THE LARGE-SCALE CROWD DENSITY ESTIMATION BASED ON SPARSE SPATIOTEMPORAL LOCAL BINARY PATTERN

Hua Yang, Hang Su, Shibao Zheng, Sha Wei and Yawen Fan

Institution of Image Communication and Information Processing,
Department of EE, Shanghai Jiaotong University, Shanghai, 200240, China
Shanghai Key Laboratory of Digital Media Processing and Transmission
Email:{hyang, Hmilyanjohn, sbzh, venessa724, yw_fan}@sjtu.edu.cn

ABSTRACT

Over the past decade, a wide attention has been paid to the crowd control and management in intelligent video surveillance area. This paper proposes a sparse spatiotemporal local binary pattern (SST-LBP) descriptor to extract the dynamic texture of the walking crowd with the application to crowd density estimation. Firstly, the sparse selected location is extracted, which is notably variant in temporal domain and scale invariant in spatial domain. Afterwards, considering the spatial and temporal symmetry, the authors propose a sparse spatiotemporal local binary pattern algorithm and utilize its statistical property to describe the crowd feature. Finally, the crowd features are classified into a range of density levels by adopting support vector machine. The experiments on real video show that the proposed SST-LBP method is effective and robust on the large-scale crowd density estimation. Compared with the other methods, the proposed method does not base on the premise that the background should be extracted perfectly, which is too complicated to implement in real surveillance.

***Index Terms*—** video surveillance, crowd density, local binary pattern, sparse point, support vector machine

1. INTRODUCTION

Over the past decade, crowd control and management has attracted wide attention from technical and social research disciplines along with the steady population growth and worldwide urbanization. A succession of fatal accidents, such as the Water Festival stampede in Cambodia, the Love Parade stampede in Germany and the Ivory Coast incident during the World Cup Qualifier in 2009, make the administrative organization put increasingly emphasis on preventing the large-scale crowd from losing control in crowded special event. To prevent the happening of such unfortunate events, the crowd phenomenon has become an essential research scene in social activities, especially for the video surveillance application. Among various parameters of crowd phenomenon, crowd density is an essential issue for crowd feature analysis, which is closely related to the security level [1].

Video surveillance, along with the rapid development of computer vision technology, has made the density estimation for the large-scale crowd possible. Compared with the conventional manual surveillance system, intelligent system could not only save a great deal of manpower, resources and time, but also achieve higher accuracy and a more objective parameter. Early in 1995, Davies [2] has started to estimate the crowd density according to the foreground occupied area ratio in the image. From the view of crowd feature extraction, recent study could be divided into two main classes of approaches: one is based on *pixel statistical feature* algorithm, and the representative features include edge orientation, blob size histograms, foreground area, perimeter-area ratio, and shape appearance [3], or the combination of these features [4]. In general, the accuracy of the methods based on pixel statistical feature decreases if the occultation becomes serious. Another category is based on the *texture analysis* algorithm, which bases on the hypothesis that the crowd with high density tends to appear as *find texture*, while the crowd with low density appears to be *coarse grain*. The typical texture features include grey level dependence matrix [5][6] and wavelet [7].

Compared with the pixel statistical feature algorithm, the methods based on texture could solve the occultation to some extent, but it is incapable to obtain a desirable result when the density is low. An important reason for its incapability is that the conventional methods are based on the static texture analysis and ignore the temporal information of the crowd, which could provide valuable information for the crowd. In this case, the texture of background or other objects will dominate the result if the density of the crowd is low. The problem could be solved by dynamic texture analysis method, which combines both spatial and temporal information.

Dynamic texture is a spatially repetitive, time-varying vi-

Thanks to Scientific and Technological Committee of Shanghai grant No. 10231204002, National Natural Science Foundation of China grant No. 61025005 and National Basic Research Program (973 Program) grant No. 2010CB731401, 2010CB731406 for funding.

