Hai Tao, Harpreet S. Sawhney, and Rakesh Kumar, Dynamic layer representation and its applications to tracking, in *Proc. IEEE conf. on Computer Vision and Pattern Recognition (CVPR2000)*, vol 2, pp. 134-141, June 2000.

Dynamic Layer Representation with Applications to Tracking

Hai Tao, Harpreet S. Sawhney, and Rakesh Kumar

Sarnoff Corporation,
201 Washington Rd., Princeton, NJ 08543
{htao, hsawhney, rkumar}@sarnoff.com

Abstract

A dynamic layer representation is proposed in this paper for tracking moving objects. Previous work on layered representations has largely concentrated on two-/multi-frame batch formulations, and tracking research has not addressed the issue of joint estimation of object motion, ownership and appearance. This paper extends the estimation of layers in a dynamic scene to incremental estimation formulation and demonstrates how this naturally solves the tracking problem. The three components of the dynamic layer representation, namely, layer motion, ownership, and appearance, are estimated simultaneously over time in a MAP framework. In order to enforce a global shape constraint and to maintain the layer segmentation over time, a parametric segmentation prior is proposed. The generalized EM algorithm is employed to compute the optimal solution.

We show the results on real-time tracking of multiple moving or static objects in a cluttered scene imaged from a moving aerial video camera. The moving objects may do complex motions, and have complex interactions such as passing. By using both the appearance and the segmentation information, many difficult tracking tasks are reliably handled.

1 Introduction

This paper proposes a representation of moving objects in terms of layers and applies the formulation to tracking. In order to track and maintain identity of objects over time, the object state must consist of representations of motion, appearance and ownership masks. This is called an object *layer* in the current work in accordance with the layered representation of scenes that has been studied in the past few years [Wang93]. With an object state represented as a layer, maximum *a posteriori* estimation (MAP) in a temporally incremental mode can be applied to update the state optimally for tracking. Tracking with such a complete state representation is important for applications that require segmented object appearance (for example, indexing and object insertion/removal). For applications concerning only positions and geometric transformations, it produces more robust results because most existing trackers only use partial representations. For example, change-based trackers ignore the appearance information and thus have difficulty dealing with close-by or stalled objects. Template trackers typically update motion only and hence can drift off or get attached to other objects of similar appearance [Hager96]. Some template trackers use parametric motion (affine/similarity etc.) to update both the motion and the shape of the template [Black95], however since there is no explicit updating of template ownership, drift may still occur.

The main contribution of this paper is to formulate multi-object tracking as a 2D layer estimation and tracking problem with a view towards achieving completeness of representation. In the context of layered representations, the current work enables dynamic estimation and updating of layers in contrast with the previous two-frame/multi-frame batch formulations.

Compact representation of layer ownership is a key issue for the viability of this representation. The traditional bit-mask representation of the ownership does not enforce global shape constraint and cannot be efficiently updated within a MAP framework. In this paper, we propose a parametric representation of the layer ownership prior and demonstrate its viability in object tracking.

Before describing the details of our method, a brief review of layer representation is presented in the following sub-section.

1.1 Layer representation

In the last ten years, layered representations and their associated algorithms have emerged as powerful tools for motion analysis. With compact and comprehensive underlying representation, these algorithms have convincingly demonstrated the ability to precisely segment and estimate motion of multiple independent 2D components in dynamics scene. Compared to the model-based approach, the layer representation is data-driven and imposes weaker prior constraints on segmentation, motion, and appearance of objects.

The key idea behind layer estimation is to simultaneously estimate the object motion and segmentation based on motion consistency. The bulk of the previous works focus on formulating various constraints on layer motion and layer segmentation. The constraints on motion reflect the

TABLE 1

	Local constraints	Global constraints	Multi-frame consistency
Motion constraints	**Smooth dense flow:** Weiss 97	**2D affine:** Darrell91, Wang93, Hsu94, Sawhney96, Weiss 96, Vasconcelos97 **3D planar:** Torr99	**2D rotation and translation & constant velocity:** This paper - Section 2.2
Segmentation constraints	**MRF segmentation prior:** Weiss96, Vasconcelos97	**Background+Gaussian segmentation prior:** This paper - Section 2.1	**Constant segmentation prior:** This paper - Section 2.1
Appearance constraints			**Constant appearance:** This paper - Section 2.3

image formation conditions and the dynamic behaviors of scenes. The constraints on segmentation, on the other hand, encode the knowledge of the scene geometry. In TABLE 1 and the following several paragraphs, we will briefly examine these constraints and propose several new constraints.

We classify the various constraints into three categories. Besides the previously mentioned motion constraints and segmentation constraints, we add one more category called appearance constraints. They impose constraints on the appearance of each layer. Each category is further divided into three types, namely, local spatial constraints, global spatial constraints, and multi-frame temporal constraints. Various constraints and some related works are listed in TABLE 1. It should be noted that this table is by no means exhaustive.

It is observed that most existing motion constraints are global. The motion of the pixels in each layer is modeled either as a single 2D affine [Darrel91] [Wang93] [Hsu94] [Weiss96] [Sawhney96] [Vasconcelos97] or projective motion [Torr99]. Local motion constraints have also been proposed in [Weiss97]. The idea is to model each motion group as a linear combination of some local basis flow fields.

