Spatial Divide and Conquer with Motion Cues for Tracking through Clutter

Zhaozheng Yin and Robert Collins
Department of Computer Science and Engineering
The Pennsylvania State University
{zyin, rcollins}@cse.psu.edu

Abstract

Tracking can be considered a two-class classification problem between the foreground object and its surrounding background. Feature selection to better discriminate object from background is thus a critical step to ensure tracking robustness. In this paper, a spatial divide and conquer approach is used to subdivide foreground and background into smaller regions, with different features being selected to distinguish between different pairs of object and background regions. Temporal cues are incorporated into the process using foreground motion prediction and motion segmentation. Appearance weight maps tailored to each spatial region are merged and combined with the motion information to form a joint weight image suitable for mean-shift tracking. Examples are presented to illustrate that divide and conquer feature selection combined with motion cues handles spatial background clutter and camouflage well.

1. Introduction

Persistent tracking of moving objects through changes in appearance is a challenging problem. To successfully handle appearance variation, a tracker's object appearance model must be adapted over time. However, adaptation must be done carefully to avoid drifting off the object. Common appearance-based tracking approaches such as mean-shift [6] and Lucas-Kanade [2] do not explicitly model which pixels belong to the object and which belong to the background. As a result, it is easy for pixels in the background to be mistakenly incorporated into the object appearance model, thus contributing to tracker drift.

Figure-ground separation is emerging as a key technique for drift-resistant tracking. By explicitly separating object pixels from background, a tracker can adapt to object and background appearance changes separately, and in a principled way. Common methods for figure-ground separation include motion segmentation [12] and active contours [8]. More recently, figure-ground separation for tracking has been addressed as a two-class classification problem, where object pixels must be discriminated from background pixels based on local image cues such as color or texture.

With the realization that tracking can be formulated as discriminative figure-ground classification comes a growing realization that choice of features for separating object from background is also important. By using tracking features that clearly separate object from background classes, the tracker is much less likely to drift off the object onto similar background scene patches. In this paper we consider the problem of choosing features that discriminate between object and background. We particularly focus on cases with background clutter and camouflage, where it is difficult to achieve good figure-ground separation using only a single feature.

Related Work

Our work is most closely related to Collins et.al. [5] and Avidan [1]. In [5], samples of pixels from the object and background are analyzed to perform on-line selection of discriminative features to use for tracking. The variance ratio is used to rank each feature by how well it separates empirical distributions of object and background feature values. Features that maximize average separability between the foreground object and the entire surrounding background are ranked most highly by this approach, thus is best suited to backgrounds that are relatively uniform in appearance.

Avidan [1] maintains an ensemble of weak classifiers combined via Adaboost to perform strong classification of foreground from background pixels. Each weak classifier is a least-squares linear decision function (hyperplane) in the raw, multi-dimensional feature space. This approach is slower than histogram-based methods, but generalizes to high-dimensional feature spaces. Note that it is a feature weighting approach, as opposed to feature selection, which aims to choose a lower-dimensional subset of features. Like [5], this method also discards spatial information that could be used to reason about the layout of clutter and distractor objects.

In this paper, we consider an explicit approach to deal with spatial layout of background clutter, distractors, and camouflage. We explore the idea that different features may be necessary to discriminate between the object and different portions of the scene background. For example, consider tracking a car from an aerial view. One feature may distinguish well between the car and the road in front and behind it, a second may be better at discriminating between the car and foliage at the side of the road, while yet a third feature may be needed to separate the car from a vehicle of nearly the same color passing it on the left. We generalize this idea into a divide-and-conquer strategy that spatially decomposes the background into "cells"

while choosing a feature for each cell that best discriminates it from the foreground object. Like [1] [5], we perform a "soft" object-background classification by forming a weight image representing likelihood that each pixel belongs to the object versus the background (pixels more likely to be object have high weight, while pixels more likely to be background have low weight). Foreground motion prediction and motion segmentation are fused with the appearance weight image to form a joint color-motion weight image. Mean-shift is then performed on the joint weight image to find the local mode of object location. More sophisticated techniques based on statistical sampling could be used to robustly find the mode [11].