978-1-61284-350-6/11/$26.00 ©2011 IEEE

sual pattern that forms an image sequence with certain temporal stationarity [8]. In other words, it implies that the conventional image texture is extended to the spatiotemporal domain. The walking crowd is one of the typical dynamic texture patterns and could be analyzed in this angle. In this paper, the authors propose a novel dynamic texture method which based on the sparse spatiotemporal local binary pattern (SST-LBP) descriptor. In contrast to the previous methods, the dynamic texture combines both the spatial and temporal properties. Furthermore, a crowd density estimation system is proposed base on the SST-LBP. Theoretical analysis and experimental study both prove that SST-LBP has distinct properties with different density level of crowd, which makes the system robust in all scale density level. Additionally, the system does not depend on individual detecting or tracking, and there is no need for background modeling, which is too complicated to implement with heavy crowded scene. Therefore, the system is more suitable and practical for large-scale crowd analysis.

The remainder of this paper is organized as follows: the structure of the system is presented in section 2. Then the details of the crowd estimation algorithm are discussed in section 3, containing the methods to extract the sparse selected locations, the SST-LBP code generation method considering the spatiotemporal symmetry, the dynamic texture analysis for the crowd and the density level mapping method based on support vector machine (SVM). In section 4, experimental results of our system will be shown and compared with the conventional methods. Finally, the conclusion is made in section 5.

2. SYSTEM ARCHITECTURE

The large-scale crowd density analysis is very helpful in finding potential risk in the moving pedestrians. In order to estimate the density of the crowd, both the temporal and spacial information are important. The temporal domain reflects the motion information which we are focusing and the spatial domain reflects the occultation information. Dynamic texture is one of the effective methods that could combine motion features with appearance features.

Dynamic texture has attracted the attention of the researcher for a long time, and the methods based on optical flow [9] are currently the most popular ones. However, the conventional methods consider textures to be homogeneous and describe the global features over the whole image, which limit the applicability of the dynamic texture.

LBP operator [10] is one of the powerful methods that could describe the local texture pattern with binary code. Its simple computation and high performance makes it attractive for many kinds of application. Zhao [11] proposed a method to extend the LBP to the temporal domain and utilized it to fulfill the facial expression recognition, but the crowd feature is rather more complicated than the facial feature. In this paper, we propose a novel SST-LBP methods to extract the crowd feature. The schematic diagram of the proposed system is shown in Figure 1, and in this paper, we focus on the module that aims to extract the crowd feature, because it is the essential part that affects the performance of the system.

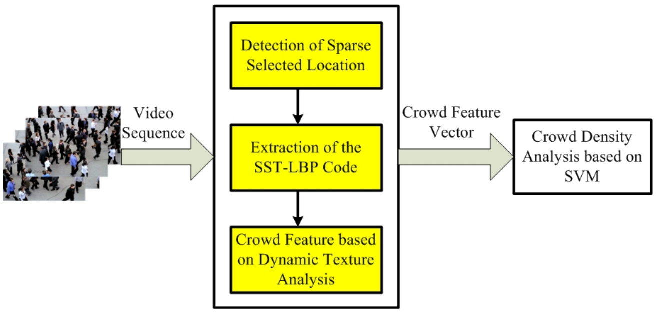
Video
Sequence
Detection of Sparse Selected Location
Extraction of the SST-LBP Code
Crowd Feature based on Dynamic Texture Analysis
Crowd Feature Vector
Crowd Density Analysis based on SVM

Fig. 1. System Structure of Crowd Density Estimation

The module is divided into three parts: the first stage is to search the sparse selected location in the spatiotemporal volume. The selected locations should be notable variance in temporal domain and scale invariant in spatial domain. In this paper, we utilize the Hessian and DOG detectors to extract the sparse selected locations. Next, we propose a sparse spatiotemporal local binary pattern (SST-LBP) algorithm to extract the dynamic texture of the walking crowd, and then employ spatial and temporal symmetry processing in order to enhance the adaptability of the feature. Afterwards, we propose a spectrum analysis method to avoid the dimension curse for the spatiotemporal signal. Then we employ the histogram to extract the statistical property for the SST-LBP. Finally, the support vector machine (SVM) is employed to build the relationship between crowd feature base on SST-LBP and the density level of the crowd.

3. CROWD DENSITY ESTIMATION BASED ON SST-LBP

3.1. Detection of the Sparse Selected Location

Some essential issues should be addressed in the feature extraction for the crowd density estimation: first, the feature should reflect the temporal information of the walking crowd, which means that the location with notable motion should be paid more attention. Additionally, the distortion caused by 3D to 2D projection should be considered, because object with the same sizes will be smaller in the image when it is farther from the camera. And in this paper, we employ the multi-scale analysis method in order to extract the scale invariant feature. In this case, the pixel is a selected location if and only if it is scale invariant in scale space and discontinuous in the temporal domain.

