International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 24 (2015) pp 45665-45673

Moving Object Segmentation Using Background Modeling With Orthogonal Polynomials

Dr. R. Punidha[1], Ms. K. Gayathri Devi[2], K. Suguna[3] and R. Subaja[4]

[1]Professor/CSE, Vel Tech Multitech Dr.Rangaraja Dr.Sakunthala Engineering College, Chennai-62.
[2,3,4]Assistant Professor/IT, Vel Tech Multitech Dr.Rangaraja Dr.Sakunthala Engineering College, Chennai-62.
E-Mail : punidhar@gmail.com

ABSTRACT
A new background modeling and moving object segmentation algorithm with orthogonal polynomials based transformation is proposed in this paper. From a set of orthogonal polynomials functions, the polynomial operators and polynomial basis operators of different width are obtained for the proposed orthogonal polynomials transformation. The video frames are first compressed with the proposed transformation as per in MPEG compression standard. In this work, we present the modeling background algorithm directly from the compressed video using mixture of Gaussians and a two-stage segmentation approach based on the proposed background model. The proposed method utilizes orthogonal polynomials transformed coefficients at block level to represent background and adapt the background by updating the proposed transformed coefficients. The proposed segmentation approach can extract the foreground objects with pixel accuracy through a two-stage process. First a new background subtraction technique in the orthogonal polynomials transformed domain is exploited to identify the block regions fully or partially occupied by foreground objects and then pixels from these foreground blocks are further classified in the spatial domain. The experimental results shows that the proposed background modeling algorithm can achieve comparable accuracy to their counterparts in the spatial domain and the associated segmentation scheme can visually generate good segmentation results with less computational effort.

Keywords—Background modeling, Orthogonal polynomials transform, Moving object segmentation.

I. INTRODUCTION

In recent years, research on video segmentation has received continuously increasing interests to support a variety of video applications, such as video summarization, editing, annotation and indexing. Usually there are two different video segmentation approaches, i.e., shot-based segmentation that uses a set of key-frames to represent a video shot and object-based segmentation which partitions a video shot into objects and background. Extracting video contents at different semantic levels, these two methods are usually used separately and independently in a video analysis framework [1]. For one unstructured raw data, shot-based segmentation partitions the video sequence into a set of video shots, and extracts key-frames to represent the major content of each video shot. Thus shot-based video segmentation can provide compact abstraction and delineation for video indexing, browsing and retrieval [2]. Differently, object-based video segmentation is to decompose one video shot into objects and background, which are usually application-dependent. One video object can be considered as a collection of image pixels that corresponds to the projection of a real object (e.g., moving cars or people) in successive image planes of a video sequence [3]. Unlike shot-based video segmentation that has a frame as the basic unit, object-based segmentation can provide objects that represent a raw video at a higher semantic level.

Many object-based segmentation (or moving object segmentation) algorithm have been reported in various literatures [4-9]. A popular technology in the existing moving object segmentation approaches is the background subtraction technique, which extracts moving objects in an image sequence captured from a static camera by comparing each coming frame with a background model. Most existing background modeling approaches operate in the spatial domain, and follow the same philosophy that a background model is independently estimated for each spatial location through a series of pixel values (gray or color) at temporal axis. The running average (RA) approach [10], is an early robust approach, which has the desirable computational speed and low memory requirement. Koller et al. [11] described its modified version called selective running average (SRA) and its application is in real-time traffic monitoring. The median filtering approach [12]-[14] is another background modeling technique used by many systems. Cheung and Kamath [15] reported the simple median approach and provided competitive performance with low computational complexity. McFarlane and Schofield [16] roughly estimated the median through adaptively increasing or decreasing the background value by one. Another background modeling technique used a Gaussian distribution [17] to model each background pixel in their Pfinder system. Piccardi [18] called this a running Gaussian average (RGA) and argued the standard deviation introduced could provide an adaptive threshold for classifying pixels. The mixture of Gaussians (MoG) based approach [19, 20] has obtained tremendous popularity due to its capability to model multimodal backgrounds. Elgammal et al. [21] demonstrated the use of non-parametric estimation to model background, which can be considered as an

extension of [19] and [20] with more Gaussian mixtures. More complex approaches are reported in [22–24]. The Wiener prediction filter to estimate the current background is employed in [22]. Han et al. [23] presented a sequential density approximation method to identify density modes, and then a Gaussian component was assigned for each mode to represent the background. In [24], a method called SACON is presented that estimates a background model through computing sample consensus, and reported promising performance on the Wallflower image sequences. Contrastively, [25] and [26] exploited a new philosophy that a background model was estimated based on pixel blocks, instead of independent pixels. Although the spatial correlation of neighboring pixels is taken into account, the robustness of their models could not be guaranteed due to lack of a mechanism to adaptively maintain a background model in real-time. Moreover, their approaches involve very high computational complexity.