Segmentation constraints usually appear in the form of priors in layer representations. Only local smoothness models such as the first order Markov random fields (MRF) have been extensively investigated. The assumption behind this model is that pixels spatially close to each other tend to be in the same layer. This is obviously insufficient to encode global shape constraints such as layers in a scene having priors for a round or square shape. In this paper, we will show how a Gaussian shape prior can be used to handle these cases.

The traditional layer methods are limited to two-frame or multi-frame batch formulation. When more than two images are given, additional constraints are available across the images. These constraints can be either enforced in a batch mode or in a tracking mode. In this paper, we will only discuss the tracking mode, in which the MAP solution of the current layer representation is obtained. In the context of tracking, multi-frame constraints are imposed as temporal constraints. A temporal motion constraint states that the motion in each frame should satisfy a dynamic model, e.g. a constant velocity model. A temporal constraint on the segmentation prior on the other hand, represents the dynamics of the shape change across images.

When multiple images are considered, constraints for the layer appearance need to be considered. A constant appearance model is applied in this paper.

With the above new constraints, the problem of estimating layer representation over time is equivalent to optimizing

$$P(motion_t, appearance_t, segmentation\_prior_t \mid image_t, image_{t-1}, motion_{t-1}, appearance_{t-1}, segmentation\_prior_{t-1})$$

The quantities in the current image are marked by a subscript t. Those in the previous image have subscript $t-1$. The solution is obtained by applying EM algorithm with the actual segmentation as the hidden variable. The details of each constraint used in our algorithms are presented in Section 2. Section 3 describes the MAP estimation problem. Some implementation issues and experimental results will be shown in Section 4, followed by discussions and conclusions in Section 5.

2 Constraints for the dynamic layer tracker

In this section, it is assumed that the number of layers in the scene and the initial layer representation are provided by an external agent (see Section 4 for details). We denote the number of layers as g, n as the number of pixels in the input image I_t, and x_i as the image coordinates of the ith pixel.

2.1 Dynamic segmentation prior

The motivation for employing a global parametric shape prior is twofold. Firstly, this prior prevents a segment from evolving into arbitrary shapes in the course of tracking. As a result, it assists in tracking when ambiguous or cluttered measurements occur. Secondly, only the compact parametric form needs to be updated in the state over time. This makes the estimation process computationally tractable. The layer segmentation prior is application-dependent. It encodes the knowledge regarding the geometry of the objects. We emphasize that since the segmentation constraints are only priors, they need not encode the exact shape of the tracked objects.

For the problem of tracking vehicles in airborne video, the dominant image region is the ground. Its motion can be modeled as projective planar (layer 0). Vehicles moving on the ground are the foreground layers (layer 1 to $g-1$). Their supports are usually rectangular, whose prior can be conveniently modeled as local Gaussian distributions for computational reasons (see Figure 1).

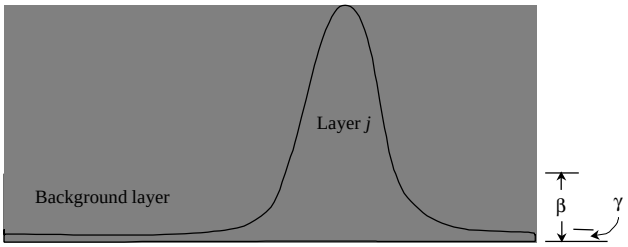


Figure 1. A background+Gaussian segmentation prior function $L_{t,j}(x_i)$.

At each instant of time, the prior of the background layer has a constant value b for every pixel. It means that every pixel has a constant prior of belonging to the background layer. The prior for each foreground layer j is $g+\exp[-(x_i-m_{t,j})^T S_{t,j}^{-1}(x_i-m_{t,j})/2]$. $m_{t,j}$ is the center of the distribution and $S_{t,j}$ is the covariance matrix that defines the span of the distribution. One of the consequences of this model is that pixels with larger distances from any foreground layer center will have higher prior of belonging to the background. This prior is combined with the image likelihood to produce the final segmentation. g is a small positive value. It allows pixels to belong to a foreground layer even if they are relatively far away from the layer center as long as their likelihood values are high. Therefore, g represents the uncertainty of the layer shape. Including this uncertainty in the prior is important because the shapes of vehicles are not exactly elliptical and change constantly over time.

In summary, the prior for a pixel x_i belonging to layer j is defined as:

$$\mathrm{L}_{t,j}(x_i)=\begin{cases} g+\exp[-(x_i-m_{t,j})^T S_{t,j}^{-1}(x_i-m_{t,j})/2] & j=1,\ldots,g-1 \\ b & j=0 \end{cases}$$

The normalized prior is computed as:

$$S_{t,j}(x_i)=L_{t,j}(x_i)\Big/\sum_{j=0}^{g-1}L_{t,j}(x_i)\,. \qquad (1)$$

The covariance matrix $S_{t,j}$ is defined as

$$S_{t,j}=R^T(-w_{t,j})Diag[1/l_{t,j}^2,1/s_{t,j}^2]R(-w_{t,j})$$

where $l_{t,j}$ and $s_{t,j}$ are proportional to the lengths of the major and the minor axes of the iso-probability contours and thus describe the shape of each foreground layer. The translation $m_{t,j}$ and the rotation angle $w_{t,j}$ are motion parameters and will be discussed in the next subsection. $F_{t,j}=[l_{t,j},s_{t,j}]$ denotes the shape prior parameter of layer j in the current image. The dynamic model for the shape prior is a constant shape model:

$$P(F_{t,j}\,|\,F_{t-1,j})=N(F_{t,j}:F_{t-1,j},diag[s_{ls}^2,s_{ls}^2]) \qquad (2)$$

where $N(x:h,S)$ is a Gaussian distribution.