Considering of the success of the Hessian detector [12] for the interest points in spatial domain, we use it as the motion detector. The Hessian matrixes in temporal domain along with horizontal and vertical direction are shown in equation (1).

$$H_{xt} = \begin{bmatrix} \frac{\partial^2 I}{\partial x^2} & \frac{\partial^2 I}{\partial x \partial t} \\ \frac{\partial^2 I}{\partial t \partial x} & \frac{\partial^2 I}{\partial t^2} \end{bmatrix} \text{ and } H_{yt} = \begin{bmatrix} \frac{\partial^2 I}{\partial y^2} & \frac{\partial^2 I}{\partial y \partial t} \\ \frac{\partial^2 I}{\partial t \partial y} & \frac{\partial^2 I}{\partial t^2} \end{bmatrix}. \tag{1}$$

In function H_{xt}, $\frac{\partial^2}{\partial x^2}$ denotes second partial derivative in x direction and $\frac{\partial^2}{\partial t \partial x}$ is the mixed partial second derivative in t and x directions, and likewise the elements in H_{yt}. It is important to note that the derivatives are computed in the current iteration scale and thus are derivatives of an image smoothed by a Gaussian kernel, as discussed in [12], the derivatives could be scaled appropriately by a factor related to the Gaussian kernel, as equation (2) shows:

$$\begin{aligned} \frac{\partial^2 I}{\partial x \partial t} &\approx \left(\frac{\partial g_\sigma(x)}{\partial x} * \frac{\partial g_\sigma(t)}{\partial t}\right) * I(x,t), \\ \frac{\partial^2 I}{\partial y \partial t} &\approx \left(\frac{\partial g_\sigma(x)}{\partial y} * \frac{\partial g_\sigma(t)}{\partial t}\right) * I(y,t). \end{aligned} \tag{2}$$

The sample candidate locations that the hessian matrixes extract are shown in Figure 2, and the database we use in this paper is provided by Performance Evaluation of Tracking and Surveillance (PETS). A pixel is chosen as the candidate location if it is extracted from either H_{xt} or H_{yt} Hessian matrix, which have been marked as red point in Figure 2.

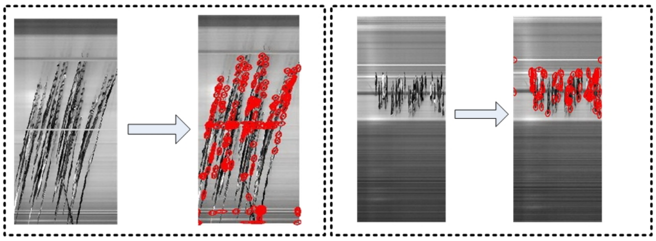

Fig. 2. Candidate Locations in the Temporal Domain

After that, the scale invariance of the candidate pixels are checked in the spatial domain. Motivated by the success of scale invariant feature transform (SIFT) [13], in this paper, we utilize the difference-of-Gaussian function (DOG operator) $D(x,y,\sigma)$ to extract the scale invariant selected location, and the DOG operator is defined as equation (3) shows:

$$D(x,y,\sigma) = (G(x,y,k\sigma) - G(x,y,\sigma)) * I(x,y), \tag{3}$$

where $G(x,y,\sigma)$ denotes the two dimensional Gaussian function with scale parameter σ. As David Lowe [13] has proved that

$$G(x,y,k\sigma) - G(x,y,\sigma) \approx (k-1)\sigma^2 \nabla^2 G. \tag{4}$$

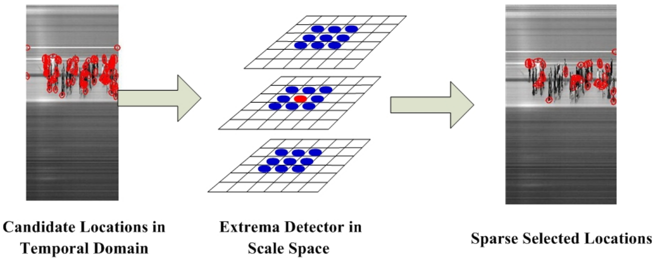
Candidate Locations in Temporal Domain
Extrema Detector in Scale Space
Sparse Selected Locations