In recent years, most algorithms of moving object segmentation are in pixel-domain and little research has been done on moving object segmentation in compressed domains. The segmentation in compressed domains offers an obvious advantage in operation because of very large amounts of multimedia data to be compressed for storage and transmission. Consequently, the technology of moving object segmentation in compressed domain has its undeniable practical value. Most of the compressed domain approaches based on discrete cosine transformation (DCT) only exploit dc coefficients [27–31] to identify moving regions. So they can only obtain very rough representations of moving objects at the block resolution, and cannot obtain object contours in pixel accuracy. Babu et al. [32] presented a novel video object segmentation approach in compressed domain with pixel accuracy, which are based on sparse motion vectors in MPEG compressed domain.

In this paper, first a video encoding scheme with orthogonal polynomials based transformation (OPT) has been proposed. And then, we propose a background subtraction frame-work in compressed domain, which models background directly from compressed video using the proposed transformed coefficients and is able to extract moving objects at the pixel resolution. To demonstrate the feasibility of the framework, three styles of background modeling approaches, i.e., the RA algorithm, the median algorithm, the MoG algorithm, are designed in the DCT domain. They can generate background with comparable accuracy to their counter-parts in the spatial domain in a more efficient way. Based on the proposed background models, we present a two-stage segmentation approach to extract moving objects with pixel accuracy. Compared with [30], our segmentation approach based on back-ground models has lower computational cost.

Section II present the orthogonal polynomials based video coder in which the DCT of MPEG coding stream is replaced with orthogonal polynomials based transformation. Since the DCT is poor at approximating discontinuities or impulse in the imagery signal [33], a new orthogonal polynomials based coder is proposed in this paper. The proposed coding scheme has resulted from our investigations into some low level feature extraction problems such as detection of textures and edges, in monochrome and color images [34–36]. In these works, we design a point-spread operator M due to a class of orthogonal polynomials and define a linear two dimensional transformation to analyze the low level primitives of the image under analysis [37]. In another work, we have designed a coding scheme that separates significant signal components from the noise, with the help of statistical design of experiments approach [38]. In section III, the background modeling based on OPT is presented. Section IV details a two stage approach to extracting the moving objects with pixel accuracy based on the previous background model. The experimental results are presented in section V and section VI concludes the paper.

II. ORTHOGONAL POLYNOMIALS BASED TRANFORM CODING

a. Orthogonal Polynomials model

In order to devise a transform coding for lossless image coder, a linear 2-D image formation system is considered around a Cartesian coordinate separable, blurring, point spread operator in which the image I results in the superposition of the point source of impulse weighted by the value of the object function f. Expressing the object function f in terms of derivatives of the image function I relative to its Cartesian coordinates is very useful for analyzing the image. The point spread function $M(x, y)$ can be considered to be real valued function defined for $(x, y) \in X \times Y$, where X and Y are ordered subsets of real values. In case of gray-level image of size $(n \times n)$ where X (rows) consists of a finite set, which for convenience can be labeled as $\{0, 1, \ldots, n\text{-}1\}$, the function $M(x, y)$ reduces to a sequence of functions.

$$M(i, t) = u_i(t), \quad i, t = 0, 1, \ldots, n\text{-}1 \tag{1}$$

The linear two dimensional transformation can be defined by the point spread operator $M(x, y)$ ($M(i, t) = u_i(t)$) as shown in equation (2).

$$\beta'(\zeta,\eta)=\int_{x\in X}\int_{y\in Y} M(\zeta,x)M(\eta,y)I(x,y)dxdy \tag{2}$$

Considering both X and Y to be a finite set of values $\{0, 1, 2, \ldots, n-1\}$, equation (2) can be written in matrix notation as follows

$$\left|\beta'_{ij}\right| = \left(|M|\otimes|M|\right)^t |I| \tag{3}$$

where $\otimes$ is the outer product, $|\beta'_{ij}|$ are n^2 matrices arranged in the dictionary sequence, $|I|$ is the image, $|\beta'_{ij}|$ are the coefficients of transformation and the point spread operator $|M|$ is

$$|M| = \begin{vmatrix} u_0(t_1)u_1(t_1)\cdots u_{n-1}(t_1) \\ u_0(t_2)u_1(t_2)\cdots u_{n-1}(t_2) \\ \vdots \\ u_0(t_n)u_1(t_n)\cdots u_{n-1}(t_n) \end{vmatrix} \tag{4}$$

Consider a set of orthogonal polynomials $u_0(t)$, $u_1(t)$, …, $u_{n-1}(t)$ of degrees 0, 1, 2, …, $n\text{-}1$ respectively to construct the polynomial operators of different sizes from equation (4) for $n \geq 2$ and $t_i = i$. The generating formula for the polynomials is as follows.

