MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Achieving Real-Time Object Detection and Tracking Under Extreme Conditions

Fatih Porikli

TR2006-039 August 2006

Abstract

In this survey, we present a brief analysis of single camera object detection and tracking methods. We also give a comparison of their computational complexities. These methods are designed to accurately perform under difficult conditions such as erratic motion, drastic illumination change, and noise contamination.

Journal of Real-Time Image Processing

Achieving Real-Time Object Detection and Tracking Under Extreme Conditions

Fatih Porikli

***Abstract*— In this survey, we present a brief analysis of some of the recent single camera object detection and tracking methods that are developed to function robustly under extreme conditions such as erratic object and camera motion, drastic illumination changes, and severe noise contamination while achieving a real-time performance.**

I. Introduction

OBJECT tracking is one the most important tasks in computer vision. In video surveillance, it is used to understand movement patterns of people to uncover suspicious events. It is a key component in real-time traffic management to estimate vehicle motion statistics and congestion status. Advanced vehicle control systems depend on the tracking information to keep the vehicle in lane and prevent from collisions. In medical field, tracking helps improving the quality of life for physical therapy patients and disabled people. In retail space instrumentation, it is used to advance architecture design by analyzing the shopping behavior of customers. In addition, it provides comprehensible visual information to attain environmental awareness in robotics. In video summarization, it is applied to generate object-based representations and automatic annotations of the content. Tracking is also a fundamental technology to extract regions of interest and video object layers defined in JPEG-2000 and MPEG-4 standards.

As essential as it is in many applications, robust object tracking under uncontrolled conditions still poses a challenge. Practical systems have to track objects when the lighting condition changes suddenly, the relative camera-object motion becomes large, the color contrast becomes low, and the image noise arises to an intolerable level. To make everything worse, the computational complexity is required to be kept at a minimum level to achieve real-time performance.

In the rest of this paper, we briefly describe just a few of the state-of-art methods designed to address these difficulties.

II. Object Detection

Background subtraction is a common approach for discriminating moving regions in fixed camera setups. Basically, a pixel-wise reference model for the stationary part of the scene is estimated. Then, the observed image is compared with this reference to obtain the foreground.

Simply, a reference frame can be computed by aggregating the previous frames in a moving temporal window by α-blending. Although this approach has a minimum computational cost, it induces ghost effects and rarely works in real-life circumstances where the background is often nonstationary due to the illumination changes, shadows, swaying trees, etc. Alternatively, predictive techniques such as Kalman [1] and Wiener [2] filters are applied to learn the underlying pixel intensity distribution. These techniques strongly depend on the preset state transition parameters and fail in case the distribution does not fit into a single model or varies randomly.

To handle such multimodal backgrounds, mixture of models that are flexible enough to handle variations in lighting, moving scene clutter, multiple moving objects and other arbitrary changes to the observed scene, are proposed. In addition to color, there are variants that accommodate gradient and optical flow information [3]. Often, these models are assigned as Gaussian functions. Online approximations such as expectation maximization (EM) algorithm are used to update the models [4]. However, online EM update tends to intermingle weak modes into stronger modes, thus, distorts the model means in the long term as shown in Fig. 5. To adapt the models accurately, we developed a Bayesian update mechanism [5] that can also estimate the number of required layers. Another approach that models the background distribution is the non-parametric kernel density estimation [6]. This method keeps the color values of the multiple frames and estimates the density function using all the available data instead of online approximation. Proportional to the number of frames, both memory and computational costs become prohibitive to adapt this method for real-time applications. Although the mixture of models approach can converge to any arbitrary distribution provided enough number of components, its computational complexity boosts exponentially as the number of models increases.

A major shortcoming of the above methods is that they all neglect the temporal correlation among the color values. This prevents them from detecting a structured or periodic change, which is often the case, since real-world physics often induces near-periodic phenomenon in the environment: the motion of plants driven by wind, the action of waves on a beach, and the appearance of rotating objects. To distinguish such periodic motion from the object's motion, we proposed a frequency decomposition based representation of the background, *wave-back* [7]. This algorithm detects new objects based solely on the dynamics of the pixels in a scene rather than their appearance. This is accomplished by directly estimating models of cyclostationary processes to explain the observed dynamics of the scene and then comparing new observations against those models. For a given frame, we compute the frequency coefficients and compare them to the background coefficients to obtain a distance map as in Fig. 4.