2. Measuring Feature Separability

When seeking good tracking features, we would like simple features that reliably separate the object from the background. Similar to [5], we use raw features chosen from a set of linear combinations of RGB values. Each feature is normalized into the range of 0 to 255 and then quantized into histograms of 2b bins. Other cues could be used in the feature selection process, including edge orientations, shape contexts, texture features and flow.

2.1. Extended Variance Ratio

An evaluation criterion is needed to select the best features from the candidate set. Features that produce separable object and background class distributions should score most highly. For unimodal distributions, the variance ratio is a good measure of separability. Given a feature, let $H_{obj}(i)$ be the histogram on the object and $H_{bg}(i)$ be the histogram on the background. We normalize to form probability density functions of object and background, $p(i)$ and $q(i)$. The overall combined density of the object and background is then

$$p_{tot} = \frac{p(i) + q(i)}{2} \tag{1}$$

The original variance ratio is defined in terms of the raw object and background distributions as

$$VR(p,q) = \frac{\mathrm{var}(p_{tot})}{\mathrm{var}(p) + \mathrm{var}(q)} \tag{2}$$

where *var(p)* denotes the variance of a distribution *p*. The intuition behind the variance ratio is to select features that maximize the difference between object and background classes while minimizing the variation within each class.

To evaluate separability of multimodal distributions, we adopt an extended variance ratio criterion [5]. A log likelihood transformation[1]

$$L(i) = \log \frac{p(i)}{q(i)} \tag{3}$$

nonlinearly maps raw feature values into a new feature space such that values that appear more often on the object map to unimodal positive values, and values that appear more often on the background map to unimodal negative values. The variance ratio is then applied to this new log likelihood feature to evaluate separability of the original raw feature distributions. This can be thought of in terms of an extended variance ratio

$$EVR(L;p,q) = \frac{\mathrm{var}(L;p_{tot})}{\mathrm{var}(L;p) + \mathrm{var}(L;q)} \tag{4}$$

where *var(L;p)* denotes the variance of log likelihood function *L* with respect to distribution *p* [9].

2.2. Comparing Separability Measures

One question to ask is whether the extended variance ratio is the best measure of separability of two distributions, or whether alternative information theoretic measures like KL divergence might be more appropriate.

In this section we compare the extended variance ratio to KL divergence, cross entropy and a measure similar to mutual information. The Kullback-Leibler divergence or relative entropy is a measure of the difference between two probability distributions; however it is not symmetric and does not satisfy the triangle inequality. The KL divergence between $p(i)$ and $q(i)$ is defined as

$$KL(p,q) = \sum p \log \frac{p}{q} \tag{5}$$

The cross entropy measures the overall difference between two probability distributions, which is defined as

$$H(p,q) = -\sum p \log q \tag{6}$$

It can be seen from the definitions of (6) and (7) that

$$H(p,q) = KL(p,q) + H(p) \tag{7}$$

where $H(p) = -\sum p \log p$ (8)

To make the KL measure and the cross entropy be symmetric measures, we modified them as:

$$KL'(p,q) = \sum p \log \frac{p}{q} + \sum q \log \frac{q}{p} \tag{9}$$

$$H'(p,q) = -\sum p \log q - \sum q \log p \tag{10}$$

so that the modified KL divergence and cross entropy have the following relation:

$$H'(p,q) = KL'(p,q) + H(p) + H(q) \tag{11}$$

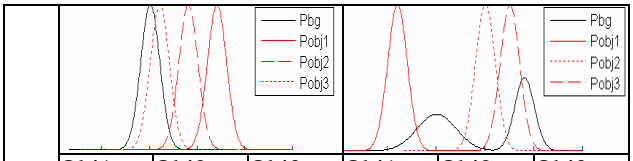
The last measure we evaluate is similar in form to mutual information, but it quantifies the distance between overall distribution p_{tot} and the product of $p(i)$ and $q(i)$:

$$I(p,q) = \sum p_{tot} \log \frac{p_{tot}}{p \cdot q} \tag{12}$$

[1] In practice, this function is modified to avoid dividing by zero or taking the log of zero. This modification is omitted here, for clarity.