$u_{i+1}(t) = (t-\mu)\, u_i(t) - b_i(n)\, u_{i-1}(t)$ for $i \geq 1$, (5)
$u_1(t) = t - \mu$, and $u_0(t) = 1$,

where $b_i(n) = \dfrac{\langle u_i, u_i \rangle}{\langle u_{i-1}, u_{i-1} \rangle} = \dfrac{\sum_{t=1}^{n} u_i^2(t)}{\sum_{t=1}^{n} u_{i-1}^2(t)}$

and $\mu = \dfrac{1}{n}\sum_{t=1}^{n} t$

Considering the range of values of t to be $t_i = i$, $i = 1, 2, 3, \ldots, n$,

$$b_i(n) = \frac{i^2\left(n^2 - i^2\right)}{4\left(4i^2 - 1\right)}, \qquad \mu = \frac{1}{n}\sum_{t=1}^{n} t = \frac{n+1}{2}$$

The point-spread operators $|M|$ of different size from equation (4) is constructed using the above orthogonal polynomials for $n \geq 2$ and $t_i = i$. For the convenience of point-spread operations, the elements of $|M|$ are scaled to make them integers.

b. The orthogonal polynomials basis

For the sake of computational simplicity, the finite Cartesian coordinate set X, Y is labeled as {1, 2, 3}. The point spread operator in equation (3) that defines the linear orthogonal transformation for image coding can be obtained as $|M| \otimes |M|$, where $|M|$ can be computed and scaled from equation (4) as follows.

$$|M| = \begin{vmatrix} u_0(x_0) & u_1(x_0) & u_2(x_0) \\ u_0(x_1) & u_1(x_1) & u_2(x_1) \\ u_0(x_2) & u_1(x_2) & u_2(x_2) \end{vmatrix} = \begin{vmatrix} 1 & -1 & 1 \\ 1 & 0 & -2 \\ 1 & 1 & 1 \end{vmatrix} \quad (6)$$

The set of polynomial basis operators O_{ij}^{n} $(0 \leq i, j \leq n-1)$ can be computed as

$$O_{ij}^{n} = \hat{u}_i \otimes \hat{u}_j^{t}$$

where $\hat{u}_i$ is the $(i+1)^{st}$ column vector of $|M|$.

The complete set of basis operators of sizes (2×2) and (3×3) are given below.

Polynomial basis operators of size (2×2) are

$$\left[O_{00}^{2}\right] = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \left[O_{01}^{2}\right] = \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix},$$
$$\left[O_{10}^{2}\right] = \begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix}, \left[O_{11}^{2}\right] = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

Polynomial basis operators of (3×3) are

$$\left[O_{00}^{3}\right] = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix},$$
$$\left[O_{01}^{3}\right] = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix},$$
$$\left[O_{02}^{3}\right] = \begin{bmatrix} 1 & -2 & 1 \\ 1 & -2 & 1 \\ 1 & -2 & 1 \end{bmatrix}$$
$$\left[O_{10}^{3}\right] = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix},$$
$$\left[O_{11}^{3}\right] = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix},$$
$$\left[O_{12}^{3}\right] = \begin{bmatrix} -1 & 2 & -1 \\ 0 & 0 & 0 \\ 1 & -2 & 1 \end{bmatrix}$$
$$\left[O_{20}^{3}\right] = \begin{bmatrix} 1 & 1 & 1 \\ -2 & -2 & -2 \\ 1 & 0 & 1 \end{bmatrix},$$
$$\left[O_{21}^{3}\right] = \begin{bmatrix} -1 & 0 & 1 \\ 2 & 0 & -2 \\ -1 & 0 & 1 \end{bmatrix},$$
$$\left[O_{22}^{3}\right] = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

It is also proved that the set of $(n\times n)$ $(n>=2)$ polynomial operators forms a basis, i.e. it is complete and linearly independent.

III. ORTHOGONAL POLYNOMIALS TRANSFORM BASED BACKGROUND MODELING

a. Orthogonal polynomials transform based video coder

Recent progress in digital technology has made the widespread use of compressed digital video signals practical. For efficient storage and transmission of video signals, the raw video data are compressed by removing spatial and temporal redundancy. A common framework of popular International video compression standards, such as MPEG-1, 2, 4 and H.26X is shown in fig. 1, in which the DCT is

replaced with OPT for transformation. The input video sequence generally consists of I-frames, P-frames, B-frames. An I-frame is encoded and decoded independently, while a P-frame or B-frame needs to refer to adjacent P-frames or I frames to be encoded or decoded. Each I-frame is divided into 8 by 8 pixel blocks in the spatial domain, and then each block is transformed by OPT into a set of coefficients in the frequency domain to reduce spatial redundancy as given in equation (3) of section II.A. The transformation (*D*) can be represented as a concise matrix multiplication

$$D=MI \tag{7}$$

where *M* ($M = |M| \otimes |M|^t$) is OPT matrix and *I* is input block pixel values. The inverse OPT is applied with orthogonal polynomials basis operator as described in section II.b and the reconstructed frame is obtained.