Fig. 3. Schematic for Sparse Locations in Spatiotemporal Domain

The derivatives could also be estimated by the Gaussian kernel. The maxima and minima of this scale-space function are determined by comparing each pixel in the pyramid to its neighbors, both from the identical scale space and the neighboring space. The schematic of the sparse locations in spatiotemporal domain is shown in Figure 3. The blue point denotes the candidate point, and the blue points denotes the neighboring points in scale space.

The whole procedure to extract the sparse selected locations is illustrated in Figure 4, and in this paper, the X and Y dimensions denote the spatial domain, and T dimension denotes the temporal domain. For a specific pixel, it is first processed along the X-T plate and Y-T plate with the Hessian matrixes H_{xt} and H_{yt}. On condition that it has been detected as a candidate point from either of the matrix, it will be searched in the scale space to judge the scale invariance. The pixels that satisfies all these terms will be chosen as the selected location. Finally, the spatiotemporal volume is formed with the scaled current frame with the scaled parameter σ, for which scale space the extrema lies and the previous and latter frames. The red point of the spatiotemporal volume in Figure 4 denotes the selected location and the blue ones are the neighboring locations in the spatiotemporal domain.

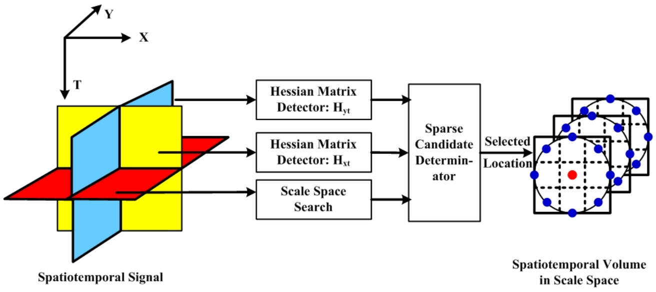
Y
X
T
Hessian Matrix Detector: H$_{yt}$
Hessian Matrix Detector: H$_{xt}$
Scale Space Search
Sparse Candidate Determin-ator
Selected Location
Spatiotemporal Signal
Spatiotemporal Volume in Scale Space

Fig. 4. Schematic for Sparse Locations in Spatiotemporal Domain

Another important reason for the sparse selected locations is that because the calculation amount for the three-dimensional signal $I(x,y,t)$ increases sharply compared with the 2D signal, the sparse location would make the algorithm more practical.

3.2. the Spatiotemporal Local Binary Pattern Extraction

The main difference between a dynamic texture and ordinary texture is that the notion of self-similarity central to conventional image texture is extended to the spatiotemporal domain [8]. The basic local binary pattern (LBP) operator is a gray-scale invariant texture primitive statistic, which has been shown excellent performance in the classification of various kinds of textures [10]. In this paper, we propose a spatiotemporal local binary pattern algorithm to describe the dynamic texture of the walking crowd.

The schematic for the dynamic texture of the walking crowd based on SST-LBP method is shown in Figure 5. When the sparse selected locations are extracted, a spatiotemporal volume $X \times Y \times T$ is formed with the corresponding location as its center. After that, the spatiotemporal LBP for the volume is calculated, as is shown in the blocks printed yellow in Figure 5. Compared with the static texture descriptor with LBP, the number of the patterns is quite large, which will lead to *dimension curse* in analysis. In this case, the SST-LBP need a transformation in feature space. The previous texture analysis method based on LBP always add the number with various weight, but in this paper, we utilize the spectrum analysis on the binary sequence, which will be shown reasonable next. Finally, in order to estimate the distribution of the SST-LBP, we employ the histogram as the estimation method.

