Proceedings of the 2004 IEEE
International Conference on Robotics & Automation
New Orleans, LA • April 2004

A Region-Based Background Modeling and Subtraction Using Partial Directed Hausdorff Distance

Shih-Shinh Huang[1], Li-Chen Fu[1,2], and Pei-Yung Hsiao

Department of Computer Science and Information Enginnering[1],
Department of Electrical Enginnering[2],
National Taiwan University, Taipei, Taiwan, R.O.C.

***Abstract*— This paper presents a region-based approach for background modeling to economize the use of space. In order to be immune to noise and changes resulting from illumination or motion, the partial directed Hausdorff distance is adopted while subtracting the foreground objects from the scene robustly. Instead of determining the threshold values manually, we use an adaptive method to automatically choose the threshold values. Finally, experimental results of applying our approach on a sequence of an indoor scene are provided to demonstrate the effectiveness of the proposed method.**

I. Introduction

Detecting foreground regions from static scene is an important and essential technique applied in many visual applications, such as video monitoring system, intelligent traffic monitoring system, intrusion surveillance, airport safety, etc. It was firstly proposed in 1988 [13] to segment foreground objects from static background scene.

In the simplest approach [7], the background model is considered as the long-term average image. As compared with the background model, the pixels that have great color difference are taken as the foreground pixels. However, this method will result in considerable false detection due to changes or noise in environment.

A. Related Works

In case of traffic surveillance, Fridman [6] modeled the pixel intensity as a weighted mixture of three Gaussian distributions corresponding to road, vehicle, and vehicle distribution. An incremental version of the EM algorithm was used to learn and update the parameters of the Gaussian mixture models. In [4], [5], a nonparametric background model and the background subtraction process based on nonparametric kernel density estimation were proposed to handle the situations where the background is non-static and contains small motions. In 2000, Haritaoglu and Harwood [8] modeled the background scene by representing each pixel by its minimum and maximum intensity values and the maximum intensity difference. Another appraoch that represents the color of each pixel by a group of clusters was proposed in [1] to adapt to noise and background variation.

These proposed methods are pixel-based approaches that represent each pixel by a individual model. The models for different pixels are independent of one another (no context). Without taking the contextual information into consideration, the pixel-based approaches have a tendency toward false detection even applying some sophisticated methods when changes or noise occur.

For the sake of overcoming this shortcoming, block-based approaches have been used for modeling the background . In [10], each block was represented by its median template and block standard deviation. Moreover, blocks with too much difference relative to its template are considered as foreground. However, the major drawback of block-based approaches is that they are only suitable for coarse detection.

In this paper, we propose a region-based approach to model the background scene as a set of regions. And, each region contains only small number of homogeneous colors. By modeling background this way, we can reduce false detection successfully without sacrificing any benefits from pixel-based approaches.

B. Organizations

The remainder of this paper is organized as follows. In section 2, we describe how to segment the background image to a set of regions. In section 3, the means for region representation and the algorithm for automatically determining threshold values is proposed. In section 4, the partial directed hausdorff distance is introduced to extract foreground objects from the background scene. Finally, we conclude this paper with some experimental results and conclusions.

II. Color Segmentation

The color segmentation algorithm that divides the background scene into a set of regions is the first step in region-based background modeling. In the first place, colors in the image are quantized without degrading the color quality by using the k-means clustering algorithm. After quantization, the two passes algorithm described by Rosenfeld and Pfaltz [11] is used to find the connected components. The result is taken as the initial input for the J color segmentation algorithm.

0-7803-8232-3/04/$17.00 ©2004 IEEE

A. Criteria

We define a segmentation $SEG^I = \{R^1, R^2, .., R^m\}$ of an image I as a set of mutually exculsive regions that compose the entire image. In other words, $R^i \bigcap R^j = \emptyset$ for $i \neq j$ and $\bigcup_{i=1}^{m} R^i = I$. In our application, we expect that each region $R^i \in SEG^I$ resulting from segmentation should have the following properties:

- The number of quantized colors contained in a single region should be a few.
- The pixels that have the same quantized color in a segmented region should distribute over the region uniformly.

The purpose for imposing these two properties to the color segmentation algorithm is aimed at finding a set of regions that can represent the background scene in an appropriate manner. In the next subsection, we will introduce a quantitative J measure for the color segmentation algorithm called J segmentation algorithm. The regions resulting from the J segmentation algorithm would well suit to the proposed properties.

