Density Based Crowd Detection & Notifications

Dattaprasad S. Vispute[1]
[1]Department of Computer Engineering,
Late G.N. Sapkal college of Engineering India,

Hitesh H. Pawar[2]
[2]Department of Computer Engineering,
Late G.N. Sapkal college of Engineering India,

Mayur S. Nalnikar[3]
[3]Department of Computer Engineering,
Late G.N. Sapkal college of Engineering India,

Ravikumar K. Sonawane[4]
[4]Department of Computer Engineering,
Late G.N. Sapkal college of Engineering India,

Abstract: **We are designing new system for automatic pedestrian detection and tracking. Moving crowds was detected by classification between the background and foreground pixels. The human face is the only body part that can be robustly detected and tracked, so in this research study we present a method for tracking pedestrians by detecting and tracking their face rather than their full bodies. We describe the multi-scale person detector, which provides the dense detection score map in the cost function. We use the state of the art object detector and train it on extended face regions obtained by manual annotation of face rectangles in our training images of crowds. The high acceleration of crowd get detected by detecting motion of crowd.**

Keywords: crowd monitoring, motion detection, image processing based surveillance.

I. INTRODUCTION

The increasing demands on public transport networks have led to extensive deployment of CCTV systems to improve the safety and confidence of the travelling public. Some networks use CCTV in an “active” mode, where one or two human observers are responsible for permanently looking at the monitors to detect situations of interest. Given staffing resource limitations, there is a limit to what such operators can do. There are situations where operator intervention is immediately required. In such cases, automatic systems should also be able to direct the attention of operators through audible alarms, physical location etc. This could be referred to as an “Alarm Mode”. In either mode, the main concern is to decrease operator workload and increase passenger safety and site capacity

This paper presents a novel real-time abnormal motion detection scheme. The algorithm uses the macro block motion vectors that are generated anyway as part of standard video compression methods,These Motion features are derived from the motion vectors. Normal activity is characterized by the joint statistical distribution of the motion features, estimated during a training phase at the inspected site. During online operation, improbable-motion feature values indicate abnormal motion. Relying on motion vectors rather than on pixel data reduces the input data rate by about two orders of magnitude, and allows real-time operation on limited computational platforms.

1. *Problem Statements*

In high-density crowds some popular techniques such as background subtraction fail. We cannot deal with this problem in ways similar to simple object tracking, where the number of objects is not large and most part of the object are visible. Inter-object occlusion and self-occlusion in crowd situations makes detection and tracking challenging. In such situations we also cannot perform segmentation, detection as well as tracking also. We should consider it as single problem.

Fig.[1] Example of a large crowd

2. *Objective & Scope*

A. *Density Map*

Detecting and tracking people in crowded scenes is a crucial component for a wide range of applications including observation, group behavior modeling and mass crowd disaster prevention.

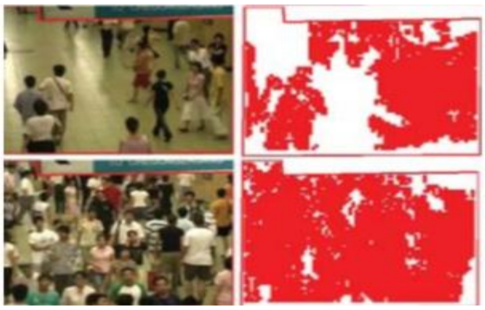


Fig.[2] Density Map

The consistent person detection and tracking in high crowds is a highly challenging task due to heavy occlusions, view variations and changing density of people as well as the ambiguous appearance of body parts.

B. *Motion Detection*

A motion detector detects moving objects, like people. A motion detector is often integrated as a component of a system that automatically performs a task or alerts a user of motion in an area.Motion Detection System works in prohibited area & detects the motions of area.

Fig [3]. Motion Detection

C. *Face Detection & Recognition*

Human face detection and recognition techniques play an important role in applications likevideo surveillance,face recognition and human computer interface and face image databases. Using color info in images is one of the various possible techniques used for face detection.

