Motion and Appearance Contexts for Tracking and Re-Acquiring Targets in Aerial Videos

Saad Ali
Computer Vision Lab
University of Central Florida
sali@cs.ucf.edu

Vladimir Reilly
Computer Vision Lab
University of Central Florida
vsreilly@cs.ucf.edu

Mubarak Shah
Computer Vision Lab
University of Central Florida
shah@cs.ucf.edu

Abstract

In this paper we use Motion and Appearance Contexts *for persistent tracking of objects in aerial imagery. The motion context in a given environment is a collection of trajectories of objects which are representative of the motion of the occluded or unobserved object. It is learned using a clustering scheme based on the Lyapunov Characteristic Exponent (LCE) which measures the mean exponential rate of divergence of the nearby trajectories. The learned motion context is then used in a regression framework to predict the location of the unobserved object. The appearance context of an occluded (target) object consists of appearance information of objects which are currently occluded or unobserved. It is incorporated by learning a distribution of interclass variation for each target-unobservable object pair. In addition, intra-class variation distribution is constructed for each occluded object using all of its previous observations. Qualitative and quantitative results are reported on challenging aerial sequences.*

1. Introduction

This paper attempts to solve the problem of persistent tracking and reacquiring of objects in aerial videos. Persistent tracking implies the ability to track objects through occlusions, while reacquiring implies the ability to maintain the correct labels of objects when they leave and come back into the field of view. Persistent tracking and reacquiring is a challenging problem due to number of a reasons: i) the motion of the platform on which camera is mounted is unconstrained, ii) targets of interest move independently with respect to the motion of the camera, iii) the field of view of the camera is restricted, iv) appearances and shapes of the objects change due to illumination and pose variation, and v) terrain features act as sources of occlusion resulting in loss of observations of target objects.

The main thrust of our proposed idea is to exploit the contextual knowledge present in the environment. For the aerial imagery this will consist of the information which is necessary for interpreting the event that is taking place on the ground. We employ two pieces of this contextual knowledge, namely *Motion Context* and *Appearance Context* for tracking and reacquiring target objects.

The *Motion Context* captures an intuitive observation that *the locomotive behavior of an object (e.g. car) provides information about locomotive behaviors of nearby objects (cars) that are in the same environment.* This is true because cars moving along the same stretch of a road are often subjected to similar constraints in terms of the paths that they can take, shape of the path, road conditions, and speed restrictions. Therefore, the motion of a car contains the information which can be used to interpret how neighboring cars would behave. The *Appearance Context* captures the notion that *for an occluded object, a more discriminative appearance model can be constructed if we have the knowledge about which other objects are currently occluded or unobserved*. Building an appearance model that takes into account the appearance of other unobservable objects will make the model more discriminative, thus making it easier to establish the correspondence of the appearance of the newly detected object with the appearances of cars in the unobservable set. Figure 1 provides a pictorial description of the motion and the appearance context.

2. Overview

The motion context is implemented in a regression framework which is inspired by the research conducted in the field of oceanography for search and rescue operations at sea [1][2][3]. The goal in such a scenario is to narrow the search area based on the best possible prediction of the lost object, given its initial position and the motion of other objects. We have mapped this setting to the scenario of aerial videos by defining the occluded or unobservable object on the ground as the one for which we want to make the prediction and we refer to it as the 'predictand'. Tracks from objects which are observable and are moving along or have moved along the same path in the past as predictand, are used to compute the likely location of the 'predictand' and