To evaluate these measures of separability, we tested them on a series of generated distributions. Table 1 shows the performance of the four criteria in two sets of tests. In the first test, the means of two unimodal distributions (Gaussians) are brought closer and closer together. The distribution labeled P_{obj1} is most separable from the background distribution. When the means of the distributions of the object and background get closer, they become less separable. We see that the quantity of the extended variance ratio measure decreases faster (from 29.2 to 1.2) than the other three measures.

In the second test, the background is a bimodal distribution P_{bg}, and three unimodal Gaussians with different means are compared. Based on the extended variance ratio measure, P_{obj2} is judged the most separable from the bimodal background. Moreover, the difference between the separability scores of P_{obj1} and P_{obj2} is not very large, corresponding to visual intuition that both cases are equally separable from the background. However, based on the other three measures, P_{obj2} is rated worst among the three features, which is not correct since distribution P_{obj2} is clearly more separable from P_{bg} than P_{obj3} is.



	Obj1	Obj2	Obj3	Obj1	Obj2	Obj3
EVR	29.2*	10.6	1.2	5.2	6.2*	2.2
I	27.5*	11.8	4.11	28.2*	10.4	11.9
KL	63.9*	30.9	15.9	65.9*	30.5	33.0
H	56.4*	23.4	8.4	57.5*	22.0	24.5

Table 1: For each criterion, '*' represents the best feature choice (judged most separable by the measure) and the shaded one represents the worst feature choice (judged least separable).

During the simulation, many other cases also showed that the extended variance ratio criterion performed best for measuring separability of both unimodal and multimodal distributions [13]. We believe the reason is that the other three measures only evaluate the difference between two distributions, while ignoring the variance property of each individual distribution.

3. Divide and Conquer Approach

Many simple classification algorithms such as LDA assume the underlying class distributions are unimodal. However, when extracting a background histogram from the pixel neighborhood surrounding an object, we often get a multimodal distribution due to scene clutter. A second problem to address is nearby "confusors" having similar appearance to the foreground object. Such confusors typically have limited spatial extent, yet are highly likely to cause tracking failure. We believe that spatial reasoning is important for solving the problems of clutter and confusors. However, the necessary spatial information is discarded by the background histogram representation.

To solve this problem, we use a spatial divide and conquer approach, as illustrated in Figure 1. Like previous approaches, we select features based on the previous frame and use them to calculate the weight image of the current frame for tracking. However, first the object and the background in the previous frame are decomposed into smaller regions (cells). The idea is that the class distributions of these smaller spatial cells should be more easily separable. Guided by the extended variance ratio criterion, the best feature for each pairing of object cell and background cell is chosen. In other words, different features can be chosen for discriminating between different regions of the object and the background. Pérez et.al use a multi-region reference model for color based tracking [11]. In this paper we divide the background and foreground regions adaptively and recursively.

Features from each object-background cell pairing should produce weight images that discriminate well between the corresponding spatial regions of foreground and background. All weight images are then merged together to generate a single weight image that achieves good separation of the entire foreground and surrounding background. This weight image is used for mean-shift tracking.