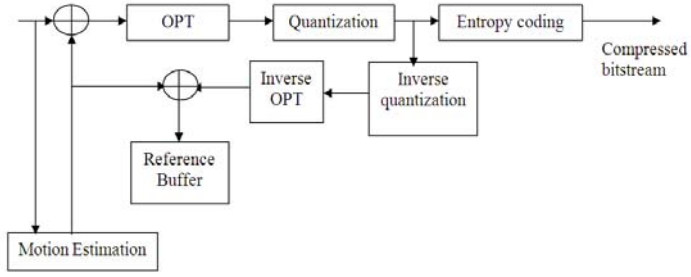

Figure 1. MPEG based video coder with orthogonal polynomials

b. Background modeling using MoG in OPT domain

The background modeling presented in this section is based on orthogonal polynomials video as presented in previous section. The MoG algorithm is used in this section in OPT domain for modeling the background. As other algorithms in the compressed domain, this algorithm does not involve the computation of inverse transformation operation. In the proposed algorithm, the OPT coefficients are used to represent background $D_t^B = d_{t,l}^B, l = 1,2,...,BL$ at time *t*, where $d_{t,l}^B$ is a 64-dimensional background vector for the l^{th} pixel block at time *t*, *BL* is the number of blocks in a frame. The MoG algorithm in the transformed domain models each block of the OPT coefficients of as a mixture of Gaussians, i.e.

$$P_r(d_l \mid \lambda_l) = \sum_{i=1}^{M} w_{l,i} G_i(d_l) \tag{8}$$

where d_l is a 64-dimensional OPT coefficient vector for the block *l*, $G_i(d_l)$ denotes a Gaussian component density and $w_{l,i}, (\sum_{i=1}^{M} w_{l,i} = 1)$ is the corresponding component weight. To avoid expensive computation in modeling background, it is assumed that each component of d_l has the same variance. Additionally, since different components of d_l are independent of each other, $G_i(d_l)$ is defined as having the following function form:

$$G_i(d_l) = 1/((2M)^{32}\sigma_{l,i}^{64}) \times \exp(-(d_l - u_{i,l})^T (d_l - u_{l,i})/2\sigma_{l,i}^2)) \tag{9}$$

where $u_{l,i}$ is the 64-dimensional mean vector and $\sigma_{l,i}$ is the variance for the block *l*. All the parameters in this model are collectively represented by the notion

$$\lambda_l = \{w_{l,i}, \mu_{l,i}, \sigma_{l,i}\}, \quad i = 1,2,...,M, l = 1,2,...,BL$$

We use a very similar procedure as [20] to adaptively update the background model. The difference primarily lies in that the update is based on blocks instead of pixels, and fewer parameters are involved. For a block $d_{t,l}$, $l=1,2,...,BL$, in the frame at time *t*, a match is defined as

$$(d_{t,l} - u_{l,i}^t)^T (d_{t,l} - u_{l,i}^t) < \delta_{l,i}^t \tag{10}$$

where $\delta_{l,i}^t$ is a model parameter, called a matching threshold. The set of the model parameters estimated for a block *l* is

$$\lambda_l^t = \{w_{l,i}^t, \mu_{l,i}^t, \delta_{l,i}^t, \quad i = 1,2,...,M, t = 0,1,2,...\}$$

First, the algorithm checks the block $d_{t,l}$ against the existing Gaussian components to find the best matched component. Among the matched components satisfying (10), the best matched component for the block *l* at time *t* is indexed by

$$\hat{i} = \arg\min_i (d_{l,i} - u_{l,i}^t)^T (d_{l,i} - u_{l,i}^t)/\delta_{l,i}^{t-1} \tag{11}$$

Once a best matched Gaussian component $\hat{i}$ is identified, its parameters are then updated as follows:

$$\begin{aligned} w_{l,\hat{i}}^t &= (1-\alpha) w_{l,\hat{i}}^{t-1} + \alpha \\ u_{l,\hat{i}}^t &= (1-\rho) u_{k,\hat{i}}^{t-1} + \rho d_l^t \\ \delta_{l,\hat{i}}^t &= (1-\rho)\delta_{l,\hat{i}}^{t-1} + \rho (d_{t,l} - u_{l,\hat{i}}^t)^T (d_{t,l} - u_{l,\hat{i}}^t) \end{aligned} \tag{12}$$

where $\alpha (0 < \alpha < 1)$ is a user-defined learning rate and $\rho = \alpha / w_{l,\hat{i}}^t$. For those unmatched components $i(\neq \hat{i}),$, their $\mu_{l,i}^t, \delta_{l,i}^t$, are kept unchanged, but $w_{l,i}^t$ are adjusted as

$$w_{l,i}^t = (1-\alpha) w_{l,i}^{t-1}, \quad i \neq \hat{i}$$

Finally the weights of all the Gaussian components are renormalized. If none of *M* Gaussian components can match the current block *l*, the Gaussian component with the least weight is replaced by a new Gaussian component with the OPT values of the block *l* as its mean, an initially high matching threshold $\delta_{l,i}$, and a low priority weight. Then the weights are renormalized. A similar method is adopted to determine whether a block $d_{t,l}$ is a foreground block. Let $i_1, i_2, ..., i_M$ denote the index of Gaussian components ordered based on the ratio of weights to matching thresholds, the first *S* Gaussian components are chosen as background model, and *S* is determined by

$$S = \arg\min_b \left(\sum_{l=i_1}^{i_b} w_{l,i}^t > T \right) \tag{13}$$

where *T* is a threshold. Compared with [20], this MoG algorithm has more compact model parameters. For an 8 by 8 pixel block, the MoG algorithm in [20] needs to estimate and update 64*3**Mx* = 192**Mx* parameters for the mixture of *Mx* Gaussians, while this algorithm only estimates