Fig.[4] Face Detection & Recognition

II. IMPLEMENTATION

1. *Haar Cascade Face Detection Algorithm*

The face detection Algorithm looks for specific Haar features of a human face &when one of these features is originate, the algorithm permits the face candidate to pass to the next stage of detection& tracking. A face candidate is a rectangular section of the original image called a sub-window or small windows. Generally these sub-windows have a fixed size (typically 24×24 pixels). This sub-window is often scaled in order to obtain a variety of different size faces. This algorithm scans the entire image with this window and denotes each respective section a face candidate. This algorithm uses an integral image in order to process Haar features of a face candidate in constant time. This uses a cascade of stages which is used to eliminate non-face candidates swiftly. Each stage consists of many different Haar features. Every feature is classified by a Haar feature classifier. It will produce an output which can then be provided to the stage comparator. It sums the outputs of the Haar feature classifiers and compares this value with a stage threshold to determine if the stage should be passed. If all stages are approved the face candidate is concluded to be a face. All these terms will be discussed in more detail in the following sections.

1.1 Integral Image

The integral image is defined as the summation of the pixel values of the original image. The value at any location (*x*, *y*) of the integral image is the sum of the image's pixels above and to the left of location (*x*, *y*). Figure 1 illustrates the integral image generation.

1.2 Haar Features

Haar features are composed of either two or three rectangles. Face candidates are scanned and examined for Haar features of current stage. The weight and size of each feature are generated using a machine learning algorithm from AdaBoost. The weights are constants generated by the learning algorithm. There is a variety in forms of features as seen below in Figure 2. Each Haar feature has a value that is calculated by taking the area of each rectangle, multiplying each by their individual weights, then summing the results. An area of each rectangle is easily found using the integral image. The coordinator of any angle of a rectangle can be used to get the sum of all the pixels above and to the left of that location using integral image. Using each angle of a rectangle, an area can be computed quickly as denoted by Figure 3. Since *L*1 is subtracted off twice it must be added back on to get the correct area of the rectangle. The area of the rectangle *R*, denoted as the rectangle integral, can be calculated as follows using the locations of the integral image: *L4-L3-L2+L1*

1.3Haar Feature Classifier

A Haar feature classifier uses the rectangle integral to calculate the value of a feature. The Haar feature classifier multiplies the weight of each rectangle by its area and the results are added together. Several Haar feature classifiers comprise a stage. Stage comparator totalities all the Haar feature classifier results in a stage and compares this summation with a stage threshold. It is also a constant obtained from the AdaBoost algorithm. Each stage will not have a set number of Haar features. The training individual data stages can have a varying number of Haar features. For example, Viola and Jones' data set used 2 features in first stage and 10 in the second. All they used a total of 38 stages and 6060 features. Our data set is based on the OpenCV data set which used 22 stages and 2135 features in total.

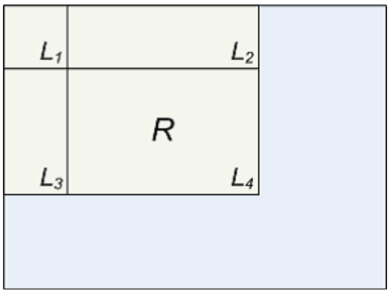
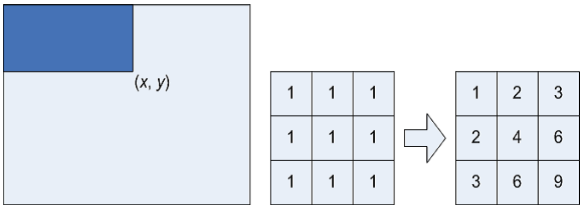


Fig.[5] Integral image generation.

The shaded region represents the sum of the pixels up to position (*x*, *y*) of the image. It shows a 3×3 image and its integral image representation.



1	1	1
1	1	1
1	1	1

1	2	3
2	4	6
3	6	9

Fig.[6] Calculating the area of a rectangle *R* is done using the corner of the rectangle: *L4-L3-L2+L1*.

1.4 Cascade

The Viola and Jones face detection algorithm eliminates face candidates quickly using a cascade of stages. It eliminates candidates by making stricter requirements in each stage with later stages being much more difficult for a candidate to pass. The selected candidates exit the cascade if they pass all stages or fail any stage. The candidate passes all stages if selected face is detected.

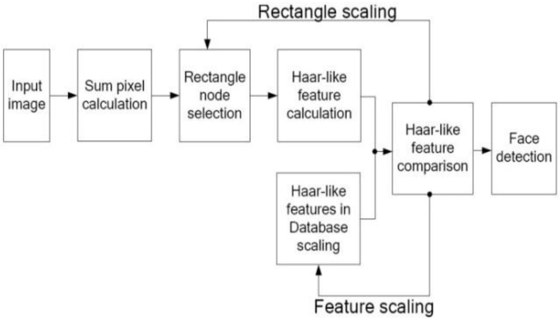


Fig.[7] Face detection Process

1.5 Architecture for Face Detection

We proposed architecture for a real-time face detection system. Figure 6 shows the overview of the proposed architecture for face detection. It consists of seven modules: image interface, frame grabber, image store, image scaler, classifier, display, and DVI interface. The image interface and DVI interface are implemented using ASIC custom chips with the FPGA board. The others are designed using Verilog HDL and implemented in an FPGA in order to perform face detection in real-time.