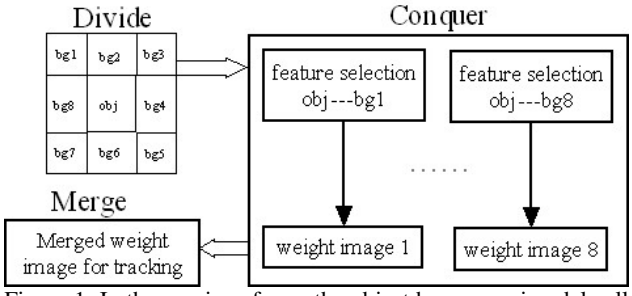
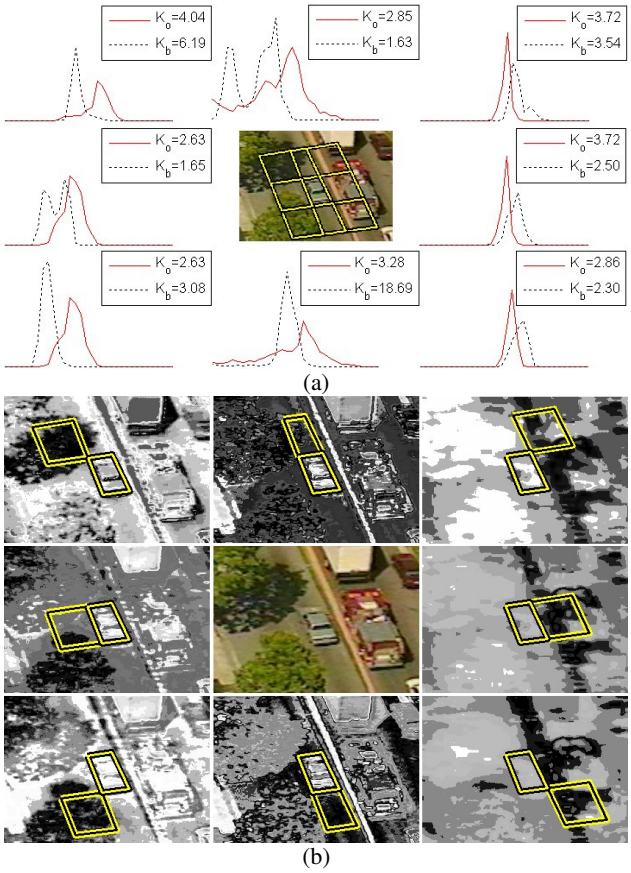


Figure 1: In the previous frame the object has one unimodal cell and the background is divided into 8 spatial cells. For each of the 8 pairings of object to background cells, the best feature is selected and a corresponding weight image is generated on the current frame. The 8 weight images are then merged together into a single weight image for tracking.

3.1. Divide and Conquer

Image segmentation could be used to break the object and the background into regions with unimodal feature distributions, but this approach would be too slow for an on-line tracking process. In fact, we only need to estimate the principal unimodal distributions in feature space, rather than analyzing the exact shape or contour of each background region. We therefore hypothesize that a coarse spatial decomposition is sufficient to provide resistance to background clutter and confusors.

Figure 2 illustrates the divide-and-conquer feature selection process. A grey car with a roughly unimodal distribution is tracked. There are many ways to spatially divide the background. In this example the background around the car is divided into 8 neighboring regions as shown in the center of Figure 2(a). Also shown are the object-background class distributions for the feature having highest extended variance ratio score for each cell.



(a)

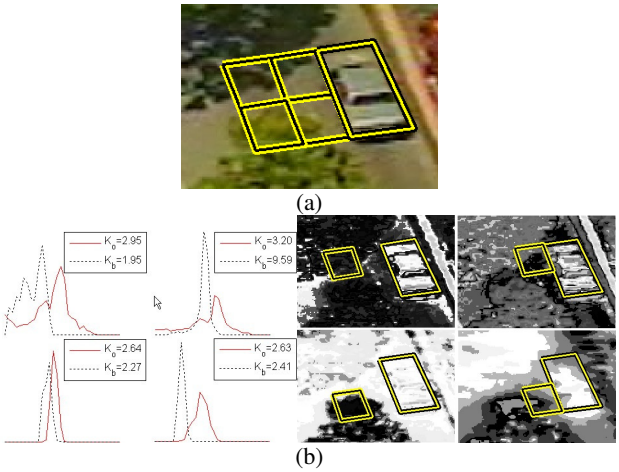
(b)

Figure 2: (a) The background around the car is divided into 8 spatial cells (the car has a roughly unimodal distribution without dividing). For each pairing of background region and object, a feature that generates the most separable class distributions is chosen. These class distributions are shown for each cell, along with the corresponding Kurtosis values. (b) The weight images based on the best feature selected for each cell.

In Figure 2(b), the corresponding weight images $g_i(x, y)$ for each cell are shown. These weight images are formed from the log likelihood values L (eq. 3) computed from the class distributions induced by the selected feature for each cell. If we were to threshold any weight image at zero, it would be equivalent to performing a binary classification of object from background at each pixel using the likelihood ratio test on the class conditional distributions.