2.2 Motion constraints

In an aerial video tracking application, the background motion is modeled as a projective planar motion. With the background motion compensated, the motion of the foreground layer j is approximated using a translation $m_{t,j}$ and rotation $w_{t,j}$. The motion parameter vector is $Q_{t,j}=[\dot{m}_{t,j}^T,\dot{w}_{t,j}]^T$, where $\dot{m}_{t,j}$ is the translation velocity and $\dot{w}_{t,j}$ is the rotation velocity. The commonly used constant 2D velocity model states that

$$P(Q_{t,j}\,|\,Q_{t-1,j})=N(Q_{t,j}:Q_{t-1,j},diag[s_m^2,s_m^2,s_w^2]) \qquad (3)$$

and $m_{t,j}=m_{t-1,j}+\dot{m}_{t,j}$, $w_{t,j}=w_{t-1,j}+\dot{w}_{t,j}$.

2.3 Dynamic layer appearance model

The appearance image of layer j is denoted by $A_{t,j}$. A local coordinate system is defined by the center and the axes of the Gaussian segmentation prior. The coordinate transform from the original image to this local coordinate system is $T_j(x_i)=R(-w_j)(x_i-m_j)$. This transform is determined by the motion parameters of layer j. For any pixel x_i in the original image, the observation model for layer j is

$$P(I_t(x_i)\,|\,A_{t,j}(T_j(x_i)))=N(I_t(x_i):A_{t,j}(T_j(x_i)),s_I^2) \qquad (4)$$

A constant appearance model assumes that, for any pixel $T_j(x_i)$ in the appearance image,

$$P(A_{t,j}(T_j(x_i)) \mid A_{t-1,j}(T_j(x_i))) = N(A_{t,j}(T_j(x_i)) : A_{t-1,j}(T_j(x_i)), s_A^2) \quad (5)$$

3 EM algorithm and the layer tracker

3.1 EM algorithm

Let $L_t = (F_t, Q_t, A_t)$ denote the layer representation at each instant of time, where F_t is the segmentation prior, Q_t is the motion, and A_t is the appearance. The goal is to find L_t that maximizes the posterior probability

$$\max_{L_t} \arg P(L_t \mid I_t, L_{t-1}, I_{t-1}) = \max_{L_t} \arg P(I_t \mid L_t, L_{t-1}, I_{t-1}) P(L_t \mid L_{t-1}, I_{t-1}) \quad (6)$$

The EM algorithm can solve this MAP problem by explicitly computing layer ownership (segmentation). According to the generalized EM algorithm, a local optimal solution can be achieved by iteratively optimizing or improving the following function Q with respect to L_t (see Appendix A for a proof).

$$Q = E[\log P(I_t, z_t \mid L_t, L_{t-1}, I_{t-1}) \mid I_t, L_t', L_{t-1}, I_{t-1}] + \log P(L_t \mid L_{t-1}, I_{t-1}) \quad (7)$$

where z_t is a hidden variable that indicates which layer each pixel belongs to and L_t' is the result of the previous iteration. According to Appendix B, this is equivalent to iteratively optimizing or improving the function

$$\sum_{i=1}^{n}\sum_{j=0}^{g-1} h_{i,j}\{\log S_{t,j}(x_i) + \log P(I_t(x_i) \mid A_{t,j}(T_j(x_i)))\} + \sum_{j=1}^{g-1}\{\log N(F_{t,j} : F_{t-1,j}, diag[s_{ls}^2, s_{ls}^2]) + \log N(Q_{t,j} : Q_{t-1,j}, diag[s_m^2, s_m^2, s_w^2]) + \sum_{i=0}^{n-1} \log(N(A_{t,j}(T_j(x_i)) : A_{t-1,j}(T_j(x_i)), s_A^I))\} \quad (8)$$

where $h_{i,j}$ is the layer ownership - the posterior probability of pixel x_i belonging to layer j conditioned on L_t'. It should be noted that the shape constraint is only employed as a prior. This is different from the shape constraints used in many model-based tracking algorithms, where the shape constraint defines the actual segmentation.

3.2 Optimization

Because it is difficult to optimize F_t, Q_t and A_t simultaneously in (8), we adopt the strategy of improving one of them with the other two fixed. This is the generalized EM algorithm and it can be proved that this also converges to a local optimal solution. Figure 2 summarizes the optimization process. As shown in the figure, motion parameters of the layers are computed first. Then the segmentation prior and the appearance are re-estimated. Every time F_t, Q_t or A_t are re-estimated, the layer ownership $h_{i,j}$ needs to be updated. Multiple iterations may be executed before proceeding to the next image. Individual steps are elaborated in the following subsections.