A. Frame Grabber

In frame grabber module, the frame grabber controller generates the control signals for controlling the A/D converter which converts the analog image signals into digital image data, and the sync separator which generates the image sync signals in the image interface module. The image sync signal and the color image data are transferred from the image interface module. The image cutter crops the images based on sync signals. These image data and sync signals are used in all of the modules of the Face detection system.

B. Image Store

The image store module stores the image data arriving from the frame grabber module frame by frame. This module transfers the image data to the classifier module based on the scale information from the image scalar module. The image of a frame is stored in a BRAM of the FPGA.

C. Image Scaler

The images are scaled down based on a scale factor by the image scaler module. The image scaler module generates and transfers the address of the BRAM containing a frame image in the image store module to request image data according to a scale factor. The image collection module transmits a pixel data to the classifier module based on the address of BRAM required from the image scalar module.

D. Classifier

The classifier module performs the classification for the face detection using Haar feature data. This module consists of the image line buffer, image window buffer, integral image window buffer, feature classifier, stage comparator, and feature training data. The face detection is performed by the Haar feature classification using an integral image. The integral image generation requires substantial computation. A general purpose computer of Von Neumann architecture has to access image memory at least *width×height*times to get the value of each pixel when it processes an image with *width×height*pixels. It may take a long latency delay every frame. In order to reduce memory access and processing time, we propose a specific architecture for the integral image generation. This architecture stores the necessary pixels for processing each pixel and its neighboring pixels together. It consists of the image line buffer, image window buffer, and integral image window buffer. Each buffer has its own controller. The image line buffer stores some parts of the image and its controller generates the control signals for moving and storing the pixel values. The image line buffer uses dual port BRAMs where the number of BRAMs is the same as that of the row in the image window buffer. Each dual port BRAM can store one line of an image.

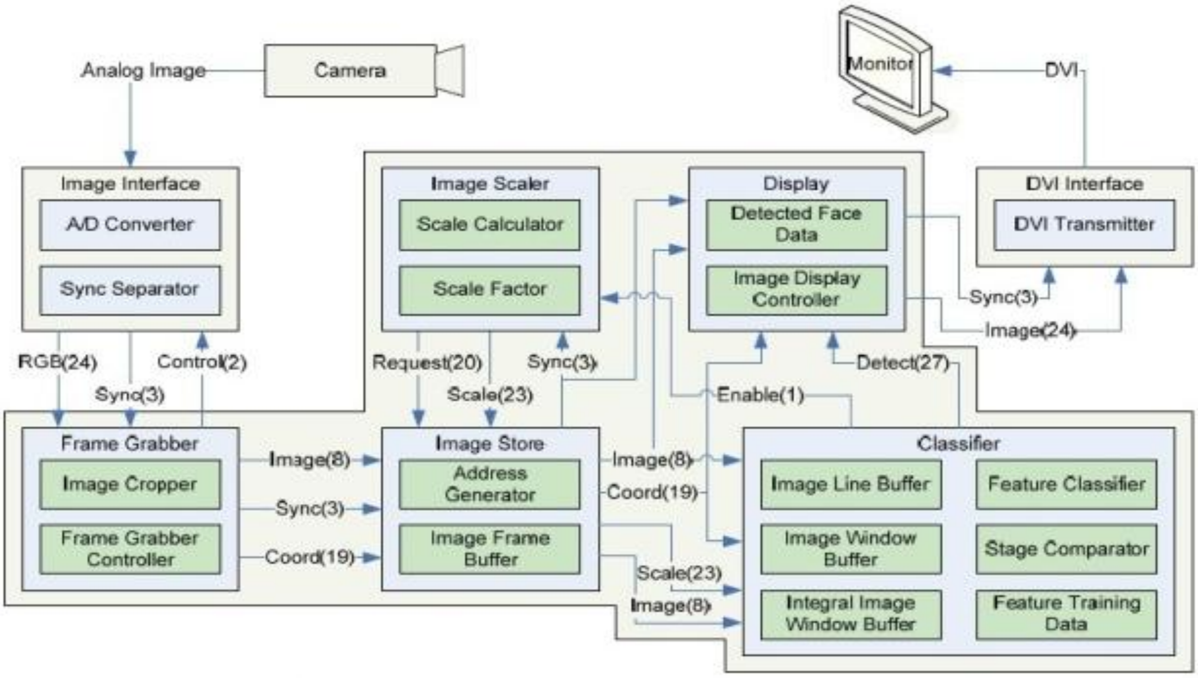


Fig.[8] Block diagram of proposed face detection system.