From the weight images, we can see that the 8 selected features can separate the object from one corresponding background cell quite well (object region has high weight and background region has low weight), but that each feature does not guarantee good separation between the object and other background cells. For example, the feature in the lower left image in Figure 2(b) discriminates the car from its lower left background well, but does a poor job at distinguishing the car from the right-side background.

If the best feature for a cell in this initial dividing step results in unimodal class distributions, the feature is accepted. Otherwise we subdivide again until unimodal distributions are achieved. For example, the left-side background region in Figure 2 has a bimodal distribution even using the best feature, and is therefore further divided into four subregions as shown in Figure 3. We stop dividing when unimodal distributions are achieved or when the number of pixels in a region becomes too small.



(a)

(b)

Figure 3: (a) The left-side background region is divided into four sub-quadrangles. (b) Object/background feature distributions and corresponding weight images, based on the best features found for each subregion.

To decide if a distribution is suitably unimodal, we use the Kurtosis measure:

$$k = \frac{E[(x-\mu)^4]}{\sigma^4} \tag{13}$$

The kurtosis of the normal distribution is 3. Distributions that are sharper than the normal distribution have kurtosis value greater than 3; distributions that are flatter have kurtosis less than 3. We decide to further divide a region when the kurtosis value is smaller than 2. The EVR value can also be used to decide whether to continue dividing.

3.2. Merge

After obtaining the weight images for each object-background cell pair, we merge them together into a single weight image. Rather than just cut and paste

regions from each weight image, we smoothly interpolate according to:

$$W(x,y) = \sum c_i \cdot V_i \cdot \varphi_i(x,y) \quad (14)$$

where V_i is the area in pixels of the background region i, c_i is the extended variance ratio score, and $\varphi_i(x,y)$ is an "enhanced" weight image for the object/background pairing, as defined by:

$$\varphi_i(x,y) = \sum_{(x,y)\in obj \ \ or \ \ (x,y)\in bg_i} \alpha g_i(x,y) + \sum_{(x,y)\notin obj \ \ and \ \ (x,y)\notin bg_i} (1-\alpha) g_i(x,y) \quad (15)$$

$\alpha \in (0.5,1]$

The intuition behind enhanced weight images is that each weight image only discriminates well in the specific spatial cells that the weight image was derived from, thus should contribute relatively more to the output values of pixels in the same cell locations in the final merged weight image, and relatively less to pixels in other cells.

Figure 4 shows the merged weight image for tracking, using a reweighting parameter of $\alpha = 0.6$. The object is well distinguished from the entire local background surrounding it. We also show the result after smoothing with a Gaussian filter. It is known that mean-shift performs hill-climbing within this filtered space [4], thus the fact that there is a single strong mode at the location of the foreground object indicates that this is a good weight image for mean-shift tracking.

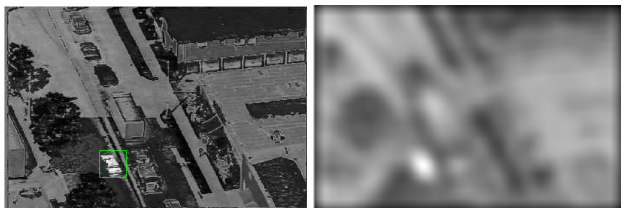


Figure 4: (left) Final merged weight image for tracking. The car is well-distinguished from the surrounding background. (right) Gaussian filtering of the weight image shows a single strong mode, indicating that mean-shift will successfully find the object location in this frame.

3.3. Different Spatial Dividing Methods

Different subdivisions of the scene background will generate different weight images for tracking. Figure 5 shows two alternative initial spatial divisions and their corresponding weight images. The top one considers only the foreground region versus the entire surrounding background, which is equivalent to the method of [5]. The bottom image subdivides the background into four regions corresponding to front, back, left and right, with respect to direction of object motion. We see that even this coarse division into four spatial regions yields a large improvement. Comparing with Figure 4 shows that increasing levels of subdivision yield even better weight images, although they are more costly in terms of computation time.