1-4244-1180-7/07/$25.00 ©2007 IEEE

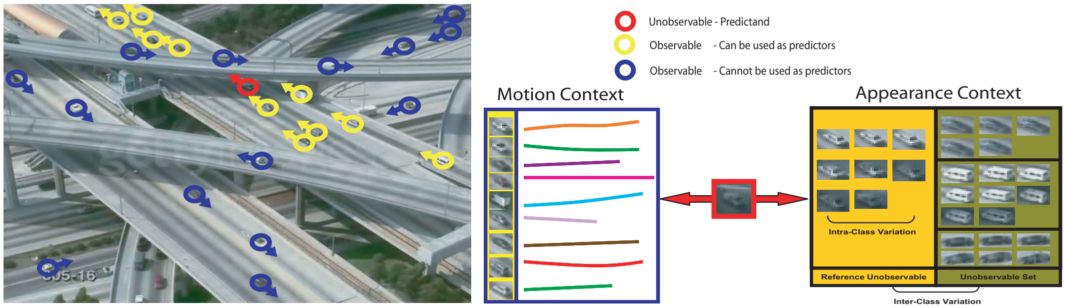

Figure 1. An illustration of the the *Motion and Appearance Context*. The motion context of a car circled in red is defined by the cars circled in yellow that have motion dynamics similar to the red car. Cars circled in blue are not part of the motion context of red car, since blue cars have motion dynamics which are different from the red car. The appearance context of the red car consists of cars which are currently unobservable and are shown in the green rectangle on the far right. It shows us that the appearance model of the red car only has to be discriminative with respect to this set, since the red car has to compete only with these cars at the re-acquiring stage.

we refer to them as 'predictors'.

Formally, let $r_1(0), r_2(0), ..., r_N(0)$ be the starting positions in the image plane of N objects O=$\{O_i\}_{i=1}^{N}$ in the image plane at $t = 0$. Corresponding to each O_i we have a set $C_i \subset O$ defining its motion context. The cardinality M of C_i is less than or equal to N. Assume that the trajectories of the first $p = N - 1$ objects $r_1(t), r_2(t), ..., r_p(t)$ are observed during the time interval $(0, T)$, while the trajectory of the last object, $r_N(t)$, is not observed. Now, the problem is to make a prediction about the location of the unobserved object O_N, given trajectories of predictors in C_N and the initial predictand position $r_N(0)$. The optimal prediction in the mean square sense [1] is $E \mid \widehat{r}_N(T) - r_N(T) \mid \mapsto min$, where $\widehat{r}_N$ is given by the conditional expectation $\widehat{r}_N(T) = E(r_N(T) \mid r_1(t), r_2(t), ..., r_p(t), 0 \le t \le T)$, based on all the observations. However, this expectation is hard to find explicitly as observed by Piterbarg et. al. [1]. To overcome this problem a regression framework is used which estimates the future location of the predictand by employing the data from the predictors in a least square sense. For the clarity purposes the above formulation assumes that all p objects are in the set C_N.

For implementing the appearance context we keep a set $U \subset O$ of the currently unobservable objects at all times t. The appearance of each object O_i undergoing occlusion is encoded in terms of the intra-class variation with respect to all of the observations of O_i, and pairwise interclass variation with respect to observations of objects in U. Let the cardinality of U be l, then for each O_i we have $(l + 1)$ sets of intra and interclass variation vectors. The variation vectors in each set are assumed to be drawn from a separate Gaussian pdf where means and variances of these pdf's are computed using the corresponding variation vectors. Now, in order to establish whether a newly detected blob B is actually O_i, we construct intra-class and inter-class variation vectors of B are with respect to previous observations of O_i and observation of objects in U, respectively. Similarity is computed by looking at how well the intra and inter-class variation vectors of B are described by the pdf's of inter and intra-class variation of O_i.

3. Related Work

Over the last few years a number of algorithms have been proposed for linking tracks in a multiple camera setup [5][6][8][7]. Huang et. al. [5] employed a probabilistic approach to model appearances and transition times for tracking cars across cameras, while Kettnaker et. al. [6] used positions, velocities, and transition times in a Bayesian framework for this purpose. Similarly, Javed et. al. [8] and Glibert et. al [7] proposed a supervised and an unsupervised framework for learning camera topologies, respectively. However in the setting of aerial videos topographic, appearance and learning based constraints used by aforementioned approaches are not easily extendable.