Detection of the time-varying phenomenon is also attempted using corner-based background models [8]. First, they detect feature points using a corner detector and represent them

TABLE I
PERFORMANCE OF BACKGROUND BASED OBJECT DETECTION

	speed*	LA†	PB†	MM†	NH†
α-Blend	5	-	-	-	-
Kalman [2]	8	+	-	-	+
GMM-EM‡ [4]	45	+	-	++	++
GMM-Bayesian‡ [5]	35	+	-	$\star$	++
Wave-back [7]	55	++	$\star$	-	-
Corner [8]	150	+	++	-	++
Intrinsic [9]	30	$\star$	++	-	$\star$

(*) msec, 320×240 color image on P4 3Ghz. (‡) 3 independent models. (†) LA: lighting adaptation, PB: periodic backgrounds, MM: multi-modal backgrounds, NH: noise handling.

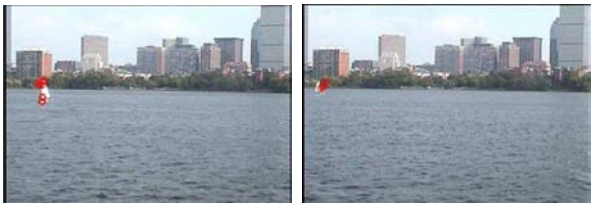

Fig. 1. Detection results by corner based background modeling [8].

as SIFT-like descriptors. Second, they dynamically learn a background model and classify each extracted feature as either a background or a foreground feature. Last, a "Lucas-Kanade" feature tracker is integrated to differentiate motion consistent foreground objects from background objects with random or repetitive motion. The key insight is that a collection of SIFT-like features can represent the environment and account for variations caused by natural effects with dynamic movements as shown in Fig. 1. Unlike the previous pixel-based techniques, this approach evaluates the background only on the corner points, as a result, it fails to detect changes if no corner point exists.

Instead of adapting models to the background, and trying to solve the issues arises due to the model fitting, it is also possible to represent the scene in terms of the multiplication of a static part and and a dynamic part. For this purpose, we decompose a scene into time-varying multiplicative backgrounds and foregrounds using the intrinsic images idea [9]. We form a set of previous images by adaptive temporal sampling and compute the spatial gradients of these images. By taking advantage of the sparseness of the filter responses, we estimate the background using the median filtered gradients as in Fig. 3. This method is very robust to sudden and severe illumination changes that a scene may undergo as shown in Fig. 2. It is also computationally feasible to implement into a real-time system.

We summarized the properties of several detection methods in table I. The following techniques are often employed to improve the speed and robustness:

- Partial update of the background models
- Assuming color channels are independent
- Limiting the temporal size of detectable variations
- Color space transformations to decrease sensitivity
- Using a single color channel

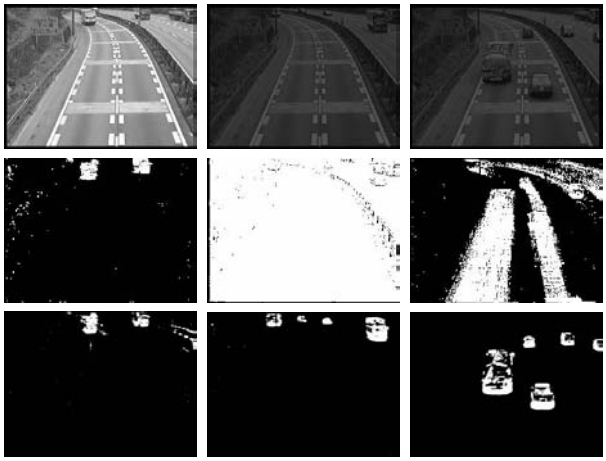

Fig. 2. Comparison of GMM [4] and intrinsic [9] background modeling. **Top:** Sudden illumination change happens. **Middle:** GMM method confuses and its recovery takes time. **Bottom:** Intrinsic background is not disturbed. Both methods use the RGB color space.

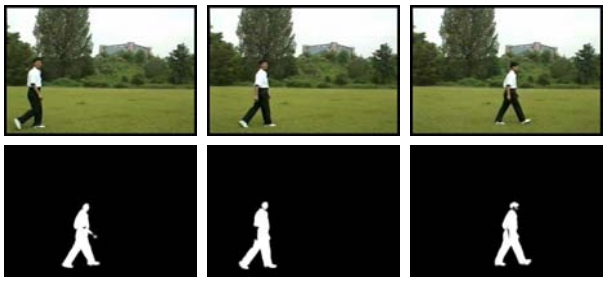

Fig. 3. Detected foreground regions by intrinsic backgrounds.