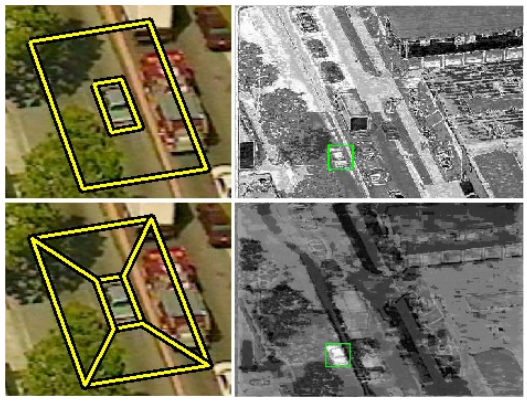


Figure 5: Alternative initial subdivisions. Top row: using a single surrounding background region. Second row: dividing the background into 4 pieces.

3.4. Motion Estimation

Motion estimation provides a powerful cue for detection and segmentation of moving objects. We use motion estimation for two tasks. First, we perform frame differencing to determine moving pixels from stationary scene background. This coarse motion segmentation is fused with the weight map produced by color appearance information to help track the object in camouflage situations and to avoid drift onto stationary background structures. Second, we perform constant velocity prediction of object motion based on previous location history to predict future object trajectory. The predicted location of the object in the next frame is used to center the grid of spatial cells for weight image construction in that frame, and to form a "gating" region [3] within which to search for the object. The predicted future object trajectory also helps track through temporary occlusions.

However, we consider videos where the camera can be moving, and both constant velocity trajectory prediction and frame differencing are invalid under camera motion. For this reason, we use estimates of frame-to-frame background motion to compensate for apparent camera motion when applying constant velocity motion prediction, and to perform stabilization when segmenting motion via frame differencing.

Two-frame background motion estimation is performed by fitting a global parametric motion model (affine or projective) to sparse optic flow. Sparse flow is computed by matching Harris corners between frames using normalized cross correlation. A good set of potential matches is founding using "marriage-based" compatibility tests, as described in [10]. Given a set of potential corner correspondences across two frames, we use a Random Sample Consensus (RANSAC) procedure [7] to robustly estimate global affine flow from observed displacement vectors. The largest set of inliers returned from the RANSAC procedure is then used to fit either a 6

parameter affine or 8 parameter planar projective transformation.

The object motion prediction module assumes the object travels with constant velocity within three consecutive frames after first compensating for the background motion. It provides a rough location for initially subdividing the object and background. The tracked object can move with varying speed and directions in the real world. For example, assuming an affine model of background motion, the formula to predict the location *Pt* of the object in the current frame *t*, given its previously observed positions *Pt-1* and *Pt-2* in the last two frames, is

$$P_t = T_t^{t-1} * \left[P_{t-1} + \left(P_{t-1} - \left(T_{t-1}^{t-2} * P_{t-2}\right)\right)\right] \quad (16)$$

where T_{t-1}^{t-2} is the affine motion between frames *t-2* to *t-1*, and T_t^{t-1} is the affine motion between frames *t-1* and *t*.

Sample results of object location prediction and motion detection are illustrated in Figure 6. Motion detection weights (Figure 6c) and the appearance weight image (Figure 6d) are fused to generate a better weight image for tracking (Figure 6e). Currently, a logical "and" operation is used for fusion such that a pixel in Figure 6e is valid only when the pixel value is larger than some threshold and motion exists at that pixel.