Recently, Amitha et. al. [9] proposed a framework for linking tracks across occlusions in the setting of aerial imagery. This work builds upon their previous work [10]. In [9], the problem is solved in two stages where at the first stage pairwise associations between tracks were established using temporal ordering, proximity of predicted location, and matching of appearance templates. While the second stage is used to handle the splitting and merging of objects. The main difference between their approach and the method proposed in this paper is that they do not employ any contextual knowledge of the object's kinematics or appearance at the association stage. The state of the art methods for modelling appearances of cars in the setting of aerial imagery use representations based on edges [11][12], line features [13], and shape features [14]. However, none of these approaches exploit the contextual knowledge available in the scene.

4. Framework

In this section, we will present theoretical and implementation details of the proposed framework.

4.1. Modelling Motion Context

Let N be the total number of objects observed up until time T, which are represented by the set O. Let $V_T \subseteq O$

be the objects which are visible in the current frame of the video. $P \subset O$ is the set of objects whose locations are being predicted. Corresponding to each $O_i \in P$ we have a set $C_i \subset O$ of objects which act as the predictors for O_i. Lastly, we maintain a set of trajectories R, where $r_i(0, ..., T)$ is the trajectory corresponding to object O_i. Our goal is to predict the next location $r_i(t)$ of object $O_i \in P$ at time t where $t > T$, given its last location $r_i(T)$. Note that $r_i(t)$ is a vector consisting of the image location $[x_i(t), y_i(t)]$. The starting and current locations $r_j(T)$ of all M predictors in C_i are also known. Given this information the current location of the object $O_i \in P$ will be predicted by using the following regression: $r_i(t) = A(t)r_i(T) + b(t) + z_i(t)$, where $A(t)$ is an unknown 2x2 matrices and $b(t)$ is an unknown 2-dimensional vector. While $z_i(t)$ is a stochastic process with a zero mean uncorrelated for a fixed T.

The unknown matrix $A(t)$ and vector $b(t)$ for O_i are computed using the initial and the current locations of predictors in the set C_i. There are six unknown parameters, i.e., four entries of matrix A and two entries of vector b. Therefore the constraint that we always need to have at least three or more predictors in the set C_i to solve the system [1]. In case we have less than three predictors, only appearance is used for re-acquisition.

The least square estimates of $A(t)$ and $b(t)$ are obtained as $\widehat{A}(t) = S(t)S^{-1}(T)$ and $\widehat{b}(t) = m(t) - \widehat{A}(t) - m(T)$, where $m(t) = \frac{1}{p}\sum_{i=1}^{p} r_i(t)$ is the center of mass of the predictor cluster. The value of $S(t)$ is calculated using the relation $S(t) = \sum_{i=1}^{p}(r_i(t) - m(t))(r_i(0) - m(0))^{'}$. Finally, the obtained estimator is used to predict the unobservable object O_i using $\widehat{r}_i(t) = m(t) + S(t)S(0)^{-1}(r_i(0) - m(0))$. The important step in this formulation is the calculation of set C_i corresponding to each O_i. Knowing this set, finding solutions for the above equations is straight forward.

4.2. Selecting Predictors

Predictors for each unobservable object are selected using a methodology based on the concept of LCE. We will now briefly describe the LCE before presenting the algorithm for the selection of predictors.

4.2.1 Lyapunov Characteristic Exponent

LCE is used as a tool for measuring the chaoticity in the dynamical systems [4]. It does this by measuring the rate of exponential divergence between the neighboring trajectories in the phase space. For a given dynamical system $\dot{x} = f(x)$, the maximum LCE is defined as $\gamma = lim_{t\rightarrow\infty}\chi(t)$, with $\chi(t) = \frac{1}{t}ln\frac{|\xi(t)|}{|\xi_0|}$, where $\xi(t)$ is the solution of the differential equation of the system while $\xi(0)$ is the initial state of the system. The values $\chi(t)$ are called the LCE, and for practical calculation it is not possible to take the limit up to infinity. Therefore, we follow its evolution up to some prespecified number of steps. In most cases we do not have the differential equations governing the temporal evolution of the dynamical system. This is also true for the situation that we are dealing with in this paper where the motion of objects in the scene are governed by a set of unknown differential equations. To overcome this problem we use an alternative approach to computing LCE first proposed by Wolf et. al. [16] where the definition of LCE is replaced by $\chi(t) = \frac{1}{N}ln\frac{d_t}{d_0}$, where d_t is the distance between two trajectories at time t, initially separated by d_0.

