 1: Car enters into the FOV of camera 1 and leaves the camera. Row 2: The red ellipse with a blue dot corresponds to the entire image of the noise model. The major and minor axes of red ellipse represent the square root of eigenvalues of the imaged 3-D covariance matrix of the noise model. Imaging of the noise model falls on the actual object indicating that the noise prior is correct. Best viewed in color.

TABLE II
VALIDATING THE PROPOSED NOISE MODEL ON TOY DATA.
A: APPEARANCE P: PRIOR

Model	# of instances	Av. processing time	Variance
A	20	64.4 s	0.089
A+P	20	33.4 s	0.071

TABLE III
RESULTS OF EXP 3 AND EXP 2. A: APPEARANCE P: PRIOR

Experiment	Model	# of objects	Correct matching	Re-dec%
Noise model	A	25	20	80%
on real data	A+P	25	23	**92%**
Distinguishing	A	25	18	72%
similar objects	A+P	25	23	**92%**

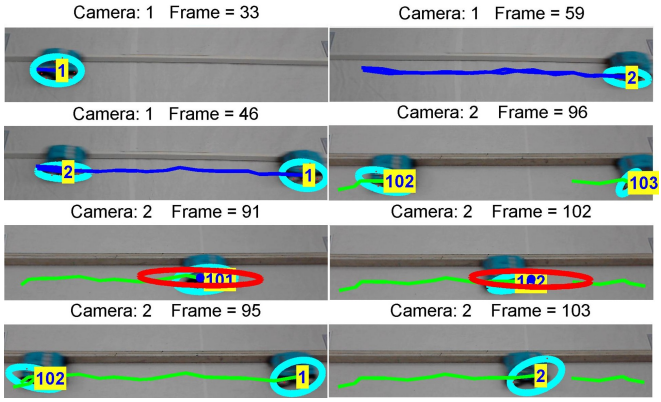

Fig. 6. Object correspondence is established after imaging the noise model (red ellipse with a blue dot) and object color histograms matching. Objects of similar type and color have been distinguished. Best viewed in color.

IV. RESULTS AND ANALYSIS

In summary, our system achieves geometrically constrained object tracking in a subsequent non-overlapping camera by constraining the probability of the re-detection location of an object on its appearance and its time of re-detection. We characterize the linear motion of each object with an additive Gaussian noise model. We use affine camera assumption to transform our noise model directly onto the image plane of a subsequent camera. We describe this mapping in Section II-B and validate it through simulation. Then, we combine the appearance likelihood with the noise prior to establish object correspondence across non-overlapping cameras. In this Section, we describe the results.

Fig. 5 shows an instance of re-detecting a toy car across non-overlapping web-cams obtained after applying our method. This also shows object tracking in each camera and imaging of the entire noise model as mentioned in Section II-B. Once the noise model is imaged, we use color histogram matching to establish object correspondence. Table II shows the improvement in offline processing time achieved when appearance and noise models are used compared to applying histogram matching alone for an average tracking instance.

Table III shows results obtained after applying our method on real and toy data (Exp. 3 & Exp. 2). Processing frames acquired by the next camera is reduced by initializing a local tracker for each object that leaves the current camera. This increases the computational efficiency of searching for the object in the next camera by preventing the processing of unwanted frames acquired during the time where object travels in the unobserved region. Fig. 6 depicts the resulting images of distinguishing between objects that are similar in appearance.

When the noise model is combined with the appearance likelihood compared to re-detecting objects based on appearance alone, object re-detection rate across non-overlapping cameras is increased. This is shown in Table III. When methods based on appearance alone is used, the search space is not reduced, and re-detection rate drops particularly when objects of similar nature move across cameras.

We also used our noise model as a predictor, and applied our framework to establish correspondence across non-overlapping cameras in a highway scenario. The cameras were mounted to perceive the frontal view of approaching objects. The camera gap was more than 10 km. Using the covariance matrix of the noise model, we identified the frame range to process in the next camera when trying to find a match for every object that left the current camera. On average, this method only used 15% of the entire search space per object for SURF feature matching with 81% accuracy.

V. CONCLUSION AND FUTURE WORK

In this work we introduced a novel methodology to condition the location of an object on its appearance and its time of re-detection when establishing correspondence across non-overlapping cameras. We obtained good results by testing our approach on different web-cam experiments and real camera setups. When establishing correspondence, the average time taken to process a vehicle by the combined appearance likelihood and noise prior is approximately 50% less than the average time taken by the appearance model alone. Our noise model provides a computationally efficient way to search for