Thus, the *x*-coordinates of the pixels can be used as the address for the dual port BRAM. For the incoming pixel where the coordinate is (*x*, *y*), the image line buffer controller performs operations such as in (1), where *n* is the image row size of window, *p*(*x*, *y*) is the incoming pixel value, and *L*(*x*, *y*) represents each pixel in the image line buffer

$$L(x, y-k) = L(x, y-(k-1)), where\ 1 \le k \le n-2$$

$$L(x, y-k) = p(x, y), where\ k = 0$$

With these operations, the pixel values in the lines of an image are stored in dual port BRAMs. Since each dual port BRAM stores one line of an image, it is possible to get one pixel value from every line simultaneously.

2. *MOTION DETECTION ALGORTHM*

Detecting moving objects in a static scene Moving objects can be detected by applying Background Subtraction Algorithms Simplest method (frame differencing):

- Subtract consecutive frames
- Ideally it will leave only moving objects
- Following conditions effect the background subtraction:
  - Moving background (e.g. swaying of trees)
  - Temporarily stationary objects
  - Object shadows
  - Illumination variation

2.1 From video to motion vectors

Common video compression schemes exploit both the spatial and the temporal (frame-to-frame) redundancy present in the image sequence. A frame is either an intra-frame that is compressed as a full still image, eliminating its spatial redundancy, or an inter frame represented by macro-block displacement vectors relative to (say) the previous frame, and an error image. Intra-frames are generated at constant intervals, to allow random-access to the content, and to reduce accumulated errors. An intra-frame is also provided when there is a significant change in the scene (e.g., an editing cut), so that representation of the current frame in terms of the previous one is inefficient.

A motion vector Vi,j= {Vxi,j, Vyi,j } is associated with each Mh × Mw macro-block (i, j) in an inter-frame. In the current implementation Mh=Mw = 16 pixels. Generally, i ∈ {1, . . . ,imax},j ∈ {1, . . . , jmax}. The motion vector points to thelocation of the most similar MH ×Mw block in the previousframe. Inter-frame l is then represented by a setofn = imax × jmaxmotion vectors V l = {V li,j, i =1, . . . , imax, j = 1, . . . , jmax} associated with its macroblocks,or by their 2n components {V lxi,j, V lyi,j , i =1, . . . , imax, j = 1, . . . , jmax}.The difference between the current block and the reference block in the previous frame is compressed as part of the error image, and used for reconstruction. When the match between the current macro-block and the reference block is poor, the current block is compressed by itself and is referred to as an intra-block.

2.2 From motion-vectors to motion features

A small set Flof m << n features are derived from the set of motion vectors V l. Its regular probability distribution is estimated during training. In the course of online operation, the feature vector Flof the incoming frame is compared to the statistical model. If its probability is low, it is declared abnormal. The surveillance domain knowledge allows “manual” selection of the m features in Fl. The advantage of human-designed features with respect

to blindly generated ones is their clear conceptual meaning, providing insight and promoting testability and maintainability.
Let

$$|V_{i,j}^{l}| = \sqrt{(}(V_{xi,j)}^{l})^2 + (V_{yi,j}^{l})^2$$

$$\emptyset_{i,j}^{l} = \arctan(\frac{V_{yi,j}^{l}}{V_{xi,j}^{l}})$$

Respectively denote the magnitude and direction of the motion vector V li,j . The current implementation uses the following m = 5 features.

2.2.1 Total absolute motion

$$F_{TAM=\sum_{I,J}|V_{Ii,j}|}^{l}$$

This feature corresponds to the total motion in the scene. No distinction is made between the motion of ‘objects’ and the motion of, say, tree branches on a windy day.