4.2.2 Predictor Selection

Using LCE to compute predictor set C_i for a given predictand $O_i \in P$ requires some special considerations. First the 'predictand trajectory' $r_i(t)$ is labelled as the reference trajectory and the locations of all the other trajectories in R are computed relative to the reference trajectory. Pairwise LCE is computed between the reference trajectory and the remaining trajectories, one at a time using Equation 1.

In order to be able to correctly predict the position $r_i(t)$ of the unobserved object O_i at time t, we have to select a set of meaningful predictors C_i to be used in the regression framework. The term 'meaningful predictor' is used to emphasize the object that *is following* or *has followed* a track similar to that of the predictand between some time interval $t_c = (t_s, t_e)$ where $0 < t_s < t_e$ and $t_s < t_e < T$. This implies that a section $r_j(t_s^j, t_e^j)$ of trajectory $r_j(t) \in R$ during some time $t_c^j = (t_s^j, t_e^j)$ is similar to the track $r_i(t)$ of the predictand in the last $F = (t_e^j - t_s^j)$ frames of the sequence, where F is a parameter defining the observation window. Note that superscript j in t_s^j and t_e^j is referring to the fact that there is a separate start and end time corresponding to each potential predictor track $r_j(t)$. However, the value of F will remain the same. In our case the measure of similarity is the LCE, thus in order to assemble a set of predictors for a particular predictand, we search all the tracks in R for a section $r_j(t_s^j, t_e^j)$, where the motion of the object is most similar to that of the predictand.

First, we define track section Z of $O_i \in P$ as $Z = r_i(t - F, t)$, where t is the current time. Second, we perform the following search. For every possible section $t_c^j = (t_s^j, t_e^j)$ of potential predictor's track $r_j(t)$, we extract a section $Z_c = r_j(t_s^j, t_e^j)$. Third, we normalize the track section Z_c with

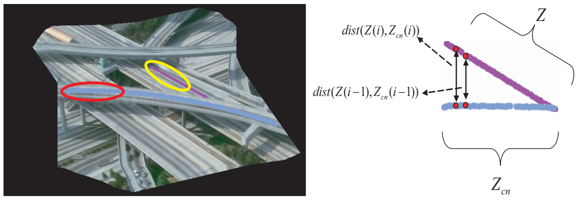

Figure 2. A portion of a potential predictor trajectory shown within a red ellipse is first normalized with respect to the predictand trajectory shown within a yellow ellipse. The Euclidian distance between points of the trajectory at each time instant is used in Equation 1 to compute the LCE.

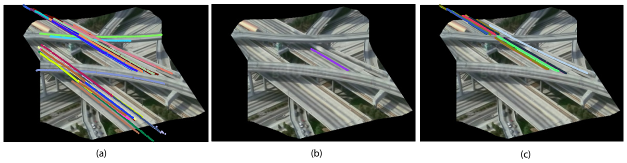

Figure 3. Illustrates the predictor selection procedure. (a) The set of trajectories that have been observed so far in this scene. This set contains trajectories generated from observed objects and trajectories that have been predicted in the past. (b) Shows the predictand trajectory. (c) Predictor selection result returned by our selection procedure emphasizing the objects whose motion dynamics are similar to the predictand trajectory.

respect to the predictand's track section Z, so that they both start at the same point. Let us call the normalized track Z_{cn}. Finally, we compute the LCE between predictand's and object's track sections Z and Z_{cn} as

$$\gamma(Z, Z_{cn}) = \frac{1}{F}\sum_{i=2}^{F} ln \frac{dist(Z(i), Z_{cn}(i))}{dist(Z(i-1), Z_{cn}(i-1))}, \qquad (1)$$

where, in our definition, $dist(Z(i), Z_{cn}(i))$ is the Euclidian distance between two track points $(x(i), y(i))$ and $(x_{cn}(i), y_{cn}(i))$ given by $dist = \sqrt{(x(i) - x_{cn}(i))^2 + (y(i) - y_{cn}(i))^2}$.