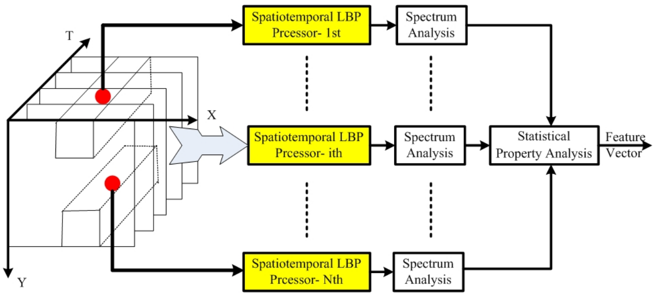
T
X
Y
Spatiotemporal LBP Prcessor- 1st
Spectrum Analysis
Spatiotemporal LBP Prcessor- ith
Spectrum Analysis
Statistical Property Analysis
Feature Vector
Spatiotemporal LBP Prcessor- Nth
Spectrum Analysis

Fig. 5. Schematic for Dynamic Texture Extraction of the Crowd with SST-LBP

Obviously, the essential part in the system is to calculate the sparse spatiotemporal local binary pattern code, which is the yellow blocks in Figure 5. The Figure 6 illuminates the detail of the process, and the blue points denote the neighboring pixels around the selected location. Suppose a spatiotemporal volume is constructed, the neighboring pixels are then determined via the predefined temporal and spatial distance. After that, we will sample along the helix in the cylinder volume. The red point denotes the selected location, and the yellow points denote the neighbors in spatial domain. The dark blue point denote the neighbors in the latter frame and the green ones denote the neighbors in the previous frame. The helix starts at the light blue point.

An important issue that should be addressed is the spatial and temporal symmetry, which means that the influence of the neighbors in the spatial domain should be identical, and the previous and the latter frames are of equal importance. In order to satisfy the symmetry, we interfuse the SST-LBP code in various spatial and temporal direction, as the equation (5) shows:

$$SST-LBP = \frac{1}{N}\sum_{i,j}\left\{\begin{array}{l}(SST-LBP_{org})\\ +SR(SST-LBP_{org})_i\\ +TR(SST-LBP_{org})_j\end{array}\right\}, \quad (5)$$

where $SST-LBP_{org}$ denotes the original SST-LBP code, as will be shown in Figure 7, $SR(\cdot)$ performs the spatial clockwise bitwise rotation on the original SST-LBP, and $TR(\cdot)$ performs the temporal reverse transformation on the original SST-LBP. Additionally, i and j denote number of different cases in spatial and temporal domain. N is the number of all cases, and in this equation $N = \sum(i+j+1)$.

A reasonable assumption in this paper is that since the spatial information would be more related to the level of occultation and the partial temporal information has been extracted in the Hessian detection step, the spatial points should have more influence. In this case, we sample more neighboring points in spatial domain, which will be discussed next as in Figure 7.

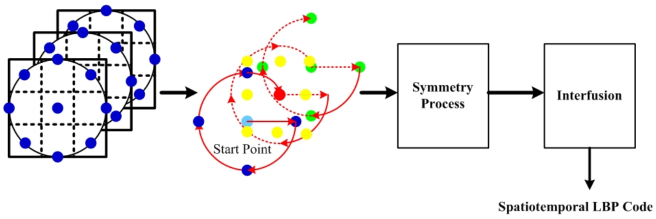

Fig. 6. Schematic for the Calculation of SST-LBP Code.

Figure 7 gives the overall computing procedure for the original spatiotemporal local binary pattern. Firstly, the neighboring pixels within a sphere with the radius R in the spatiotemporal domain are sampled. Afterwards, the binary value will be obtained with the value of the gray-level of the selected pixels as threshold. Finally, the forward SST-LBP code is produced by multiplying the binary values with weights given to the corresponding pixel and summing up the result. However, it need to be emphasized that the weight in this paper only represents the order of the binary elements, and the we will analyze the spectrum characteristics of the binary sequence. The original SST-LBP is calculated in equation (6).

$$SST-LBP = \sum_{i=0}^{N-1} s(g_p - g_s)\cdot w(p_i), \quad (6)$$

where $s(\cdot)$ denotes the step function, g_p and g_s denote the gray-scale of the corresponding pixel and the selected pixel. $w(p)$ denotes the weight of the pixels, where $w(p) = 2^i$. After the calculation of the original SST-LBP, the weight of the corresponding pixels would be rearranged and the other cases of SST-LBP code in equation (5) would be recalculated.