3.2.1 Update the layer ownership

The layer ownership $h_{i,j}$ is computed as

$$h_{i,j} = P(z_t(x_i) = j \mid I_t, L_t', L_{t-1}, I_{t-1}) = P(I_{t-1}(x_i) \mid A_{t,j}'(T_j(x_i)))S_{t,j}(x_i)/Z \quad (9)$$

The first two terms can be computed according to (4) and (1). Factor Z normalizes $h_{i,j}$ so that $\sum_{j=0}^{g-1} h_{i,j} = 1$.

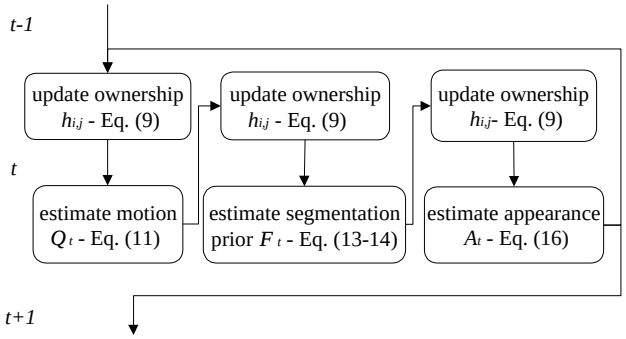


Figure 2. The dynamic layer tracking algorithm.

3.2.2 Motion estimation

If we assume that the segmentation prior and the appearance are known, motion estimation step finds Q_t that improves

$$\sum_{j=1}^{g-1} \log N(Q_{t,j} : Q_{t-1,j}, diag[s_m^2, s_m^2, s_w^2]) + \sum_{i=1}^{n}\sum_{j=0}^{g-1} h_{i,j}\{\log S_{t,j}(x_i) + \log P(I_t(x_i) \mid A_{t,j}(T_j(x_i)))\} \quad (10)$$

The motion of individual foreground layers are estimated sequentially according to

$$\min_{Q_{t,j}} \arg |\dot{m}_{t,j} - \dot{m}_{t-1,j}|/s_m^2 + |\dot{w}_{t,j} - \dot{w}_{t-1,j}|/s_w^2 - \sum_{i=0}^{n-1} 2h_{i,j}\log S_{t,j}(x_i) + \sum_{i=0}^{n-1} h_{i,j}(I_t(x_i) - A_{t,j}(T_j(x_i)))^2/s_I^2 \quad (11)$$

The first term is the motion prior. The second term is the correlation between the layer ownership and the log function of the segmentation prior. The third term is the weighted sum of squared differences between the image and the appearance of layer j under motion $Q_{t,j}$. The solution is obtained by searching in the translation and the rotation space. For the background layer, the motion can be approximated using a direct method [Bergen92].

3.2.3 *Shape estimation*

The shape prior parameter F_t is estimated as

$$\max_{F_t} \arg f = \sum_{j=0}^{q-1} \log N(F_{t,j} : F_{t-1,j}, diag[s_{ls}^2, s_{ls}^2]) + \sum_{i=0}^{n-1}\sum_{j=0}^{q-1} h_{i,j} \log S_{t,j}(x_i) \quad (12)$$

Gradient descent is used to optimize this function. According to Appendix C,

$$\frac{\partial f}{\partial l_{t,j}} = \sum_{i=0}^{n-1} \frac{h_{i,j}(D(x_i) - L_{t,j}(x_i))}{L_{t,j}(x_i) D(x_i)} (L_{t,j}(x_i) - g) y_{i,j,x}^2 / l_{t,j}^3 - (l_{t,j} - l_{t-1,j}) / s_{ls}^2 \quad (13)$$

and similarly

$$\frac{\partial f}{\partial s_{t,j}} = \sum_{i=0}^{n-1} \frac{h_{i,j}(D(x_i) - L_{t,j}(x_i))}{L_{t,j}(x_i) D(x_i)} (L_{t,j}(x_i) - g) y_{i,j,y}^2 / s_{t,j}^3 - (s_{t,j} - s_{t-1,j}) / s_{ls}^2 \quad (14)$$

where $D(x_i) = \sum_{j=0}^{q-1} L_{t,j}(x_i)$ and $[y_{i,j,x}, y_{i,j,y}]^T = R(-w)(x_i - m_j)$.

3.2.4 *Appearance estimation*

The next step is to update appearance model of each layer with Q_t and F_t fixed according to

$$\max_{A_{t,j}} \arg \sum_{i=0}^{n-1} \{\log(N(A_{t,j}(T_j(x_i)) : A_{t-1,j}(T_j(x_i)), s_A^2)) + h_{i,j} \log P(I_t(x_i) \mid A_{t,j}(T_j(x_i)))\} \quad (15)$$

From Appendix D, $A_{t,j}(T_j(x_i))$ is directly computed as

$$A_{t,j}(T_j(x_i)) = \frac{A_{t,j}(T_j(x_i))/s_A^2 + h_{i,j} I_t(x_i)/s_I^2}{(1/s_A^2 + h_{i,j}/s_I^2)} \quad (16)$$

This is the weighted average of the previous template and the current image.