If $\gamma(Z, Z_{cn}) \leq \alpha$, where α is a threshold derived empirically, we add object O_j to the set of predictors C_i of predictand O_i. We also add t_e^j to a new set B_i where the cardinalities of B_i and C_i are the same. Note that the closer the value of γ is to 0, the more similar the motion of predictor O_j between frames t_s^j and t_e^j is to the motion of the predictand between frames $t - F$ and t. The process of computing a LCE between the reference trajectory and a potential predictor trajectory is summarized in Figure 2.

Once an object has been added to the predictor set C_i of predictand O_i, we keep the same predictor in set C_i until predictor's and predictand's tracks begin to diverge. Thus, for all subsequent frames we call the above procedure to ensure that the motion of predictor $O_j \in C_i$ between frames $t_s^j + p$ and $t_e^j + p$, is similar to the motion of the predictand O_i between frames $t - F + p$ and $t + p$, where p is the frame counter. If the tracks of the predictand and the predictor begin to diverge, we remove the predictor from the set. Figure 3 shows one particular result where, for the reference trajectory shown in Figure 3(b), we were able to construct the corresponding predictor set shown in Figure 3(c) from the input tracks displayed in Figure 3(a).

4.3. Modelling Appearance Context

The appearance model of each object O_i is constructed using the contextual knowledge about which objects are in set P. All of the observations of O_i are defined by a set $X_i = (\mathbf{x}_n \mid n = 1, ...L)$, where L is the total number of observations. For each O_i we define a variable y, which is a function $y = f(x, x^i)$ of a pair of observations $x \in X$ and $x^i \in X$. We define a matrix Λ_i, which contains the values of y for all possible pairs of observations in X_i. The matrix Λ_i is then the matrix of intra-class variation for object O_i. Similarly, for object O_i the inter-class variation data is computed separately for each $O_j \in P$. Let us define a matrix of inter-class variation vectors between O_i and $O_j \in P$ by Ω_i^j. Note that $\Omega' s$ use the contextual knowledge by considering inter-class variation with respect to only those $O' s$ that are currently unobservable, thus providing an extra piece of information that is going to help in reacquiring O_i after an occlusion. In our implementation, we define function $f(x, x^{'})$ as $f(x, x^{'}) = x - x^{'}$. The input observation x_n for each object consists of RGB pixel values of the object region.

```
Target Reacquiring -
  • for each incoming frame $f$ get $V_f$ .
      – for each new object $O_k$ in $V_f$
          ∗ find if $O_k$ is in $P$ by computing distance between current
            location of $O_k$ and last predicted location of $P_i$ that is
            within $(f - u)$ frames.
          ∗ If $distance < thresh$, set $O_k = P_i$.
          ∗ Remove $P_i$ from $P$.
      – for each old object $O_l$ in $V_f$
          ∗ Update the predictor set $C_i$ by calling
            PredictorSelection$(r_l(0, ..., f), C_i, B_i, G)$
----------------------------------------------------------------
returns $C$:$\mathbf{PredictorSelection(R(0...., f), C, B, G, O)}$
  • for each track $r_i(0, .., f - 1)$ in $G$ except the track $R(0, ..., f - 1)$
      – if $O_i$ not member of $C$ then for each time instance $j = F$ to
        $(f - 1)$
          ∗ $Z = R(f - F, f)$ , $Z_c = r_i(j - F, j)$
          ∗ $Z_{cn} = normalize(Z, Z_c)$
          ∗ Compute $\gamma$ by using Equation 1
          ∗ if $\gamma < \alpha$ then add $O_i$ to $C$ and add $j$ to $B$
      – else
          ∗ Pick entry corresponding to $C_i$ from $B$, call it $j$.
          ∗ $Z_c = r_i(j - F + 1, j + 1)$, $Z = R(f - F, f)$
          ∗ $Z_{cn} = normalize(Z, Z_c)$
          ∗ Compute $\gamma$ by using Equation 1
          ∗ if $\gamma > \alpha$ then remove $C_i$ from $C$ and add $B_i$ from $B$,
            otherwise $B(i) = B(i) + 1$
```