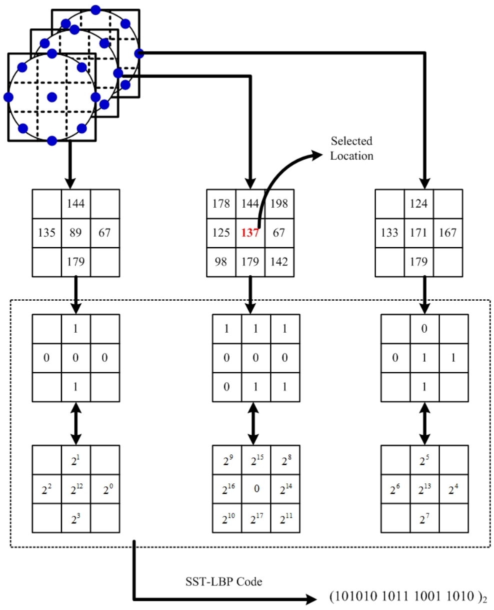

Fig. 7. Schematic for the Original SST-LBP Computing Procedure

3.3. Crowd Density Analysis based on Support Vector Machine

In this section, the relationship between the crowd feature vector and the output density is established, which is a typical regression problem. The support vector machine (SVM) [14] is an effective tool to solve the nonlinear regression estimation problem, and the traditional decision function of SVM is shown in equation (7):

$$f(\bar{x}) = (sign)(\sum_{i=1}^{l} \alpha_i K(\bar{x}_i, \bar{x}) + b), \tag{7}$$

where $\bar{x}_i$ are support vectors. And in this paper, the gaussian RBF function is employed as the kernel function.

The conventional SVM is designed for binary classification, but the crowd density estimation is a multi-class problem. Therefore, it needs to be extended for the multi-class problem. Considering the computation complexity and the feature vector property, we use the one-against-one method [15]. This method constructs $k(k-1)/2$ classifiers where each one is trained on data from two classes, and then utilizes the *MaxWins* strategy to decide the density level of the crowd.

4. SIMULATION AND ANALYSIS

In this paper, experiments are performed on the PETS 2009 dataset and results are compared to the pixels statistical feature and static texture feature. The crowd images are first separated into four groups according to the congesting degree of the crowds, which are defined by Polus [1] as free flow, restricted flow, dense flow and jammed flow.

We will first analyze the different SST-LBPs of various spatiotemporal volumes, and the schematic is shown in Figure 8. In this paper, there are mainly three categories of the selected spatiotemporal volumes: volumes from the serious occultation parts, volumes from the moving edge parts and volumes from internal or moving shadow where the gray-scale is relatively consistent. Afterwards, the extracted SST-LBP is regarded as a binary sequence and we transform it to bipolar binary sequence in order to keep the energy uniform. As the Figure 8 shows, the square wave that generates with SST-LBP of the consistent parts changes seldom, but the square wave that generates according to SST-LBP of the occultation changes more frequently, which is also reasonable in theory. Obviously, the SST-LBP extracted from the occultation volumes contains more high-frequency components. For the crowd with high density level, the occultation volumes dominate the spatiotemporal signal, as a result it would contain much more high-frequency components than the crowd of low density level.