4 Implementation and experimental results

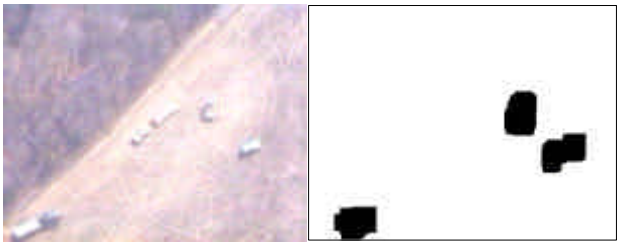
4.1 Aerial video surveillance system (AVS)

A typical video frame from an AVS video is shown in Figure 3(a). These videos are take from a camera mounted on an airplane. Our goal is to reliably track all the vehicles in the scene. The size of the video image is 320×240 pixels. The size of the vehicles ranges from 10×10 to 40×40 pixels.

4.2 Initialization and status determination

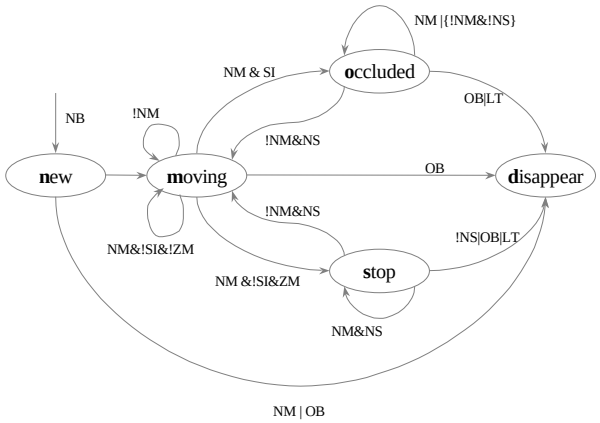
Besides the core tracking algorithm described Figure 2, other issues that need to be handled are: (1) initialization of layers, (2) deletion and addition of layers, (3) determination of object status such as stopped and occluded. These are important for practical applications. These tasks are accomplished through an external module. The inputs to this module include the change blob images (Figure 3(b)) and the estimation results of current layer representation. The kernel of this module is a state machine. As shown in Figure 4, there are five states in the state machine. Each directed edge represents a transition. The condition for transition is marked along the edge. For example, a new object is initialized if a new change blob is detected far away from existing objects. An object is deleted if it is out of the field of view. An object is marked as stopped if all the following three conditions are satisfied (1) its motion blob disappears, (2) no significant decrease of correlation score, (3) the estimated motion is zero.

When a new layer (vehicle) is added, an initialization step estimates the three components of a layer from the change blob and the image. More specifically, the position of the object is located at the center of the blob. A zero velocity is assigned. The segmentation prior is estimated from the second order moments of the blob. The appearance is obtained from the original image.



(a) (b)

Figure 3. (a) A typical frame in an aerial surveillance video and (b) its change blob image. Only three vehicles are moving.



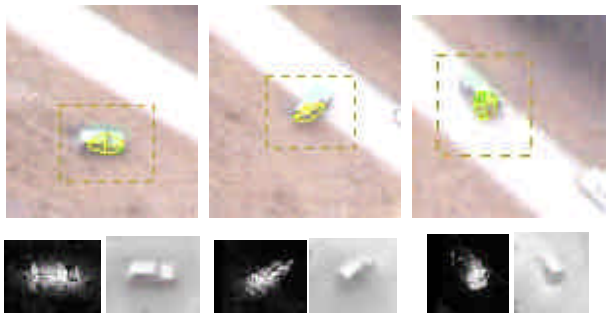
Conditions

NS = normal SSD score
OB = out of scope
LT = NM for a long time
ZM= zero motion estimation
NB = new blob, no object covering a blob
NM = no motion blob covering the object
SI = significant increase of SSD

Figure 4. State machine of the dynamic layer tracker.

4.3 A real-time tracking system

The computational bottleneck in the real-time implementation of the proposed algorithm is the motion estimation step, which accounts for more than 95% of the computation. In our implementation, the dominant background motion parameters are estimated at video rate using a hardware implementation of a direct method [Bergen92]. This information together with the video frames are then fed to a tracking system that runs on an SGI Octane workstation, where the foreground motion is estimated using a multi-resolution template matching method. A low resolution change blob image is also computed on the workstation. Though multiple iterations of the EM algorithm may be performed in each frame, we found that a single iteration is sufficient. The current system can handle two moving objects at 10 Hz or four moving objects at 5 Hz.



(a) (b) (c)

Figure 5. Vehicle turning. The first row shows the cutouts of the original video frames and the Gaussian segmentation prior. The next two rows show the segmentation and the appearance (warped to the image coordinates). (a) frame 145 (b) frame 180 (c) frame 210.

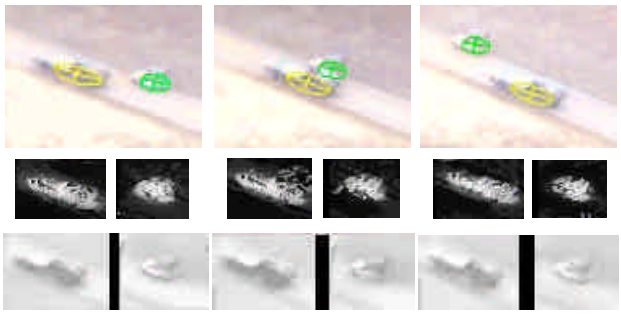
4.4 Robust tracking of multiple vehicles

A tracking system is designed to handle complex motions and complex interactions such as passing and stopping. In Figure 5, the tracking results on a clip with a turning vehicle are demonstrated. In this example, the appearance, shape, and motion of the vehicle change dramatically. The layer tracker, however, has estimated them correctly and maintains the track.