2**Mx*+64**Mx*=66**Mx* parameters. The estimation of fewer parameters means lower computational complexity in updating the model.

IV. MOVING OBJECT SEGMENTATION

In this section, a two-stage process to segment the moving objects based on the background models presented in the previous section is presented. First a background subtraction technique in the OPT domain is exploited to identify blocks fully or partially covered by foreground objects, and then the pixels from the foreground blocks are further classified into foreground or background pixels in the spatial domain. In the first stage, the difference between a coming frame and a corresponding background is evaluated at the block level in the OPT domain. For this, the Euclidean distance between $d_{t,l}$ and $d_{t,l}^B$ is calculated

$$\Omega_{t,l} = \left\| d_{t,l} - d_{t,l}^B \right\|, \quad l = 1,2,...,L \tag{14}$$

to measure the content difference of the coming frame against the background for block *l* at time *t*, where $d_{t,l}$, $d_{t,l}^B$, denote the OPT coefficient representations of the l^{th} pixel block at time *t* for the coming frame and the corresponding background, respectively. If $\Omega_{t,l} > \tau$, (where τ is a threshold) the block *l* is labeled as a foreground block. The background subtraction in the OPT domain is completely consistent with its counterpart in the spatial domain, but it takes 64 pixels as a whole. The sum of squared differences (SSD) is commonly used by video encoders to measure the extent to which a block matches another block in the computation of motion vectors, so a large SSD in computing $\left\| f_{t,l} - f_{t,l}^B \right\|$ shows the block *l* is occupied by moving objects, where $f_{t,l}$ and $f_{t,l}^B$ are the pixel values corresponding to $d_{t,k}$ and $d_{t,l}^B$. It is noticed that

$$\begin{aligned}
\left\| f_{t,l} - f_{t,l}^B \right\| &= (f_{t,l} - f_{t,l}^B)^T (f_{t,l} - f_{t,l}^B) \\
&= (K^T d_{t,l} - K^T d_{t,l}^B)^T (K^T d_{t,l} - K^T d_{t,l}^B) \\
&= (d_{t,l} - d_{t,l}^B)^T KK^T (d_{t,l} - d_{t,l}^B) \\
&= (d_{t,l} - d_{t,l}^B)^T (d_{t,l} - d_{t,l}^B) \\
&= \left\| (d_{t,l} - d_{t,l}^B) \right\|_2
\end{aligned} \tag{15}$$

Thus $\Omega_{t,l}$ is a background subtraction measure for pixel blocks. The first-stage calculation speeds up the segmentation of foreground objects, since the second-stage background subtraction does not need to be performed on background blocks. That can help those algorithms operating in compressed domain run faster than the segmentation algorithm in the spatial domain. A reasonable high threshold τ is beneficial, since more blocks are prone to be classified as background. But a high threshold also has some negative effect when very few pixels in a block are covered by foreground objects. In the case, a relatively high $\Omega_{t,l}$ makes the block be misclassified as background, which results in permanently missing the foreground pixels. To overcome the problem, a dilation labeling procedure is introduced at the end of the first-stage calculation. For each foreground block, its 4-neighbouring blocks are labeled as foreground blocks in the dilation labeling. Intuitively foreground blocks are generally connected with each other, and blocks with very few foreground pixels generally correspond to the boundary regions of foreground objects. In this work, a reasonable high threshold is used to locate pixel blocks of which most pixels are covered by foreground objects, and the dilation labeling procedure to get back the missing foreground blocks with very few foreground pixels.

In the second stage, the inverse OPT is used to transform the foreground blocks identified in the current frame and the corresponding blocks in the current background into pixel values in the spatial domain. Then any effective segmentation approach in the spatial domain can be used to extract foreground objects at pixel level. In MoG model, there are generally multiple Gaussians representing the background. To apply the background subtraction technique, a Gaussian component has to be selected as the reference background. The proposed scheme uses the latest best matched Gaussian background component as the reference background. To formulate it more clearly, we use $g_{t,l}$ to denote the best matched Gaussian background component against the block $d_{t,l}$. For a foreground block at time *t*, let $g_{t_0,l}, g_{t_1,l}, g_{t_2,l},...,g_{t_n,l},...$ denote the existing best matched Gaussian background components at different time $t_0,t_1,...,t_l,...$, where $t \geq t_0 \geq t_1 ... \geq t_l ...$, and then $g_{t_0,l}$ is chosen as the reference background at time *t*.