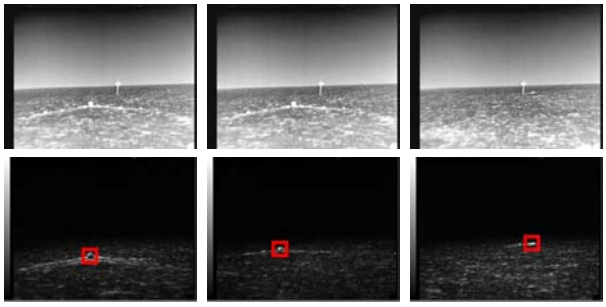

Fig. 4. Object detection in thermal IR that depicts a sea shore. The wave-back [7] method can distinguish periodic motion of the sea waves from the motion of the boat. *(Courtesy of PETS 2005)*

Fig. 5. EM update [4] vs. Bayesian update [5]. **Left:** Sample frames. **Middle:** First (top) and second (bottom) background layers of EM update. **Right:** Bayesian update result. EM update inaccurately blends distinct modes into identical layers. Bayesian update can identify separate layers, i.e. road and shadow modes. (Red means no layer)

III. Inter-Frame Tracking

Tracking, that is finding a region corresponding to a given object in the image, also faces similar challenges. Objects frequently change their appearance and pose. They occlude each other, become temporarily hidden, merge and split. Depending on the application, they exhibit erratic motion patterns and often make sudden turns.

Tracking can be considered as estimation of the state given all the measurements up to that moment, or equivalently constructing the probability density function of object location. A common approach is to employ predictive filtering and use the statistics of object's color and location in the distance computation while updating the object model by constant weights [10]. When the measurement noise are assumed to be Gaussian, the optimal solution is provided by the Kalman filter [11]. When the state space is discrete and consists of a finite number of states, Markovian filters can be applied for tracking. The most general class of filters is represented by particle filters, which are based on Monte Carlo integration methods. The current density of the state (which can be location, size, speed, boundary [12], etc.) is represented by a set of random samples with associated weights and the new density is computed based on these samples and weights. Particle filtering is a popular tracking method [13],[14]. However, it is based on random sampling that becomes a problematic issue due to sample degeneracy and impoverishment, especially for higher dimensional representations.

In contrast, the mean-shift tracker is a non-parametric density gradient estimator that is iteratively executed within the local search kernels [15]. It models the object probability density in terms of color histogram, and moves the object region towards the largest gradient direction. Thus, it is computationally simple. Still, if the object relocation between successive frames is larger than the kernel size, it simply fails. Since the histograms are used to determine likelihood, the gradient estimation becomes inaccurate in case object and background color distribution are similar. To solve this issue, we assign multiple mean-shift kernels to the motion regions that are obtained by background subtraction [16]. In order to improve the convergence in case the object resembles to the background, we associate an additional weight that pushes the kernel away from the regions similar to the background. In conjunction, a second weight derived from object template pulls the kernel towards the similar regions.

Tracking can also be considered as a classification problem and a classifier can be trained to distinguish the object from the background [17]. This is done by constructing a feature vector for every pixel in the reference image and training a classifier to separate pixels that belong to the object from pixels that belong to the background. Integrating classifiers over time improves the stability of the tracker in cases illumination changes. As in the mean-shift, an object can be tracked only if its motion is small. One obvious drawback of the local search methods is that they tend to stuck into local optimum.

Object representation, that is how to convert color, motion, shape, and other properties into a compact and identifiable form is a major concern. Most trackers either depend only on the color distributions such as histograms that disregard the structural arrangement of colors, or appearance models that ignore the statistical properties. Populating higher dimensional histograms by a small number of pixels within the window poses a major problem. Besides, histograms are easily contaminated by image noise. Appearance models, on the other hand, are sensitive to the scale changes and they tend to decay rapidly if the object localization is not accurate.