Assuming that differences between the pair of samples originate from additive Gaussian noises, we construct a multivariate Gaussian probability density function for interclass (Λ_i) and intra-class variation (Ω_i^j) sets by treating each column as a vector of random variables. For simplicity, we assume each variable to be independent and estimate the mean and standard deviation using the standard formulas. Now, when a new object O_k is detected we proceed as follows to find out if it is one of the unobservable objects whose locations we are predicting using motion context. The region belonging to O_k is extracted, resized to the predefined size, vectorized and is represented by s. For each $O_j \in P$, an intraclass variation vector set is constructed by using all of the pairwise combinations of s and observation in X_j. From this vector set, we pick up that particular vector whose distance from the learned pdf of intraclass variation for O_j is minimum. The same process is repeated for all other $O_j \in P$ and the O which has the minimum distance is selected. A second stage of verification is then performed with respect to inter-class variation pdf's of O_j. If the distance of s is minimum with respect to all these, final appearance based association is carried out.

5. Experiments and Results

In this section, we report the experimental performance of our target tracking and reacquiring approach. In each experiment, we demonstrate that the approach is able to reacquire objects accurately across occlusions and as well as for the case of entries and exits from the field of view of the camera. The first set of videos employed for testing consists of challenging sequences taken from the Getty-Images website. In these the types of occlusions include single overhead bridge to multiple overhead bridges. The second set consists of the VIVID data corpus. The detection and tracking within each sequence were done automatically for all the sequences using the COCOA system [15]. Three sequences were used in total including two from VIVID and one from Getty-Images web site. The VIVID sequence contains a convoy of cars tracked by the UAV. Due to rapid camera motion, a particular car remains visible only for short durations of time before going out of the field of view of the camera. In absence of a reacquisition methodology the obtained tracks were broken.The sequence taken from Getty-Images captures a busy road intersection, where cars are moving along multiple paths at any given time.

The experiment on VIVID data set is conducted in the following manner. The detected objects are tracked and once they become unobservable, their positions are predicted using the motion context framework discussed in Section 4.1 and 4.2. Whenever a new object enters in the field of view of the camera, we match it with all of the predicted tracks using spatial and appearance constraints using motion and appearance context. Qualitative results for three different tracks, which were repaired by our method, are shown in Figures 4 (b), 4 (d), and 4 (f). Figure 4 (d) shows the scenario where the object motion is linear. Figure 4 (g) shows repairing results on a rather challenging maneuver. Here the cars are making a U-turn and with camera tracking only part of the convoy for the most part. We would like to stress that a prediction model based on a linear motion assumption like Perera at. al. [9] will not be able to handle these kinds of situations. However, reacquisition results shown in Figure 4(h) clearly demonstrate the robustness and richness of our algorithm where we are able to assign the correct labels even after the cars have made the U-turn. Another set of results is presented in Figure 4(e) where a car leaves and enters the field of view multiple times. We are able to maintain the correct label throughout the sequence as depicted by Figure 4(f). Note that in Figure 4(h), the noisiness of the tracks near the matching points can be explained in terms of the error present in the prediction model. Figure4(a) shows a track completion result for the road intersection scenario. It shows that our algorithm was able to keep the same label for the car as it re-appears from the other side of the bridge.