It is possible that some applications do not care about the texture and color information of moving objects, and prefer only accurate shape information. In the situation, the inverse OPT can be only calculated one time for each foreground block instead of two times. Given current background D_t^B and a current frame D_t, their difference image F_t^* in the spatial domain can be calculated by

$$f_{t,l}^* = K^T (d_{t,l} - d_{t,l}^B) \tag{16}$$

where

$D_t^B = \{d_{t,l}^B\}, D_t = \{d_{t,l}\}, F^* = \{f_{t,k}^*\}, l = 1,2,...BL.$

For each pixel *p* in block *l*, if the pixel value $f_{t,l}^*(p)$ satisfies

$$\left| f_{t,l}^* \right| > \eta$$

where η is a predefined threshold, the pixel *p* is classified as a foreground pixel, otherwise background. Compared with moving object segmentation approaches in the spatial domain, the transformed domain approach works more efficiently, since it is able to locate the blocks covered by foreground objects without fully decompressing video, and the background subtraction in the spatial domain is only

performed on foreground blocks, instead of a whole video frame. Generally foreground blocks in a video frame take up a small proportion, so the evaluation of the inverse OPT is not required for a large number of background blocks. On the other hand, compared with moving object segmentation approaches based on DC coefficients in the OPT domain, the proposed approach can segment foreground objects at the pixel resolution, since the background models utilized do not lose any information and are able to obtain background with pixel resolution in the spatial domain. Thus, the proposed approach is able to combine the efficiency of the approaches based on DC coefficients and the accuracy of the background subtraction approach in the spatial domain together.

V. EXPERIMENTS AND RESULTS

The proposed segmentation algorithm is experimented with more than 3000 video sequences and two sample video frames viz. cricket and football video frames which are of size (300×225) with pixel values in the range of (0-255) are shown in figure 2(a) and 2(b) respectively. In order to quantitatively compare the proposed algorithm with their counterparts in the spatial domain, we selected sample frames from sequence I and manually label all the pixels of moving objects as the ground truth. Two metrics, false negative rate (FNR) and false positive rate (FPR), were used to quantify the segmentation performance of the proposed algorithm. They are defined as

$$FNR=\frac{the\ number\ of\ foreground\ pixels\ wrongly\ classified}{the\ number\ of\ foreground\ pixels\ in\ the\ ground\ truth}$$

$$FPR=\frac{the\ number\ of\ background\ pixels\ wrongly\ classified}{the\ number\ of\ background\ pixels\ in\ the\ ground\ truth}$$

The first I-frame is used as the initial background for the proposed background modeling algorithm. In the proposed background modeling, the unique classification algorithm for foreground blocks was utilized based on (13) and we chose T=0.7. Once a block was classified as a foreground block, the representation of the block and the background block in the OPT domain at the corresponding location were transformed into the spatial domain, where any background subtraction algorithm can be applied. The segmentation results of the proposed algorithm are shown in fig.3(a) and 3(b) respectively. Table 1 summarizes and compares the segmentation performances of the proposed algorithm in the OPT domain with their counterparts in the spatial domain. The experimental result shows the proposed algorithm have a lower FPR than their counterpart. The background subtraction technique assumes each pixel is independent of each other, so it classifies each pixel independently. That makes the segmentation procedure is very sensitive to illumination changes and noises. But a block-based classification is used by our algorithms to identify foreground blocks. Equation (15) shows the SSD of the OPT coefficients is equal to the SSD of the corresponding pixel values in that block. That means the proposed method integrates the results of background subtraction from 64 pixels in a block together to classify the block, and the correlation among neighboring pixels is considered. Even if the difference at a pixel location in the block is very high, the pixel can still be classified as background, as long as the differences at other pixel locations in the block are so small that the sum of squared differences for all the pixel locations is lower than a predefined threshold. At the same time, the proposed algorithm in the OPT domain commonly have a higher FNR than their counterparts.

(a) Cricket

(b) Football

Figure 2. Original test video frames

(a) Cricket

(b) Football

Figure 3. Results of proposed segmentation technique.