2.2.2 Regional information

Dividing the frame into K rectangular sub-frames Ak, the area of dominant motion is obtained by:

$$F_{ADM=k^{*}=arg\ max\ k}^{l}\left(\sum_{i,j\in Ak}|V_{i,j}^{l}|\right)$$

This feature is the index of the sub-area of frame l with the largest sum of absolute values of motion vectors. Officially, this is the part of the frame with the largest absolute motion. In the current implementation= 9.The ratio between the total absolute motion in the dominant area $A_{k*}$of frame l and the total absolute motion $F^{l}_{TAM}$is an indicator of motion homogeneity within the frame. Officially,

$$F_{MH}^{l} = \frac{max_{k} \sum_{i,j\in Ak}\left|V_{i,j}^{l}\right|}{F_{TAM+\in}^{l}}$$

The addition of the small positive constant to the denominator prevents division by 0 in static frames.

2.2.3 Directional information

The range of motion directions {−π, π} is divided into R equal fractions of size Δϕ= 2π/R. Letr=0 . . . R − 1 be the angular fraction index. The principal motion direction is defined as the index of the most popular angular fraction:

$$F_{PDM}^{l} = r^{*} = arg\ max_{r}\sum_{i,j}(\emptyset_{i,j}^{l} - r\Delta\varphi| < \frac{\Delta\varphi}{2}$$

Where the sum is incremented if the arithmetic conditions satisfied. A measure for the dominance of the principal motion direction is obtained by the ratio of the total motion in the principal motion direction and the total absolute motion in the frame:

$$F_{DPM}^{l} = \frac{\sum_{i,j}\left|V_{i,j}^{l}\right|\left(\left|\emptyset_{i,j}^{l} - r^{*}\Delta\varphi\right| < \frac{\Delta\varphi}{2}\right)}{F_{TAM+\epsilon}^{l}}$$

III. EXPERIMENTS

A. Dataset

Considering the example of video to track images with respect their faces, Our data set consists of 15 feature-length movies: 12 Angry Men, Inception ,Se7en, Forrest Gump, Mission Impossible, Leon the Professional, Revolutionary Road, Legally Blond, Devil Wears Prada, The Curious Case of Benjamin Button (corresponding to the ID of F1 F15).The genres range from passion, funniness to criminality and thriller. That is split into two parts: pictures of F1–F3 as the training part and the rest 12 pictures of F4–F15 as the testing part. The training set has 1 327 face tracks, and the testing set has5012tracks.

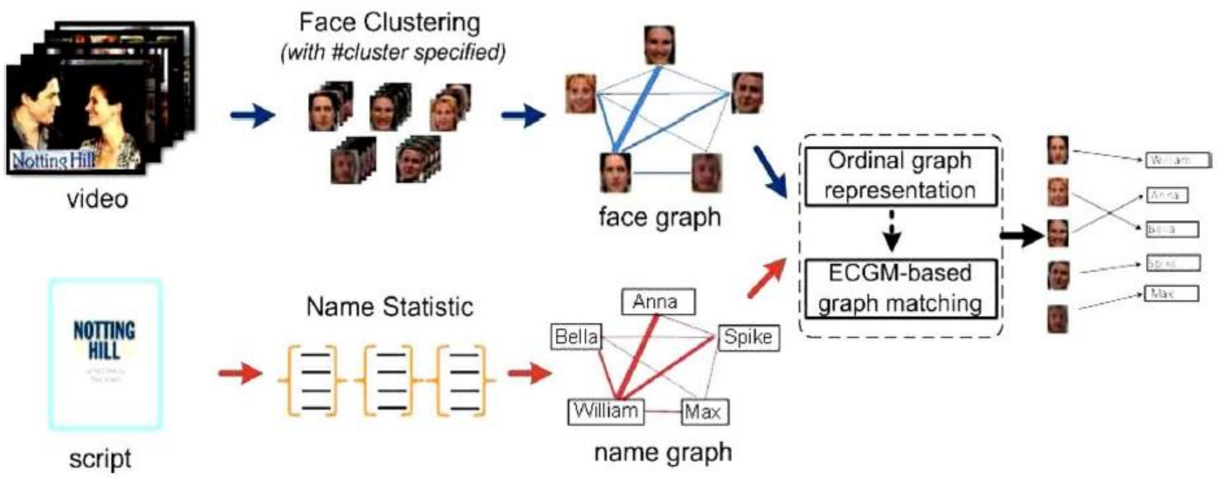
B. Experimental Results

1) Face Track Detection

We utilized a multi-view face tracker to detect and track faces on each frame. One typical movie contains up to 100 faces detected on all the frames, which are resulting from a few thousand tracks. Each face track covers about dozens of faces and the length of one face track ranges from several seconds to dozen of seconds. Since the face track detection usually follows Zhang’s work, we conducted a simple evaluation experiment by randomly selecting three 30-min clips from three movies F4, F5, F6 to retrieving the face track detection correctness