Tracking vehicle interactions is difficult for change-based trackers because motion is the only cue to distinguish merged blobs after they split. This is not reliable when the merge lasts a long period of time. A template tracker does not keep track of ownership. It would lose track too because the other vehicle will confuse the tracker. The layer tracker however, maintains the appearance information and reliable tracking can still be achieved.

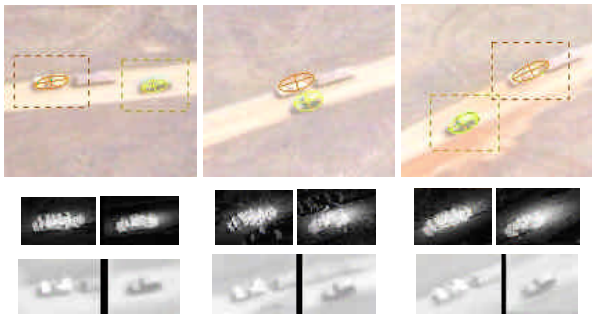
In Figure 6, the tracking results on vehicles passing from opposite directions are demonstrated. This is a relatively easy example because the passing is brief. In Figure 7, the tracking results on vehicles passing in the same direction are shown. This is more challenging because the vehicles remain close to each other longer and they have similar motions.

In Figure 8, three vehicles are tracked. One of them is stopped. A change-based tracker can not handle this scenario because appearance information is needed for tracking stopped objects.



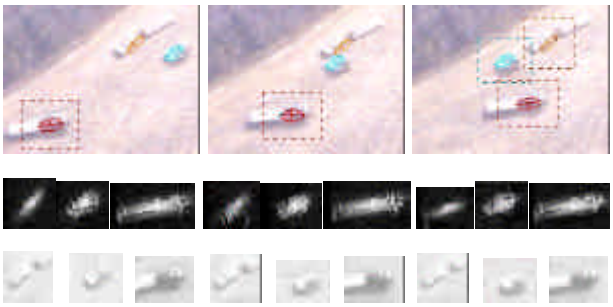
(a) (b) (c)

Figure 6. Passing (opposite directions). (a) frame 36 (b) frame 41 (c) frame 49.



(a) (b) (c)

Figure 7. Passing (same direction). (a) frame 178 (b) frame 220 (c) frame 253.



(a) (b) (c)

Figure 8. Stop and passing of vehicles. (a) frame 273 (b) frame301 (c) frame 321.

5 Discussions and conclusions

A dynamic layer representation has been proposed in this paper to solve the tracking problem. This new representation includes all three key elements for a tracker. It is updated using the EM algorithm in a MAP estimation framework. Compared to the traditional layer formulation, new extensions include an appearance model, a global segmentation prior, and three temporal consistency constraints (TABLE I).

One advantage of the proposed algorithm over many other trackers is that the background and the objects compete with each other in the layer estimation using motion cues. This improves the robustness of the tracker against the cluttered background and makes the tracking process more resilient to distraction from other close-by objects.

The difference between the Gaussian segmentation prior from a Gaussian model in a model-based approach is that in the latter, the actual pixel-wise segmentation is not computed and if the shape of the object is not similar to an ellipse, it will erroneously use the background pixel for motion estimation. In the proposed method, the global shape constraint acts as a segmentation prior and is a weaker constraint. The actual segmentation is still computed. Both the data-driven property of the layer approach and the efficiency of the model-based approach are preserved. An interesting question is how to incorporate more complicated segmentation priors for objects such as human bodies into this framework.

Acknowledgments

This work was partly supported by DARPA grant DAAB07-98-C-J023. Authors would like to thank D. Hirvonen, S. Samarasekera, and M. Hansen for their support in the development of this algorithm.

References

[Darrel91] T. Darrell and A. Pentland, Robust estimation of multi-layered motion representation, in *Proc. IEEE Workshop on Visual Motion*, pp. 173-178, Princeton, 1991.

[Wang93] J. Y. A. Wang and Edward H. Adelson, Layered representation for motion analysis, in *Proc. of IEEE conference on Computer Vision and Pattern Recognition*, pp. 361-366, 1993.

[Irani93] M. Irani and S. Peleg, Motion analysis for image enhancement: resolution, occlusion, and transparency. *Journal of Visual Communication and Image Representation*, Vol. 4, No. 4, pp. 324-335, December 1993.

[Hsu94] S. Hsu, P. Anandan, S. Peleg, Accurate computation of optical flow by using layered motion representations, in *Proc. Int. Conference on Pattern Recognition*, Jerusalem, 1994.

[Sawhney96] H. S. Sawhney and S. Ayer, Compact representations of motion video using dominant and multiple motion estimation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8), pp 814-830, 1996.