Table 1. Comparison of segmentation performance of the proposed algorithm with their counterparts in the spatial domain

	Sequence I		Sequence II	
	FNR(%)	FPR(%)	FNR(%)	FPR(%)
Proposed modeling with OPT	8.76	0.15	25.92	0.06
Proposed modeling using DCT	9.54	0.27	28.31	0.15
Proposed modeling in Spatial domain	3.52	1.27	20.71	3.61

Table 2 gives the time consumptions of segmenting moving objects from various background models. The first and second row of table 2 gives the total time consumptions that include background construction computation and foreground object segmentation. The time consumptions for only segmenting moving objects are listed in the third row and the forth row
of table 2. The proposed segmentation scheme makes the background subtraction computation be performed only on a small part of blocks, but our experimental results show it does not surely result in less time in the segmentation of moving objects. To identify foreground blocks, the proposed segmentation scheme needs to compute the difference between two blocks from current frame and current background based on (14), which involves 64 multiplications. Moreover, inverse OPT is also required to be performed on those foreground blocks before background subtraction. Compared with the segmentation computation of the algorithm in the spatial domain, these extra computations mean the proposed segmentation scheme may cost more time than their counterparts. In other words, the time for the extra computations may exceed the saved time due to no background subtractions for background blocks. The segmentation method in the OPT domain efficiently identify foreground and background through sorting weights of Gaussian components one time for each block. Comparatively, its counterpart does the same ordering evaluation for each pixel, i.e., 64 times for each block. The last row in Table 2 gives the time cost ratio of the algorithm in the spatial domain, DCT domain and the OPT domain, when we consider background modeling and segmentation of moving objects together. These ratios show that, it is efficient to model background in the OPT domain and then segment moving objects through the proposed approach.

Table 2. Comparison of time consumption of segmenting moving objects between the proposed algorithm and their counterpart.

Time (ms)	Sequence I	Sequence II
Total time (spatial)	95108	98234
Total time (OPT)	14790	16790
Segmentation time (spatial)	8222	9871
Segmentation time (OPT)	1071	1143
Ratio of total time	15.60 %	17.10%

VI. CONSLUSION

In this paper, a new background modeling and moving object extraction with orthogonal polynomials based transformation is proposed. Based on the background model in the OPT domain, the proposed segmentation scheme can effectively segment moving objects with pixel resolution. The experimental results shows that the background model in the OPT domain using MoG generate the comparable background estimations as their counterparts in the spatial domain. Our evaluation experiments also shows that the proposed segmentation scheme based on the proposed background models generate a visually better segmentation results and has lower computational complexity in segmenting moving objects. For instance, the time cost of segmenting moving objects based on the proposed background model in the OPT domain is only 19.6% of their counterpart in spatial domain.