2) Face Track Clustering

In scheme 1, we follow the same face track clustering steps as in by evidence and graph partition by using script. Cluster clarity is used to evaluate the performance of face clustering: where is the set of face tracks in the cluster and is the set of face tracks with the label (character name). The two-step clustering method, which uses both the presence evidence and the script signs, makes the first step cluster focus on evidence clustering and leave the face-name correspondence issue to the second step. The evidence based method slanted to fail when picture tells lifetime story. For example, in the picture F4 and F15, the average cluster purity from appearance-based clustering is only 71% and 65%. Their stories go along the whole growth of the hero and heroine, where the appearance of characters changes intensely. Instead of selection faces from the same character cluster together, implementing the cluster number the same with the character number will deteriorate the clustering process and generate mixed clusters. For the appearance + script-based clustering method, the faces of the equal personality are expected to be gathered into numerous clusters, with each cluster about one period.



Fig[9] Framework of scheme 1: Face-name graph matching with #cluster pre-specified.

3) *Motion Analysis*

The suggested abnormal motion detection algorithm was successfully tested on pre-recorded videos. The code runs on a Core i3 2.8GHz PC with a Windows C# graphical user interface at a rate of 50 frames per second, without optimization. This is two times faster than the video rate. In this experiment, the camera captured a pedestrian pathway from a nearby building. Roughly 20 minutes of video were acquired. About 15 minutes of normal pedestrian traffic were used for training. The 2 minute long test sequence contained normal and abnormal activity. The movie was captured using an I-Ball (25fps) digital video camera. Several representative frames from the video sequence are provided. A few frames detected as abnormal are presented in Fig 2. The frames shown belong to a jumping episode, to a running and grass-crossing episode and to service vehicle episode. Note that the semantic descriptions (jumping, running grass-crossing) are provided merely for clarity. The operation of the algorithm is based on global motion features, without segmentation, detecting or any other attempt for semantic interpretation. These events are abnormal simply in the sense that similar motion patterns had not been observed (generally, have only rarely been observed) during the training session.

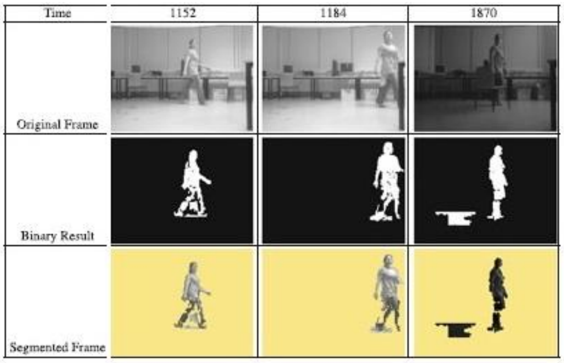


Fig.[10] Motion detection Of human body

IV. CONCLUSION

In this, we have proposed a crowd density estimation and prediction system for wide-area security. Haar Cascade based approach is applied to detect crowded areas. The number of people in a crowd is estimated by this algorithm. Compared to existing methods, the proposed method is a real time system for applications and the crowd density analysis algorithm can work properly in both low and high crowd density scenes. By calculating motion vectors we get the actual motions of human bodies& by using this we proposed motion detection system for highly prohibited area.

REFERENCES

1. Visual surveillance; crowd density analysis; motion detection; perspective distortion normalization. Paper 110939 received Aug. 5, 2011; revised manuscript received Feb. 16, 2012; accepted for publication Feb. 21, 2012; published online Apr. 27, 2012.
2. Crowd Analysis By Using Optical Flow and Density Based Clustering 18th European Signal Processing Conference (EUSIPCO-2010) Aalborg, Denmark, August 23-27, 2010
3. Motion detection and tracking based on level set algorithm Control, Automation, Robotics and Vision Conference, 2004. ICARCV 2004 8th (Volume:1) 6-9 Dec. 2004
4. MA Ruihua, LI Liyuan, HUANG Weimin, *et al.* On Pixel Count Based Crowd Density Estimation for Visual Surveillance[C]// Proceedings of 2004 IEEE Conference on Cybernetics and Intelligent Systems: December 1-3, 2004. Singapore, 2004: 170-173
5. PARAGIOS N, RAMESH V. A MRF Based Approach for Real-Time Subway Monitoring[C] Proceedings of 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition: December 8-14, 2001. Kauai, HI, USA, 2001: 1034-1040.