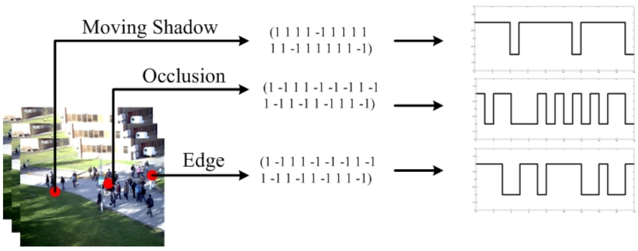

Fig. 8. Schematic for the Different SST-LBPs from Various Volumes

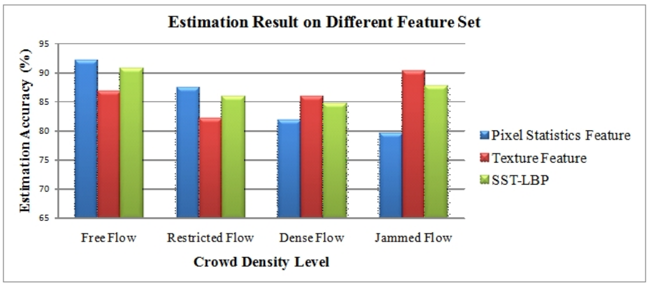

Fig. 9. Estimation Result on Different Feature Set

The estimation result on different feature set is shown in

Figure 9. The results based on the other feature, eg. pixel statistics and texture, are based on a premise that the background has be extracted perfectly, which is too complicated to implement in real surveillance. However, the SST-LBP feature does not need the background modeling. As it show, SST-LBP performs well on all-scale density level, since the feature between different density crowds differ more obviously. With the density level growing, the accuracy of the pixel statistics feature decreases significantly, because the area and edge ratio could not represent occultation between people effectively. And for restricted flow or dense flow, the estimation based on texture feature does not have a good result, because the texture feature for these density of crowd is not very distinct.

5. CONCLUSION

In this paper, we propose a system to estimate the crowd density level for the input video frame, which is essential for crowd management in intelligent surveillance system. In this proposed system, the sparse selected locations that are notably various in temporal domain and scale invariant in spatial domain are extracted firstly, and then we propose a sparse spatiotemporal local binary pattern algorithm, considering the symmetry in temporal and spatial domain. In order to avoid the dimension curse, we also propose a spectrum analysis method and employ histogram to extract the statistical property of the dynamic texture of the walking crowd. Finally, the crowd features are classified into a range of density levels by using support vector machine. In contrast to the previous methods, the SST-LBP combines both the spatial and temporal properties and there is no need for background modeling. Additionally, the system does not depend on individual detecting or tracking, which is too complicated to implement with heavy crowded scene.

6. REFERENCES

[1] A Schofer J. Ushpiz A. Polus, "Pedestrian flow and level of service," *J. Transportation Eng.*, vol. 109, no. 1, pp. 46–56, 1983.

[2] A.C. Davies, Jia Hong Yin, and S.A. Velastin, "Crowd monitoring using image processing," *Electronics Communication Engineering Journal*, vol. 7, no. 1, pp. 37 –47, Feb. 1995.

[3] Weina Ge and R.T. Collins, "Crowd density analysis with marked point processes [applications corner]," *Signal Processing Magazine, IEEE*, vol. 27, no. 5, pp. 107 –123, 2010.

[4] A.B. Chan, Z.-S.J. Liang, and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, 2008, pp. 1 –7.

[5] Ka Keung Lee Yangsheng Xu Xinyu Wu, Guoyuan Liang, "Crowd density estimation using texture analysis and learning," *IEEE International Conference on Robotics and Biomimetics*, pp. 214 – 219, 2006.

[6] G.; Liu Wei; Yan He Ping. Sen, "Counting people in crowd open scene based on grey level dependence matrix," *International Conference on Information and Automation*, pp. 228 – 231, 2009.

[7] V Marana A N. Verona V, "Wavelet packet analysis for crowd density estimation," *Proceedings of the IASTED International Symposia on Applied Informatics, Innsbruck, Austria*, pp. 535–540, 2001.

[8] Renaud Peteri Dmitry Chetverikov, "A brief survey of dynamic texture description and recognition," *Proc. Intl Conf. Computer Recognition Systems*, pp. 17–26, 2005.

[9] Renaud Péteri and Dmitry Chetverikov, "Dynamic Texture Recognition Using Normal Flow and Texture Regularity," pp. 223–230. 2005.

[10] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 971 –987, July 2002.

[11] Guoying Zhao and M. Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 6, pp. 915 –928, 2007.

[12] C. Schmid A. Zisserman J. Matas F. Schalalitzky T. Kadir L. Van Gool K. Mikolajczyk, T. Tuytelaars, "A comparison of affine region detectors," *International Journal of Computer Vision*, vol. 1-2, no. 65, pp. 43–72, 2005.

[13] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 2, no. 60, pp. 91 –110, 2004.

[14] O. Chapelle, P. Haffner, and V.N. Vapnik, "Support vector machines for histogram-based image classification," *Neural Networks, IEEE Transactions on*, vol. 10, no. 5, pp. 1055 –1064, Sept. 1999.

[15] Ulrich H.-G. Kres, "Pairwise classification and support vector machines," *Advances in kernel methods: support vector learning*, pp. 255–268, 2003.