REFERENCES

1. M. Ferman, A. M. Tekalp and R. Mehrotra, 1998, "Effective content representation for video," in Proc. IEEE Int. Conf. Image Processing, pp. 521–524.
2. P. Aigrain, H. Zhang and D. Petkovic, 1996, "Content-based representation and retrieval of visual media: a state-of-the-art review," Multimedia Tools and Applications, Vol. 3, pp. 179–202.
3. A. Cavallaro, 2002, "From Visual information to knowledge: semantic video object segmentation, tracking and description," Ph.D. dissertation.
4. T. Meier and K. N. Ngan, 1998, "Automatic segmentation of moving objects for video object plane generation," IEEE Trans. Circuits Syst. Video Technol., Vol. 8, No. 7, pp. 525–538.
5. D.Wang, 1998, "Unsupervised video segmentation based on watersheds and temporal tracking," IEEE Trans. Circuits Syst. Video Technol., Vol. 8, No. 7, pp. 539–546.
6. P. Salembier and F. Marqués, 1999, "Region-based representations of image and video: Segmentation tools for multimedia services," IEEE Trans. Circuits Syst. Video Technol., Vol. 9, No. 8, pp. 1147–1169.
7. Y. N. Deng and B. S. Manjunath, 2001, "Unsupervised segmentation of color-texture regions in images and video," IEEE Trans. Pattern Anal. Mach. Intell., Vol. 23, No. 8, pp. 800–810.
8. I. Patras, E. A. Hendriks, and R. L. Lagendijk, 2001, "Video segmentation by MAP labeling of watershed segments," IEEE Trans. Pattern Anal. Mach. Intell., Vol. 23, No. 3, pp.326–332.
9. S. Y. Chien, S. Y. Ma and L. G. Chen, 2002, "Efficient moving object segmentation algorithm using background registration technique," IEEE Trans. Circuits Syst. Video Technol., Vol. 12, No. 7, pp. 577–586.
10. K. Karmann, A. Brandt and R. Gerl, 1990, "Moving object segmentation based on adaptive reference images," in European Signal Processing Conf., pp. 951–954.
11. D. Koller, J. Weber, T. Huang, J. Mal, G. Ogasawara, B. Rao, and S. Russell, 1994, "Towards robust automatic traffic scene analysis in real-time," in Proc. Int. Conf. Pattern Recognition, pp. 126–131.
12. B. Gloyer, H. Aghajan, K. Siu and T. Kailath, 1995, "Video-based freeway monitoring system using recursive vehicle tracking," in Proc. SPIE, Image and Video Processing III, Vol. 2421, pp. 173–180.
13. R. Cutler and L. Davis, 1998, "View-based detection and analysis of periodic motion," in Proc. IEEE Int. Conf. Pattern Recognit., pp. 495–500.
14. B. P. L. Lo and S. A. Velastin, 2001, "Automatic congestion detection system for underground platforms," in Proc. IEEE Int. Symp. Intelligent Multimedia,Video Speech Processing, pp. 158–161.
15. S. S. Cheung and C. Kamath, 2004, "Robust techniques for background subtraction in urban traffic video," Proc. SPIE, Vol. 5308, pp. 881–892.
16. N. J. B. McFarlane and C. P. Schofield, 1995, "Segmentation and tracking of piglets in images," Mach. Vis. Appl., Vol. 8, No. 3, pp. 187–193.
17. C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, 1997, "Pfinder: Realtime tracking of the human body," IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 19, pp. 780–785.
18. M. Piccardi, 2004, "Background subtraction techniques: A review," in Proc. IEEE Int. Conf. Syst., Man .Cybern., pp.3099–3104.
19. N. Friedman and S. Russell, 1997, "Image segmentation in video sequences: A probabilistic approach," in Proc. 13th Ann. Conf. Uncertainty Artif. Intell., pp. 175–181.
20. C. Stauffer and W. E. L. Grimson, 1999, "Adaptive background mixture models for real-time tracking," in Proc. IEEE Int. Conf. Comput. Vis.Pattern Recognit., pp. 246–252.
21. M. Elgammal, D. Harwood, and L. S. Davis, 2000, "Non-parametric model for background subtraction," in Proc. 6th Euro. Conf. Comput. pp. 751–767.
22. K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, 1999, "Wallflower: Principles and practice of background maintenance," in Proc. IEEE Int. Conf. Comput. Vis., pp. 255–261.
23. B. Han, D. Comaniciu, and L. Davis, 2004, "Sequential kernel density approximation through mode propagation: Applications to background modeling," in Proc. Asian Conf. Comput. Vis., pp. 818–823.
24. H. Wang and D. Suter, 2007, "A consensus-based method for tracking: Modelling background scenario and foreground appearance," Pattern Recognit., Vol. 40, No. 3, pp. 1091–1105.
25. N. M. Oliver, B. Rosario, and A. P. Pentland, 2000, "A bayesian computer vision system for modeling human interactions," IEEE Trans. Pattern Anal. Mach. Intell., Vol. 22, No. 8, pp. 831–843.
26. M. Seki, T. Wada, H. Fujiwara, and K. Sumi, 2003, "Background subtraction based on co-occurrence of image variations," in Proc. IEEE Int. conf. Computer Vision and Pattern Recognition, pp. 65–72.
27. R. S. Aygun and A. Zhang, 2001, "Stationary background generation in mpeg compressed video sequences," in Proc. IEEE Int. Conf. Multimedia Expo, pp. 701–704.
28. X. Yu, L. Duan, and Q. Tian, 2003, "Robust moving video object segmentation in the MPEG compressed domain," in Proc. IEEE Int. Conf. Image Process., pp. 933–936.
29. W. Zeng, W. Gao, and D. Zhao, 2003, "Automatic moving object extraction in mpeg video," in Proc. IEEE Int. Symp. Circuits Syst., pp. 524–527.
30. M Coimbra and M. Davies, 2004, "Segmentation of moving pedestrians within the compressed domain," in IEEE Int. Conf. Acoust., Speech, Signal Process., pp. 605–608.
31. B. U. Toreyin, A. E. Çetin, A. Aksay, and M. B. Akhan, 2004, "Moving region detection in

compressed video," in Proc. 19th Int Symp. Comput. Inf. Sci., pp. 381–390.

32. R. V. Babu, K. R. Ramakrishnan, and S. H. Srinivasan, 2004, "Video object segmentation: A compressed domain approach," IEEE Trans. Circuits Syst. Video Technol., Vol. 14, No. 4, pp. 462–474.
33. Wei Hong, John Wright, Kun Huang and Yi Ma, 2006, "Multiscale hybrid linear models for lossy image representation," IEEE Trans. on Image Process., Vol. 15, No.12, pp. 3655–3671.
34. P. Bhattacharyya and L. Ganesan, 1997, "An orthogonal polynomials based frame work for edge detection in 2D monochrome images," Pattern Recognit. Lett. Vol.18, No.4, 319–333.
35. R. Krishnamoorthi, 1999, "A unified framework based on orthogonal polynomials for edge detection, texture analysis and compression of images," Ph.D. thesis, Department of Computer Science and Engineering, IIT, Kharagpur.
36. R. Krishnamoorthi and P. Bhattacharya, 1998, "Color edge extraction using orthogonal polynomials based zero crossings scheme," International Journal of Information Sciences 112, No.1-4, pp.51–65.
37. R. Krishnamoorthi and P. Bhattacharyya, 1997, "A new data compression scheme using orthogonal polynomials," in: IEEE Proceedings on ICICSP, Singapore, Vol.1, pp. 490–494.
38. R. Krishnamoorthi, 2007, "Transform coding of monochrome images with statistical design of experiments approach to separate noise," Pattern Recognit. Lett. Vol. 28, No. 7, pp. 771–777.