B. J Measure

The idea of J measure is the same as Fisher's linear discriminant [3] [2]. It measures the distances between different classes S_B against the summation of the distances among the members within each class S_W. J measure will be explicitly defined in the following. Let R be a region that contains N_R two-dimensional pixels. And, $m = (m_x, m_y)$ is the mean of all N_R pixels in the region R, i.e.,

$$m = \frac{1}{N_R} \sum_{p \in R} p \tag{1}$$

Suppose that all N_R pixels in region R are classified into T classes, R_i, $i = 1, 2, ..., T$. And, let m_i, $i = 1, 2, ..., T$ be the mean of the N_{R_i} pixels in class R_i, respectively, i.e.,

$$m_i = \frac{1}{N_{R_i}} \sum_{p \in R_i} p \tag{2}$$

The total distance S_T within the region R is defined as

$$S_T = S_W + S_B = \sum_{p \in R} (||p - m||)^2 \tag{3}$$

and the within class distance is

$$S_W = \sum_{i=1}^{T} \sum_{p \in R_i} (||p - m_i||)^2 \tag{4}$$

A higher J indicates that the pixels with different color are more separated from each other. On the other hand, J is low if the pixels with different color are uniformly distributed over the region.

The segmentation measure J_{SEG} for a segmentation $SEG = \{R^1, R^2, ..., R^m\}$ is the weighted sum of J for each region $R^i \in SEG$.

$$J_{SEG} = \sum_{i=1}^{m} N_{R^i} \times J_{R^i}, \tag{5}$$

where N_{R^i} is the number of the pixels in the region R^i, and J_{R^i} is the J measure for the region R^i, respectively. This measure will be used to determine the more suitable number of regions for the J segmentation algorithm.

C. J Segmentation Algorithm

The major idea for color segmentation proposed in this paper is to merge two adjacent regions into a large one until the number of the regions is smaller than a desired value m. At first, each connected component generated from Connected-Component-Finding algorithm is taken as a single region.

Then, we sort all regions according to their size in ascending order. The smallest region is selected and merged with the adjacent region that will result in the lowest J value. The great details of the color segmentation algorithm is shown in Algorithm 1.

Algorithm 1 J Segmentation Algorithm

```
Input ($m$): the expected number of regions
        in the resulted segmentation.
$S^0 \leftarrow$ Connected-Component-Finding($I_B$)
$i \leftarrow 0$
while $|S^i| \geq m$ do
    $R_{min} \leftarrow \arg\min_{R_i} \{|R_i|, R_i \in S^i\}$
    $R_{nei} \leftarrow \arg\min_{R_i} \{J(R_i \bigcup R_{min}),$
          $R_i \in NEIGHBOR(R_{min})\}$
    $S^{i+1} \leftarrow S^i - \{R_{min}, R_{nei}\} \bigcup \{R_{min} \bigcup R_{nei}\}$
end while
```

The next coming problem is how to determine the most suitable value m. In the following, we propose a systematic way to determine it. The process starts by repeatedly invoking the J color segmentation algorithm with different input values m ranging from 1 to 100 to obtain the different segmented image $SEG(m)$. Figure 1 shows the plot of segmentation measure $J_{SEG(m)}$ versus the value m, the number of regions in segmentation $SEG(m)$.

From Figure 1, the segmentation measure $J_{SEG(m)}$ is inversely proportional to the number of regions m. For example, in the extreme over-segmentation case where colors of all pixels in a region are the same, the segmentation measure is equal to zero. On the contrary, the segmentation measure will be large when we merge all regions into a single region.

Let the above 100 segmentation measure values be one-dimensional samples, $x_1, x_2, ..., x_{100}$ in order. The global standard deviation σ_g for all samples and local standard deviation

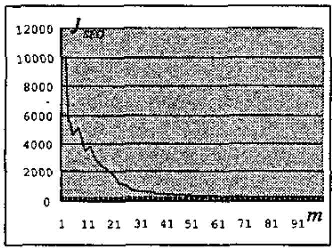

Fig. 1. The chart shows a plot of segmentation measure $J_{SEG(m)}$ vs. the number of resulted regions m after segmentation.

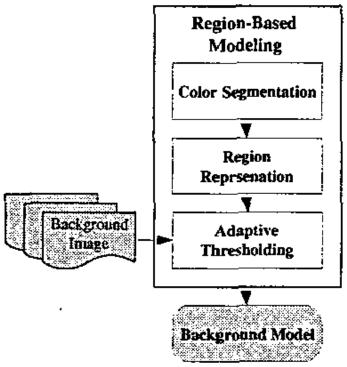

Fig. 2. The flow chart of the background modeling process.