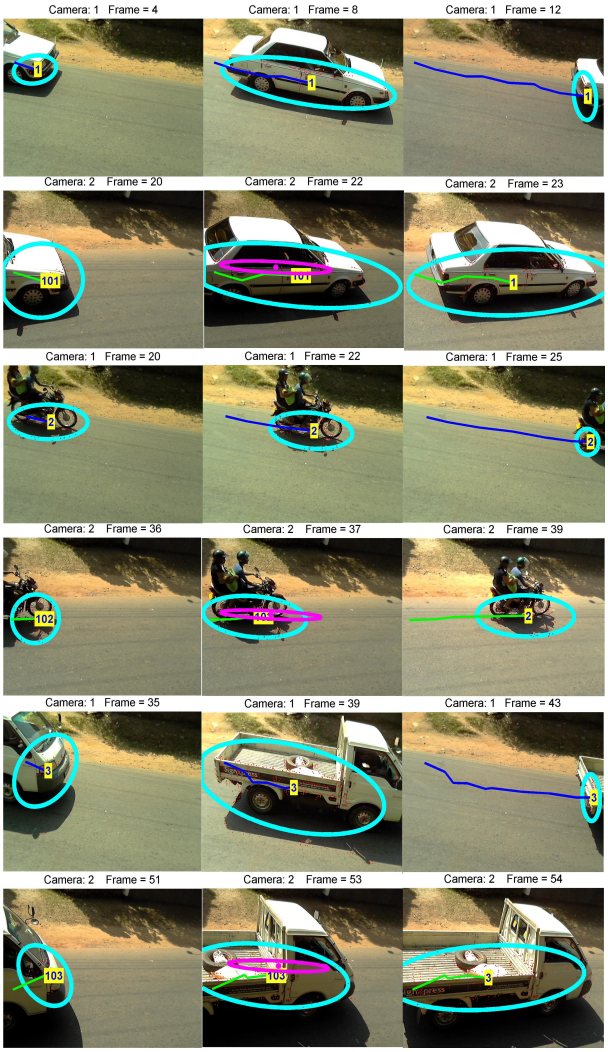

Fig. 7. Odd and Even rows: Vehicles tracked in cameras 1 and 2. After imaging the entire noise model (magenta ellipse with a magenta dot), color histogram matching is used to establish correspondence between objects across non-overlapping cameras. Since the noise model acts as a prior that constrains the search space, conditioning the location of an object on its appearance and time in a subsequent non-overlapping camera improves its re-detection. Best viewed in color.

the re-appearance location of an object in a subsequent camera. Also, this method has the potential to distinguish between similar color similar-type objects across non-overlapping cameras where methods based on appearance alone fail.

In this work, we added 5% of object speed as the standard deviation along the object's traveling direction when modeling the noise. Learning the parameters of the noise model and generalizing the imaging of the noise model to handle perspective cameras are left as future work.

ACKNOWLEDGMENT

The authors would like to thank the National Science Foundation of Sri Lanka Grant No: RG/2012/CSIT/01 for funding this work.

REFERENCES

[1] B. C. Matei *et al.*, "Vehicle tracking across nonoverlapping cameras using joint kinematic and appearance features," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Providence, RI, 2011, pp. 3465–3472.

[2] O. Javed *et al.*, "Tracking across multiple cameras with disjoint views," in *Proc. 9th IEEE Conf. Computer Vision*, Nice, France, 2003, pp. 952–957.

[3] ——, "Appearance modeling for tracking in multiple non-overlapping cameras," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, San Diego, CA, 2005, pp. 26–33.

[4] O. Javed, Z. Rasheed, K. Shafique, and M. Shah, "Modeling inter-camera space–time and appearance relationships for tracking across non-overlapping views," *Comput. Vis. Image Understanding*, vol. 109, no. 2, pp. 146–162, Feb. 2008.

[5] Y. Sheikh *et al.*, "Trajectory association across non-overlapping moving cameras in planar scenes," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Minneapolis, MN, 2007, pp. 1–7.

[6] C. C. Huang *et al.*, "Probabilistic modeling of dynamic traffic flow across non-overlapping camera views," in *Proc. 20th IEEE Conf. Pattern Recognition*, Istanbul, Turkey, 2010, pp. 3332–3335.

[7] V. Kettnaker and R. Zabih, "Bayesian multi-camera surveillance," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Fort Collins, CO, 1999, pp. 117–123.

[8] C. Stauffer and K. Tieu, "Automated multi-camera planar tracking correspondence modeling," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Madison, WI, 2003, pp. I–259–I–266.

[9] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. New York, NY: Cambridge University Press, 2003.

[10] D. J. Jayamanne *et al.*, "Appearance based tracking with background subtraction," in *Proc. 8th Int. Conf. Computer Science and Education*, Colombo, Sri Lanka, 2013, pp. 643–649.

[11] J. Y. Bouguet. (2014, Apr.) Camera calibration toolbox for matlab. [Online]. Available: http://www.vision.caltech.edu/bouguetj/calib_doc/

[12] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Fort Collins, CO, 1999.

[13] Z. Kim, "Real time object tracking based on dynamic feature grouping with background subtraction," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Anchorage, Alaska, 2008, pp. 1–8.

[14] Y. Sheikth and M. Shah, "Bayesian modeling of dynamic scenes for object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 11, pp. 1778–1792, Nov. 2005.

[15] S. Brutzer *et al.*, "Evaluation of background subtraction techniques for video surveillance," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Colorado Springs, CO, 2011, pp. 1937–1944.

[16] M. Piccardi, "Background subtraction techniques: a review," in *Proc. IEEE Conf. Systems, Man and Cybernetics*, Hague, Netherlands, 2004, pp. 3099–3104.

[17] Y. Benezeth *et al.*, "Review and evaluation of commonly-implemented background subtraction algorithms," in *19th Int. Conf. Pattern Recognition*, Tampa, FL, 2008, pp. 1–4.

[18] H. Bay, A. Ess, and L. V. Gool, "Speeded up robust features," *Comput. Vis. Image Understanding*, vol. 110, no. 3, pp. 346–359, Oct. 2008.

[19] J. Shi and J. Malik, "Normalized cuts and image segmentation," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, 1997, pp. 731–737.

[20] R. Xu and D. C. Wunsch, "Survey of clustering algorithms," *IEEE Trans. Neural Netw.*, vol. 16, no. 3, pp. 645–678, May 2005.