For analyzing the prediction accuracy, we proceeded as follows. For a given sequence with short and rare periods of occlusion and enough potential predictors, we allowed an object to be observed for F frames. After this period of observation, we artificially added the object to the list of unobservable objects, and predicted its position using our motion framework. At the end of the sequence, for every position of the predicted track of length N, we computed the distance between the predicted position p_i and the actual position t_i of the object. Next, for each sub-interval of length n of the track, we computed the mean prediction error as $MPE_n = \frac{1}{n}\sum_{i=1}^{n} dist(p_i, t_i)$. The mean prediction error calculated for two particular tracks can be seen in Figure 4(f) and 4(m). As can be seen from these figures, the mean prediction error increases with the duration of the prediction. The increase is greater when the object undergoes non-linear motion as in Figure 4 (m), indicating the difficulties of the linear regression model in predicting the position of the object. However, the error is still sufficiently small for the track repair to work, as was shown in Figure 4 (l). Similarly, we computed the variance of the error for each sub-interval of length n as $Variance_n = \frac{1}{n}\sum_{i=1}^{n}(MPE_n - dist(p_i, t_i))^2$. Figures 4(k) and 4(n) show the variance calculated for the tracks shown in Figure 4(i) and 4(l), respectively.

6. Conclusion

In this paper we have developed two new ideas of motion and appearance contexts. We have demonstrated their application for the task of target tracking and re-acquistion in aerial imagery. The future work includes development of a mechanism for robustly fusing the motion and appearance context information.

Acknowledgement: This research was funded in part by the U.S. Government VACE program.

References

[1] L. I. Piterbarg et. al., "A Simple Prediction Algorithm for the Lagrangian Motion in Two-Dimensional Turbulent Flows", SIAM Journal on Applied Mathematics, 63, 2002.

[2] L. I. Piterbarg, "Short-Term Prediction of Lagrangian Trajectories", Journal of Atmospheric and Ocean Technology, 18(8), 1398-1410, 2001.

[3] T. Ozgokmen, "Predictability of Drifter Trajectories in the Tropical Pacific Ocean", Journal of Physical Oceanography, 31(9), 2001.