[Weiss96] Y. Weiss and E. H. Adelson, A unified mixture framework for motion segmentation: incorporating spatial coherence and estimating the number of models, in *Proc. of IEEE conference on Computer Vision and Pattern Recognition*, pp. 321-326, 1996.

[Weiss97] Y. Weiss, Smoothness in Layers: motion segmentation using nonparametric mixture estimation, in *Proc. of IEEE conference on Computer Vision and Pattern Recognition*, 520-526, 1997.

[Vasconcelos97] N. Vasconcelos and A. Lippman, Emprical Bayesian EM-based motion segmentation, in *Proc. of IEEE conference on Computer Vision and Pattern Recognition*, pp. 527-532, 1997.

[Torr99] P. H. S. Torr, R. Szeliski, and P. Anandan, An integrated Bayesian approach to layer extraction from image sequences, in *Proc. of IEEE conference on Computer Vision and Pattern Recognition*, pp. 983-990, 1999.

[Bergen92] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani, Hiearchical model-based motion estimation, in *Proc. of 2nd European Conference on Computer Vision*, pp. 237-252, 1992.

[Black95] M. J. Black and Y. Yacoob, Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motion, in *Proc. Fifth International Conf. on Computer Vision, ICCV'95*, pp 374-381, 1995.

[Hager96] G. Hager and P. Belhumeur, Real-time tracking of image regions with changes in geometry and illumination, in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 403-410, 1996.

Appendix A

Suppose x is the hidden variable, y is the observed variable, and q is the variable to be estimated. Our task is to find q that maximizes the posterior probability $P(q \mid y)$. In general, it is difficult to solve this optimization problem. Generalized EM algorithm finds a local solution by iteratively improving q.

Suppose some initial estimation q' is available. Taking the expectation of $\log P(q, y)$ with respect to $P(x \mid q', y)$. The result is still $\log P(q, y)$ because it does not contain variable x. This is written as

$$\log P(q, y) = E[\log P(q, y) \mid q', y].$$

By applying the identity $\log P(y \mid q) = \log P(x, y \mid q) - \log P(x \mid q, y)$, the right side is expanded as

$$\begin{aligned} & E[\log P(q, y) \mid q', y] \\ &= E[\log P(y \mid q) \mid q', y] + E[\log P(q) \mid q', y] \\ &= E[\log P(x, y \mid q) \mid q', y] - E[\log P(x \mid q, y) \mid q', y] + \\ & \quad E[\log P(q) \mid q', y] \end{aligned} \tag{I}$$

The goal is to find $q = q''$ to improve this quantity. We note without proof that the second term in (I) is minimized when $q = q'$. Now consider any value q'' such that

$$E[\log P(x,y\,|\,q'')\,|\,q',y]+E[\log P(q'')\,|\,q',y]> \quad \Leftrightarrow$$
$$E[\log P(x,y\,|\,q')\,|\,q',y]+E[\log P(q')\,|\,q',y]$$

$$E[\log P(x,y\,|\,q'')\,|\,q',y]+\log P(q'')> \quad \text{(II)}$$
$$E[\log P(x,y\,|\,q')\,|\,q',y]+\log P(q')$$

and note that if we replace q' by q'' we increase the second term as well. As the result

$$P(q'',y)>P(q',y) \;\text{ or }\; P(q''\,|\,y)>P(q'\,|\,y)\,.$$

Therefore, any q'' satisfying (II) is an improvement over q'.

Appendix B

If we reasonably assume that the segmentation prior of each pixel is independent to each other conditioned on the shape parameters, i.e.

$$\log P(z_t\,|\,L_t,L_{t-1},I_{t-1})=\sum_{i=0}^{n-1}\log P(z_t(x_i)\,|\,L_t,L_{t-1},I_{t-1})$$

and the likelihood of each pixel belonging to a certain layer is independent to each other too, i.e.

$$\log P(I_t,|\,z_t,L_t,L_{t-1},I_{t-1})=\sum_{i=0}^{n-1}\log P(I_t(x_i)\,|\,z_t(x_i),L_t,L_{t-1},I_{t-1})\,.$$

Then, the function Q in (7) can be expanded by explicitly computing the expectation

$$\begin{aligned}Q=&\sum_{i=0}^{n-1}\sum_{j=0}^{q-1}P(z_t(x_i)=j\,|\,I_t,L_t',L_{t-1},I_{t-1})\{\\&\log P(z_t(x_i)=j\,|\,L_t,L_{t-1},I_{t-1})+\\&\log P(I_t(x_i)\,|\,z_t(x_i)=j,L_t,L_{t-1},I_{t-1})\}+\log P(L_t\,|\,L_{t-1})\end{aligned}$$

If we use $h_{i,j}$ to denote $P(z_t(x_i)=j\,|\,I_t,L_t',L_{t-1},I_{t-1})$. It is the distribution over which the expectation is taken.