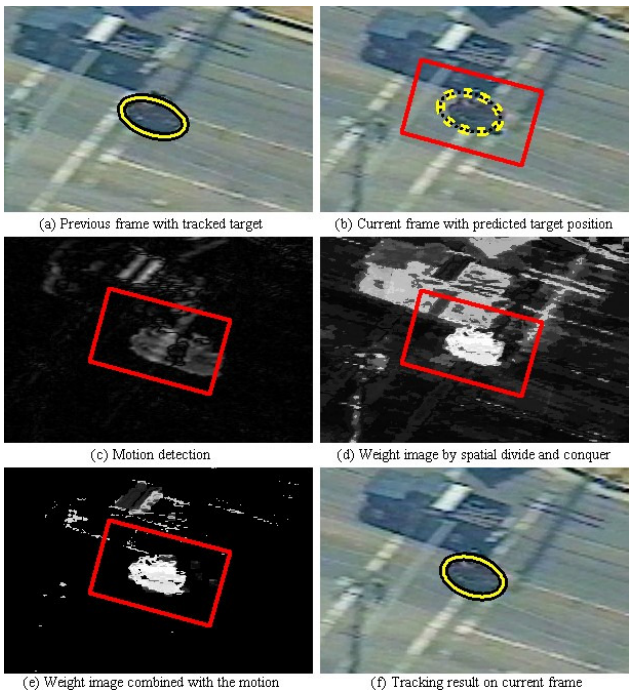


Figure 6: (a) previous frame. (b) current frame. The predicted object location is showed as a dashed ellipse. The search region is the enlarged elliptical area around the predicted location. (c) motion mask computed by frame difference between the current frame and the warped previous frame. (d) appearance weight image by divide and conquer approach. (e) weight image fused with the motion information. (f) mean-shift tracking result on the joint color-motion weight image.

3.5. Tracking Algorithm

Due to the continuous nature of video, the distribution of the object and the background should remain almost the same between two successive frames. Thus it is reasonable to select features from the previous frame and apply them to the current frame to generate a weight image for tracking. The object is then localized in the weight image using mean shift, and the orientation of the object is determined by weighted ellipse fitting. We incorporate the divide and conquer feature selection approach into the following tracking algorithm:

Initialization:

1) Manually select foreground object in the first frame.

2) Automatically divide the object/background into small regions and select the best feature for each object-background pair. If the features distributions of object and background satisfy the Kurtosis threshold or the object/background region is too small, the feature is accepted and subdivision is stopped. Otherwise, continue to divide and conquer the feature selection.

Tracking:

For subsequent frames, do:

1) Predict the target position using background motion estimation and constant velocity foreground motion prediction.

2) Generate weight images from cell layout around the predicted location using the best selected features assigned to the different spatial regions in the previous frame. Merge the weight images together to generate an appearance weight image.

3) Compute a motion detection image using motion compensated frame differencing between the warped previous frame and current frame.

4) Use mean shift to find the nearest local mode in a combined weight image containing both appearance generated weights (from step 2) and motion detection weights (from step 3).

5) Fit an ellipse centered on the object location determined by mean-shift to get the orientation and scale of the object for the next round of divide and conquer.

6) Update the spatial cell layout and selected tracking features from the current frame using the divide and conquer approach.

Note that step 4 uses both appearance-based and motion-based weights. This further improves tracking performance by ensuring that the tracker is not distracted by stationary regions in the background, regardless of how similar they may appear to the tracked object. The joint color-motion weight image also keeps the tracker away from distracting motion near the target.

4. Experiments

In this section we present three challenging tracking examples that illustrate the benefits of spatial divide and conquer with motion cues for tracking through clutter.

The first aerial video shows a grey car chased by the police, from the VIVID Tracking Evaluation Website (http://www.vividevaluation.ri.cmu.edu/). During the chase, the car passes cars of similar color, is partially occluded by traffic signs, and passes through shadows. Camera motion is also prominent, including scale changes and rotation. Figure 7 shows sample frames from the sequence. The first row shows the tracked object in the original image. The second row shows weight images generated by the divide and conquer approach. Spatial decomposition as described above is used to minimize the appearance of clutter and distractions in the weight image. The third row illustrates motion detection results. Note that motion detection can add new clutter, as shown in the last column of Figure 7 where two cars move in parallel. However, using both the appearance weight image and motion detection results, the tracker succeeds.

The second video is from the OTCBVS Benchmark Dataset Collection (http://www.cse.ohio-state.edu/otcbvs-bench/). Figure 8 shows the tracking result with the weight images and motion results. The object being tracked is a pedestrian walking from sunshine into a moving cloud shadow. The pedestrian also becomes partly occluded and separated into pieces by a sculpture. Nonetheless, the pedestrian is successfully tracked until they leave the view. Note that the pedestrian gait and body shape is very clear in the weight images.