$\sigma_l(i)$ for each sample x_i are

$$\sigma_g = \sqrt{\frac{\sum_{i=1}^{i=100}(x_i - \mu_{100})^2}{100}}$$
$$\sigma_l(i) = \sqrt{\frac{\sum_{j=-1}^{j=1}(x_{i+j} - \mu_3)^2}{3}} \tag{6}$$

We take the most suitable value m for background modeling as the index of the first sample whose local standard deviation is smaller than the global standard deviation.

III. Region-Based Background Modeling

After color segmentation, the background scene is partitioned into a set of mutual exclusive regions that may contain several different quantized colors. In this section, we propose a method to represent each region by a group of clusters with two parameters, threshold θ and fraction f. The detailed flow chart for background modeling is shown in Figure 2.

A. Region Representation

In order to keep a record of color information in a region, the pixels with the same quantized color are modeled by a cluster C. Each cluster is represented as $C = (\mu, \Sigma)$.

- $\mu = (\mu_R, \mu_G, \mu_B)$ is a 3×1 mean vector. μ_R, μ_G, and μ_B are the mean values of red, green, and blue color channels of all pixels in the cluster, respectively.
- Σ is a 3×3 covariance matrix. For simplicity, we assume independence among three different color channels, so that

$$\Sigma = \begin{pmatrix} \sigma_R^2 & 0 & 0 \\ 0 & \sigma_G^2 & 0 \\ 0 & 0 & \sigma_B^2 \end{pmatrix},$$

where σ_R, σ_G, and σ_B are the standard deviation values of red, green, and blue color channels, respectively.

Under such representation, we adopt Mahalanobis distance [3] r for measuring the color distance between a pixel $p = (p_R, p_G, p_B)$ and a cluster $C = (\mu, \Sigma)$.

$$r^2(p, C) = \sum_{i=R,G,B} \frac{(p_i - \mu_i)^2}{\sigma_i^2} \tag{7}$$

Assume that a region R is represented by the K clusters $C_1, C_2, ..., C_K$, the distance from a pixel p to the region R is defined as:

$$d_R(p) = min\{r(p, C_i) | i = 1, 2, ..., K\}, \tag{8}$$

where K is the number of quantized colors in region R.

B. Adaptive Thresholding

The objective of adaptive thresholding is to select a threshold value for further background subtraction process. We propose a histogram-based approach in this paper. By using the next 10 background scenes, the distances between the pixels in a region and the correspondent region model are used to generate a distance histogram for each region. We make an assumption that some noise pixels in each region may result in an additive Gaussian distance distribution. Under such assumption, the mixture probability density function of the distance histogram can be formulated as

$$\begin{aligned} P(d) &= P_1(d) + P_2(d) \\ &= \frac{w_1}{\sqrt{2\pi}\sigma_1} exp(-\frac{(d-\mu_1)^2}{2{\sigma_1}^2}) + \\ &\quad \frac{w_2}{\sqrt{2\pi}\sigma_2} exp(-\frac{(d-\mu_2)^2}{2{\sigma_2}^2}), \end{aligned} \tag{9}$$

where μ_1 and μ_2 are the mean values of the distance distributions resulted from background and noise pixels, respectively. And, σ_1 and σ_2 are the standard deviation values of two distance distributions.

For a threshold value θ, μ_1, μ_2, σ_1, and σ_2 are given, respectively, by

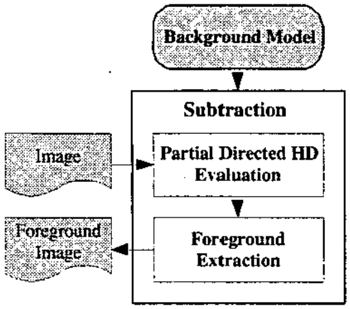

Fig. 3. The flow chart of background subtraction process

$$w_1(\theta) = \sum_{d=1}^{\theta} m(d) \quad w_2(\theta) = \sum_{d=\theta+1}^{a} m(d)$$

$$\mu_1(\theta) = \frac{1}{w_1(\theta)} \sum_{d=0}^{\theta} d \times m(d)$$

$$\mu_2(\theta) = \frac{1}{w_2(\theta)} \sum_{d=\theta+1}^{\infty} d \times m(d)$$

$$\sigma_1^2(\theta) = \frac{1}{w_1(\theta)} \sum_{d=0}^{\theta} (d - \mu_1(\theta))^2$$

$$\sigma_2^2(\theta) = \frac{1}{w_2(\theta)} \sum_{d=\theta+1}^{\infty} (d - \mu_2(\theta))^2; \qquad (10)$$

where $m(d)$ is the probability of the distance value d. The threshold value for each region is found by minimizing the mean squared error e between the mixture probability density function $P(d)$ and $m(d)$.

$$e = \sum_{d=0}^{\infty} (P(d) - m(d))^2 \qquad (11)$$

Another parameter f is defined as $\sum_{d=0}^{\theta} w_1 \times P_1(d)$. Intuitively, the fraction f stands for the percentage of the actual background pixels in a region.