As the segmentation prior $P(z_t(x_i)=j\,|\,L_t,L_{t-1},I_{t-1})$ equals to the $S_{t,j}(x_i)$ defined in (1),

$$\begin{aligned}Q=&\sum_{i=0}^{n-1}\sum_{j=0}^{q-1}h_{i,j}\{\log S_{t,j}(x_i)+\\&\log P(I_t(x_i)\,|\,z_t(x_i)=j,L_t,L_{t-1},I_{t-1})\}+\log P(L_t\,|\,L_{t-1})\\=&\sum_{i=0}^{n-1}\sum_{j=0}^{q-1}h_{i,j}\{\log S_{t,j}(x_i)+\log P(I_t(x_i)\,|\,A_{t,j}(T_j(x_i)))\}+\\&\log P(L_t\,|\,L_{t-1})\end{aligned}$$

According to (2), (3), and (5), substitute the prior with

$$\begin{aligned}&\log P(L_t\,|\,L_{t-1})=\log P(F_t,Q_t,A_t\,|\,F_{t-1},Q_{t-1},A_{t-1})\\&=\sum_{j=0}^{q-1}\{\log N(F_{t,j}:F_{t-1,j},diag[s_{ls}^2,s_{ls}^2])+\\&\log N(Q_{t,j}:Q_{t-1,j},diag[s_m^2,s_m^2,s_w^2])+\\&\sum_{i=0}^{n-1}\log(N(A_{t,j}(T_j(x_i)):A_{t-1,j}(T_j(x_i)),s_A^2))\}\end{aligned}$$

We obtain

$$\begin{aligned}Q=&\sum_{i=0}^{n-1}\sum_{j=0}^{q-1}h_{i,j}\{\log S_{t,j}(x_i)+\log P(I_t(x_i)\,|\,A_{t,j}(T_j(x_i)))\}+\\&\sum_{j=0}^{q-1}\{\log N(F_t:F_{t-1},diag[s_{ls}^2,s_{ls}^2])+\\&\log N(Q_{t,j}:Q_{t-1,j},diag[s_m^2,s_m^2,s_w^2])+\\&\sum_{i=0}^{n-1}\log(N(A_{t,j}(T_j(x_i)):A_{t-1,j}(T_j(x_i)),s_A^2))\}\end{aligned}$$

Appendix C

Taking the derivative of the objective function (12), we have

$$\begin{aligned}&\frac{\partial f}{\partial l_{t,j}}=\frac{-(l_{t,j}-l_{t-1,j})^2/2s_{ls}^2}{\partial l_{t,j}}+\sum_{i=0}^{n-1}\frac{h_{i,j}}{S_{t,j}(x_i)}\frac{\partial S_{t,j}(x_i)}{\partial l_{t,j}}\\&=-(l_{t,j}-l_{t-1,j})/s_{ls}^2-\\&\quad 1/2\sum_{i=0}^{n-1}\frac{h_{i,j}D(x_i)}{L_{t,j}(x_i)}\frac{D(x_i)-L_{t,j}(x_i)}{D^2(x_i)}(L_{t,j}(x_i)-g)\cdot\\&\quad\frac{\partial(x_i-m_j)^T R^T(-w)Diag[1/l_{t,j}^2,1/s_{t,j}^2]R(-w)(x_i-m_j)}{\partial l_{t,j}}\\&=-(l_{t,j}-l_{t-1,j})/s_{ls}^2-1/2\sum_{i=0}^{n-1}\frac{h_{i,j}(D(x_i)-L_{t,j}(x_i))}{L_{t,j}(x_i)D(x_i)}(L_{t,j}(x_i)-g)\\&\quad\frac{\partial y_{i,j}^T Diag[1/l_{t,j}^2,1/s_{t,j}^2]y_{i,j}}{\partial l_{t,j}}\\&=-(l_{t,j}-l_{t-1,j})/s_{ls}^2+\sum_{i=0}^{n-1}\frac{h_{i,j}(D(x_i)-L_{t,j}(x_i))}{L_{t,j}(x_i)D(x_i)}(L_{t,j}(x_i)-g)y_{i,j,x}^2/l_{t,j}^3\end{aligned}$$

where $D(x_i)=\sum_{j=0}^{q-1}L_{t,j}(x_i)$ and $[y_{i,j,x},y_{i,j,y}]^T=R(-w)(x_i-m_j)$.

Appendix D

Taking the derivative of the objective function (15) with respect to the brightness value of each template pixel and if the gradient equals to 0, we have

$$\begin{aligned}&\frac{\partial}{\partial A_{t,j}(T_j(x_i))}\{-(A_{t,j}(T_j(x_i))-A_{t-1,j}(T_j(x_i)))^2/2s_A^2-\\&\qquad h_{i,j}(I_t(x_i)-A_{t,j}(T_j(x_i)))^2/2s_I^2\}\\&=-(A_{t,j}(T_j(x_i))-A_{t-1,j}(T_j(x_i)))/s_A^2-h_{i,j}(A_{t,j}(T_j(x_i))-I_t(x_i))/s_I^2\\&=-(1/s_A^2+h_{i,j}/s_I^2)A_{t,j}(T_j(x_i))+A_{t-1,j}(T_j(x_i))/s_A^2+h_{i,j}I_t(x_i)/s_I^2\\&=0\Leftrightarrow\\&A_{t,j}(T_j(x_i))=\frac{A_{t-1,j}(T_j(x_i))/s_A^2+h_{i,j}I_t(x_i)/s_I^2}{(1/s_A^2+h_{i,j}/s_I^2)}\end{aligned}$$