TABLE II
Performance of Inter-Frame Tracking

	speed*	AC†	EM†	FM†	SO†
Predictive [10]	10	+	-	-	-
Mean-shift [15]	12	+	+	-	-
Multi-kernel [16]	20‡	+	++	++	-
Particle [13]	25	++	+	-	-
Ensemble [17]	20	⋆	+	-	-
Covariance [19]	150	++	++	⋆	+

(*) msec, 20×40 single object in 320×240 color image.
(†) AC: appearance changes, EM: erratic motion, FM: fast motion, SO: small objects. (‡) Exponentially increases with object number.

Covariance matrix representation [18] embodies both spatial and statistical properties of objects, and provides an elegant solution to fusion of multiple features. Covariance is an essential measure of how much the deviation of two or more variables or processes match. In our case, these variables correspond to point features such as coordinate, color, gradient, orientation, and filter responses. This representation has much lower dimensionality than histograms. It is robust against noise and severe lighting changes as well. To track objects, we apply an eigenvector based distance metric to compare the covariance matrices of object and candidate regions and incorporate a Lie algebra based update mechanism to adapt to temporal variations [19]. Covariance tracker does not make any assumption on the motion. It is not limited to a maximum speed. This means that it can keep track of objects even if their motion is erratic and fast. It can compare any regions without being restricted to a constant window size.

In spite of this advantages, computation of the covariance matrices for all possible rectangular regions within a given image is computationally prohibitive using the conventional methods. Thus, we adapted an integral image based algorithm that requires constant time [20]. This technique significantly improves the computational load of the covariance matrix extraction process by taking advantage of the spatial arrangement of the points.

IV. Future Directions

To achieve real-time performance under uncontrolled conditions, there remains need for algorithmic improvement:

- Time spent in the computation of the likelihood between the object and candidate regions is a bottleneck. Tracking methods that use histograms become more demanding as the histogram bin size increases. Some histogram distance metrics (Bhattacharya, KL) are inherently expensive. For the covariance tracker, computing eigenvectors is also costly. Fast computation of the distance norms will directly improve the tracking speed.

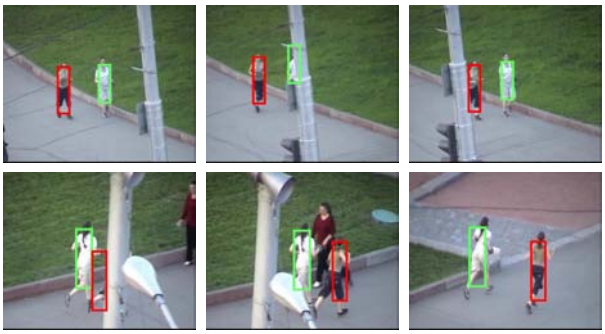

Fig. 6. Covariance tracker [20] results for temporary occlusion.

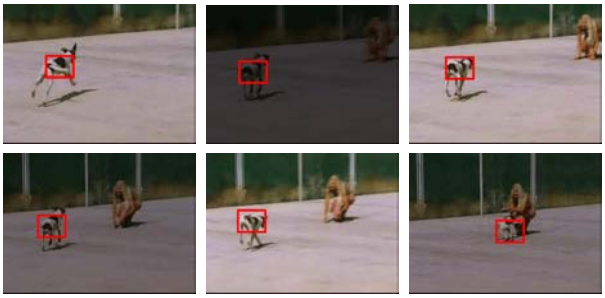

Fig. 7. Covariance tracker [20] results for severe illumination change that is manually generated by changing the intensity. Intensity is not discarded, in contrast, the RGB color space is used.

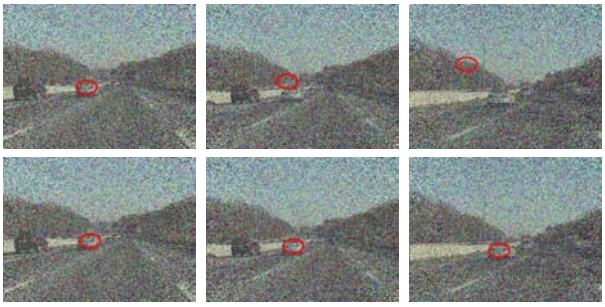

Fig. 8. Noise performance for the frames 1 (left), 40 (middle), and 200 (right). **Top:** Mean-shift tracker [16] using color histogram. **Bottom:** Covariance tracker [20] using 7 features. Almost 95% of pixels are distorted by noise.