IV. Background Subtraction

In order to be immune to noise and changes, the partial Hausdorff distance is adopted to detect the regions that contain the foreground objects. Then, after obliterating some pixels in these regions, the foreground objects are extracted from the image. Figure 3 shows the details of the subtraction process.

A. Partial Directed Hausdorff Distance

The Hausdorff distance HD is one of commonly used measures for object matching [9] [12]. The classical HD between two pixel sets $A = \{a_1, a_2, ..., a_{N_A}\}$ and $B = \{b_1, b_2, ..., b_{N_B}\}$ of size N_A and N_B, respectively, is defined as

$$H(A,B) = max(h(A,B), h(B,A)), \qquad (12)$$

where $h(A,B)$ is called the directed HD from the point set A to B.

The partial directed HD is defined as

$$h_K(A,B) = K^{th}_{a \in A} d_B(a), \qquad (13)$$

where $d_B(a)$ is the function measuring the minimum distance from the point $a \in A$ to the point set B and $K^{th}_{a \in A}$ denotes the Kth ranked value of $d_B(a)$.

In this paper, the partial directed HD is used to measure the distance between the regions R_I^i in the input image and the regions R_M^i in background scene represented by a group of clusters. Substituting for $d_B(a)$ by $d_R(p)$ defined in Eq.(8), the partial directed HD between two regions, R_I^i and R_M^i is

$$h_K(R_I^i, R_M^i) = K^{th}_{p \in R_I^i} d_{R_M^i}(p). \qquad (14)$$

The value K used in the paper is $f \times N_{R_I^i}$, where f is the fraction parameter of the region R_M^i. The region R_I^i is taken as the potential region that contains the foreground objects if the value $h_K(R_I^i, R_M^i)$ is greater than the threshold parameter θ of the region R_M^i.

B. Foreground Extraction

After determining the potential regions, the next step is to identify which pixels belonging to the foreground object. The following is the strategy adopted in this paper for foreground extraction.

- In the potential regions, the pixels that have the distance less than the value θ are considered as the background pixels.
- A fraction $(1 - f)$ of the pixels that have the minimum distance greater than the value θ are also taken as the background pixels.
- The remainder of the pixels are extracted as the foreground pixels.

From the above, our approach can tolerate noise or slight camera vibration.

V. Experiment

Some experimental results of our approach proposed in this paper are shown in this section. The size of images acquired by AXIS 2310 PTZ Network Camera is 352 × 240. And, the camera is set up on the ceiling for monitoring the front door of our e-home demo site in the Intelligent Robotics Laboratory of National Taiwan University. The number of the regions after applying the J segmentation algorithm to background scene is between 57 to 65. Three cases are used to exhibit the effectiveness of our proposed approach.

In case one, a scene containing a person walking through the corridor is used to show that our approach can detect the foreground objects correctly under normal condition. Figure 4(d)(f) shows the experimental results that a person is detected successfully.

In the second case, the images captured by the camera are manually shifted up by several rows to test our approach. Figure 5(b)(d) show that our approach can cope with the

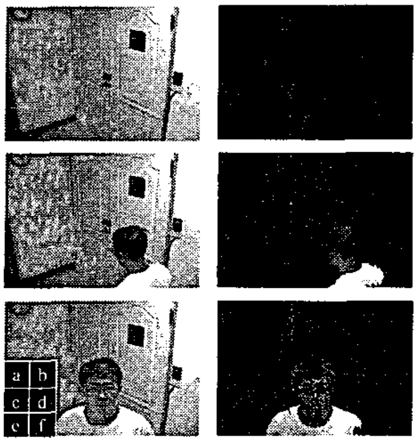

Fig. 4. (a) is the background scene of our e-home demo site. (b) is the result after detection when the scene contains no foreground objects. (d) and (f) are the results after applying our approach to the scenes (c) and (e).