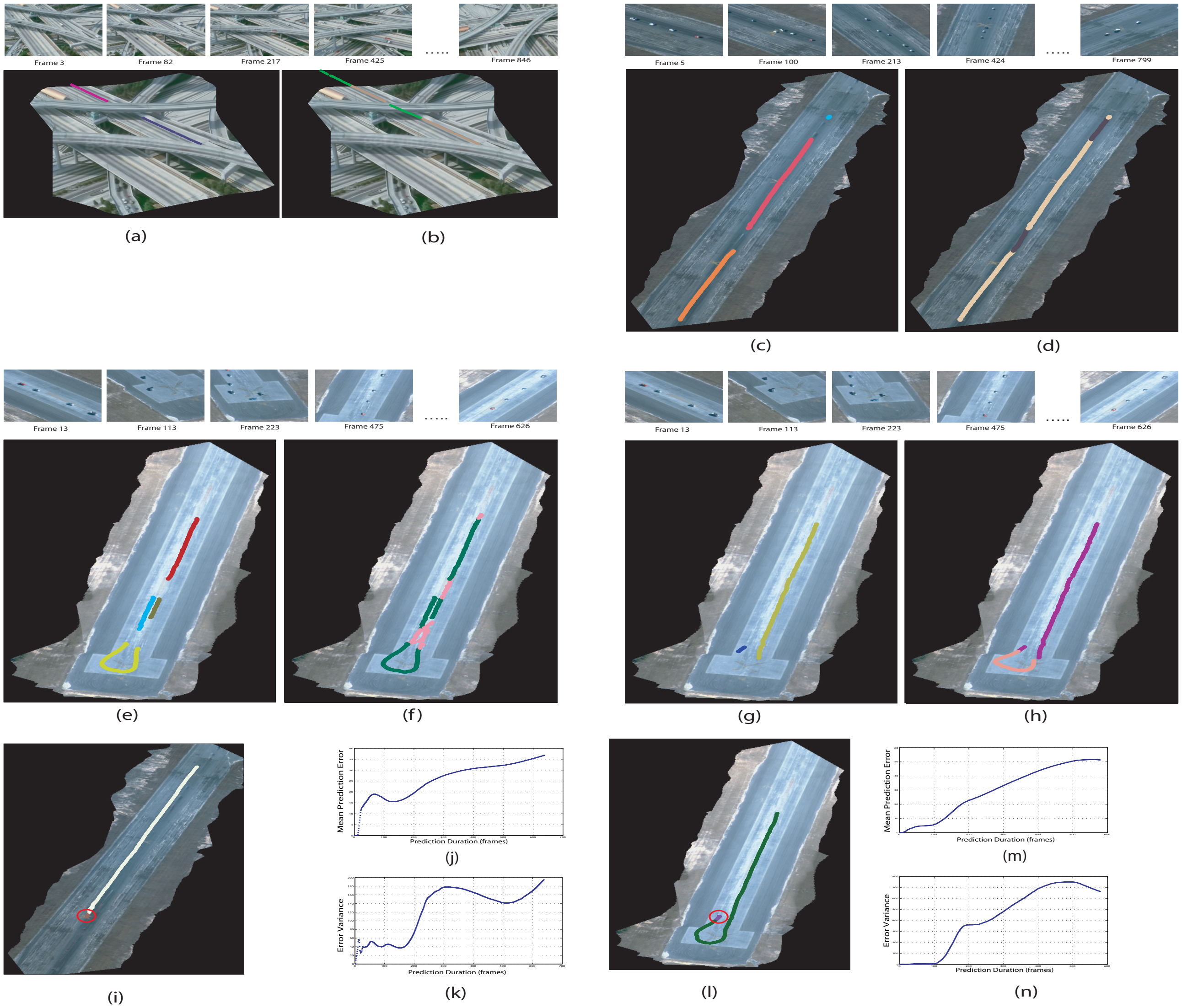

Figure 4. Figures (a), (c), (e), and (g) show differently colored track sections belonging to the same track. Figures (b), (d), (f), and (h) show results of track completion where different sections are assigned to the same track. (i) shows location estimation for a linear track. The circled part is the observed portion, the white part is the predicted portion. (j) The mean distance between the predicted and the actual track increases with duration of prediction. (k) The Variance of the distance around the mean distance. (l) Shows location estimation for curved track. (m) Mean distance between the predicted and the actual track increases with the duration of prediction. (n) The variance of the distance around the mean distance.

[4] C. Froeschle, "The Lyapunov Characteristic Exponents - Applications to Celestial Mechanics", Celestial Mechanics, 34, 1984.

[5] T. Huang et. al., "Object Identification: A Bayesian Analysis with Application to Traffic Surveillance", Artificial Intelligence, 103, 1998.

[6] V. Kettnaker and R. Zabih, "Bayesian Multi Camera Surveillance", CVPR, 1999.

[7] A. Gilbert et. al., "Tracking Objects Across Cameras by Incrementally Learning Inter-Camera Colour Calibration and Patterns of Activity", ECCV, 2006.

[8] O. Javed et. al., "Tracking in Multiple Cameras with Disjoint Views", CVPR , 2003.

[9] A. G. Perera, "Multi-Object Tracking Through Simultaneous Long Occlusions and Split and Merge Conditions", CVPR , 2006.

[10] R. Kaucic et.al., "A Unified Framework for Tracking through Occlusions and across Sensor Gaps", CVPR, 2005.

[11] Y. Shan et. al., "Vehicle Identification between Non-Overlapping Cameras without Direct Feature Matching", ICCV, 2005.

[12] Y. Shan et. al., "Unsupervised Learning of Discriminative Edge Measures for Vehicle Matching between Non-Overlapping Cameras", CVPR, 2005

[13] Y. Guo et. al., "Vehicle Fingerprinting for Reacquisition and Tracking in Videos", CVPR, 2005.

[14] O. C. Ozcanli et. al., "Augmenting Shape with Appearance in Vehicle Category Recognition", CVPR, 2006.

[15] S. Ali et. al., "COCOA - Tracking in Aerial Imagery", SPIE, 2006

[16] A. Wolf et. al, "Determining Lyapunov Exponent from a Time Series", Physica D, 16, pp. 285-317, 1985.