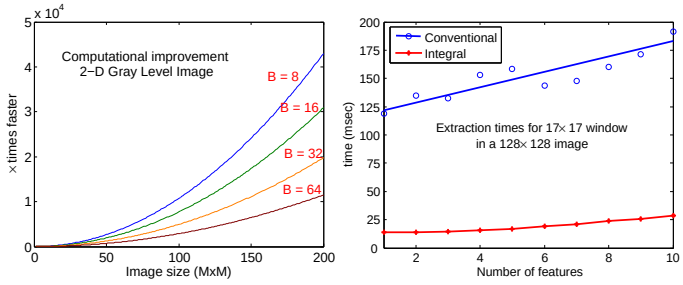

Fig. 9. **Left:** Extraction of histograms is accelerated thousands time using the integral histogram method [20]. **Right:** Covariance extraction speed also improves by using the integral images.

- Complexity is proportional to the number of the candidate regions to be tested. It may be possible to apply hierarchical search methods as widely accepted in motion estimation to accelerate the search process.
- Localized search methods such as mean-shift and ensemble tracking becomes slow by the increasing object size. Adaptive switching down to a lower resolution in which the object properties are still distinguishable may provide expedite convergence of those methods.

REFERENCES

[1] K.-P. Karman and A. von Brandt, "Moving object recognition using an adaptive background memory," in *Time-varying Image Processing and Moving Object Recognition*, Capellini, Ed., vol. II. Amsterdam, The Netherlands: Elsevier, 1990, pp. 297–307.

[2] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, "Wallflower: Principles and practice of background maintenance," in *Proc. 7th Intl. Conf. on Computer Vision*, Kerkyra, Greece, 1999, pp. 255–261.

[3] K. Javed, O. Shafique and M. Shah, "A hierarchical approach to robust background subtraction using color and gradient information," in *IEEE Workshop on Motion and Video Computing*, 2002.

[4] C. Stauffer and E. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Fort Collins, CO, vol. II, 1999, pp. 246–252.

[5] F. Porikli and O. Tuzel, "Bayesian background modeling for foreground detection," in *Proc. of ACM Visual Surveillance and Sensor Network*, 2005.

[6] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction," in *Proc. European Conf. on Computer Vision*, Dublin, Ireland, vol. II, 2000, pp. 751–767.

[7] F. Porikli and C. Wren, "Change detection by frequency decomposition: Wave-back," in *Proc. of Workshop on Image Analysis for Multimedia Interactive Services*, Montreux, 2005.

[8] Q. Zhu, S. Avidan, and K. Cheng, "Learning a sparse, corner-based representation for time-varying background modelling," in *Proc. 10th Intl. Conf. on Computer Vision*, Beijing, China, 2005.

[9] F. Porikli, "Multiplicative background-foreground estimation under uncontrolled illumination using intrinsic images," in *Proc. of IEEE Motion Multi-Workshop*, Breckenridge, 2005.

[10] C. Wren, A. Azarbayejani, T. Darell, and A. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, pp. 780–785, 1997.

[11] Y. Boykov and D. Huttenlocher, "Adaptive bayesian recognition in tracking rigid objects," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Hilton Head, SC, vol. II, 2000, pp. 697–704.

[12] M. Isard and I. Blake, "Condensation – conditional density propagation for visual tracking," in *Intl. J. of Computer Vision*, vol. 29, 1998, pp. 5–28.

[13] N. Bouaynaya, W. Qu, and D. Schonfeld, "An online motion-based particle filter for head tracking applications," in *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Philadelphia, 2005.

[14] S. Zhou, R. Chellappa, and B. Moghaddam, "Visual tracking and recognition using appearance-based modeling in particle filters," in *Proc. Intl. Conf. on Multimedia and Expo.* Baltimore, 2003.

[15] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Hilton Head, SC, vol. 1, 2000, pp. 142–149.

[16] F. Porikli and O. Tuzel, "Object tracking in low-frame-rate video," in *Proc. of PIE/EI - Image and Video Communication and Processing*, San Jose, CA, 2005.

[17] S. Avidan, "Ensemble tracking," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, San Diego, CA, 2005.

[18] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification," in *Proc. 9th European Conf. on Computer Vision*, Graz, Austria, 2006.

[19] F. Porikli, O. Tuzel, and P. Meer, "Covariance tracking using model update based on lie algebra," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, New York, NY, 2006.

[20] F. Porikli, "Integral histogram: A fast way to extract histograms in cartesian spaces," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, San Diego, CA, vol. 1, 2005, pp. 829 – 836.