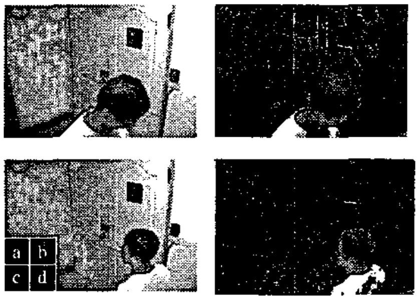

Fig. 5. (a)(c) are the images after shifting the original images up by one and three rows, respectively.

camera displacement situation. On the upper part of Figure 5(b)(d), only a few pixels that reside in the boundary between two regions are misclassified as the foreground pixels.

Finally, the images resulting from inclusion of a zero mean Gaussian noise to RGB color channels independently are shown in Figure 6(a)(c)(e). Two Gaussain noises with different standard deviations, $N(0,\sigma^2)$ and $N(0,\sigma^{10})$, are added for verifying our approach. Even under noisy condition, our system is still able to detect the foreground objects successfully as shown in Figure 6(e)(f).

VI. Conclusion

In this paper, we present a region-based approach to model the background scene. The background scene is first partitioned into a set of representative regions by applying the J segmentation algorithm. And each region is modeled by a group of clusters to economize the use of space. The partial directed Hausdorff distance is adopted for subtracting the foreground objects from the background scene robustly even under noisy or non-static environment. The background scene under three different conditions are provided to demonstrate the effectiveness of the proposed method. In the near future, boundary variation and the relations among the different regions will be taken into consideration to adapt to the background changes in a more robust way.

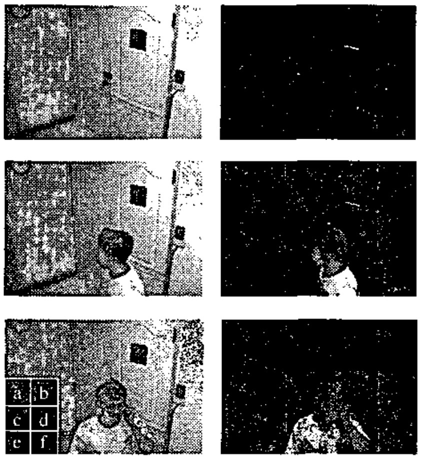

Fig. 6. (a)(c) are the images corrupted by a Gaussian noise $N(0,2^2)$. A Gaussian noise $N(0,10^2)$ is added to generate the image (e).

References

[1] D. Bulter and S. Sridharan. "Real-Time Adaptive Background Segmentation". *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2003.

[2] Y. Deng, B. S. Manjunath, and H. Shin. "Color Image Segmentation". *Proceeding of IEEE Conference on Computer Vision and Pattern Recognition*, 1999.

[3] R. O. Duda, P. E. Hart, and D. G. Stork. *"Pattern Classification"*. Wiley Interscience, 2000.

[4] A. Elgammal, R. Duraiswami, D. Harwook, and L. S. David. "Background and Foreground Modeling Using Nonparametric Kernel Density Estimation for Visual Surveillance". *Proc. of IEEE*, 90(7):1151–1163, July 2002.

[5] A. Elgammal, D. Harwood, and L. Davis. "Non-parametric Model for Background Subtraction". *IEEE International Conference on Computer Vision Frame-Rate Workshop*, 1999.

[6] N. Friedman and S. Russell. "Image Segmentation in Video Sequence: A Probabilistic Approach". *International Conference on Uncertainty in Artificial Intelligence*, August 1997.

[7] S. Gupte, O. Masoud, R. F. K. Martin, and N. P. Papanikolopoulos. "Detection and Classification of Vehicles". *IEEE Transactions on Intelligent Transportation Systems*, 3(1):37–47, March 2002.

[8] I. Haritaoglu, D. Harwood, and L. S. Davis. "W4: Real-Time Surveillance of People and Their Activities". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):809–830, August 2000.

[9] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge. "Comparing Images Using the Hasudorff Distance". *IEEE Transaction on Pattern Recongition and Machine Intelligence*, 15(9):850–863, September 1993.

[10] T. Matsuyama, T. Ohya, and H. Habe. "Background subtraction for nonstationary scenes". *in Proc. 4th Asian Conference on Computer Vision*, pages 662–667, 2000.

[11] A. Rosenfeld and J. L. Pfaltz. "Sequential Operations in Digital Picture Processing". *Journal of the Association for Computing Machinery*, 13:471–494, 1966.

[12] W. Rucklidge. *"Efficient Visual Recognition Using the Hausdorff Distance"*. Springer, 1996.

[13] N. L. Seed and A. D. Houghton. "Background updating for real-time image processing at TV rates". *SPIE*, 901:73–81, 1988.