The third experiment is an aerial video sequence of a motorcycle (Figure 9). The motorcycle is small and fast moving, and often passes vehicles with similar colors. Sometimes the object is running through or passing by shadows. Sometimes it is partial occluded by traffic signs or nearby vehicles. Again, we see that the weight images produced by the divide and conquer approach are quite good for distinguishing the motorcycle from its surrounding background, and that motion detection results alone would not succeed, due to motion clutter.

5. Summary

This paper presents a divide and conquer approach to consider the spatial layout of background clutter with respect to the foreground object. Guided by an extended variance ratio criterion for determining separability of distributions, features are selected that discriminate between spatial pairs of object and background cells. Weight images with good discriminative quality are generated for each cell, and merged to produce a single weight image for mean shift tracking. A motion prediction module allows for reduced object search regions and motion detection via compensation of background camera motion. The mean-shift procedure thus considers both appearance and motion based weight images to determine object location in a new frame. After finding the location of the object, weighted ellipse fitting is executed to find the object orientation. The experiments show good performance on video scenes containing spatial clutter, distractions and camouflage.

Future work will more directly address the problem of drift during object appearance adaptation. Although spatial appearance and temporal motion information are combined in the present algorithm, we still need to evolve the object model carefully to avoid drift. Future work will focus on imposing shape-guided foreground/background segmentation into the tracking process, to avoid model drift during adaptive tracking.

Acknowledgements

This work was funded under the NSF Computer Vision program via grant IIS-0535324 on Persistent Tracking.

References

[1] S.Avidan, “Ensemble Tracking”, IEEE Conf on Computer Vision and Pattern Recognition (CVPR), 2005,pp. 494-501.

[2] S.Baker and I.Matthews, “Lucas-Kanade 20 Years On: A Unifying Framework,” International Journal of Computer Vision, 56(3):221-255, March, 2004.

[3] S.Blackman and R.Popoli, Design and analysis of Modern Tracking Systems, Artech House, Norwood MA., 1999.

[4] Y.Cheng, “Mean Shift, Mode Seeking, and Clustering,” IEEE Trans on Pattern Analysis and Machine Intelligence, 17(8):790-799, August 1995.

[5] R.Collins, Y.Liu and M.Leordeanu, “Online Selection of Discriminative Tracking Features,” IEEE Pattern Analysis and Machine Intelligence, 27(10):1631-1643, Oct 2005.

[6] D.Comaniciu, V.Ramesh, and P.Meer, “Kernel-based Object Tracking,” IEEE Transactions Pattern Analysis and Machine Intelligence, 25(5):465-575, May 2003.

[7] M.Fischler and R.Bolles,“Random Sample Consensus”, Communications of the ACM, 24(6):381-395, 1981.

[8] M. Isard and A. Blake, “CONDENSATION - Conditional Density Propagation for Visual Tracking”, International Jounal of Computer Vision, 29(1):5-28, 1998

[9] A.Karr, Probability (Springer Texts in Statistics), Springer Verlag, 1996, ISBN: 3540940715.

[10] D.Nister, O. Naroditsky, and J.Bergen, “Visual Odometry,” IEEE Conf on Computer Vision and Pattern Recognition, 652-659, June 2004.

[11] P. Pérez, J. Vermaak, A. Blake. Data fusion for visual tracking with particles. Proc. IEEE, 92(3):495-513, 2004.

[12] H.Tao, H. Sawhney and R. Kumar, “Object Tracking with Bayesian Estimation of Dynamic Layer Representations”, IEEE Transactions Pattern Analysis and Machine Intelligence, 24(1): 75-89, January 2002

[13] Z. Yin and R. Collins, “Spatial Divide and Conquer with Motion for Tracking”, Tech Report CSE-05-009, Penn State University, 2005

Figure 7: A car passes similar cars, partially occlusions, moves through shadows and turns a corner.

Figure 8: A pedestrian walks into a moving shadow and is partially occluded by a sculpture.

Figure 9: A fast-moving motorcycle passes vehicles of similar color, partial occlusions, and moves through shadows.