**PEDESTRIAN DETECTION AND COUNTING
IN SURVEILLANCE VIDEOS**

**A Thesis presented to
the Faculty of the Graduate School
at the University of Missouri**

**In Partial Fulfillment of the Requirements for the Degree
Master of Science**

**By
Di Wu
Dr. Zhihai He, Thesis supervisor**

December 2013

Pedestrian Detection and Counting in Surveillance Videos

Presented by Di Wu,

a candidate for the degree of Master of Science,

and hereby certify that, in their opinion, it is worthy of acceptance.

Zhihai He, Ph.D., Department of Electrical and Computer Engineering

Tony Xu Han, Ph.D., Department of Electrical and Computer Engineering

Jianlin Cheng, Ph.D., Department of Computer Science

ACKNOWLEDGEMENTS

First of all, I am heartily thankful to my advisor, Dr. Zhihai He, who gave me a lot of theoretical and technical support on my thesis work. Dr. He is very kind and patient to students. His supervision and encouragement enabled me to accomplish my graduate work.

Besides my advisor, I would like to also express my sincere gratitude to Dr. Marjorie Skubic. As a research assistant in the Center for Eldercare and Rehabilitation Technology, she gave me lots of support and valuable guidance.

As a member of the Video Processing and Networking Lab, I also like to thank York Chung, Xiaobo Ren, Chen Huang and Zhi Zhang. They helped me with a lot of academic advice in my research and we enjoyed encouraging each other.

Finally, I would like to acknowledge my parents and my girlfriend Meng Niu, who have been my spiritual supporting.

Table of Contents

ACKNOWLEDGEMENTS	ii
List of Figures	v
List of Tables.....	viii
ABSTRACT.....	ix
CHAPTER 1	1
INTRODUCTION AND BACKGROUND.....	1
1.1 Research Topic and Objectives	1
1.2 Challenges	2
1.3 Background and Related Work.....	3
1.3.1 Background Subtraction.....	4
1.3.2 Shadow Removal	4
1.3.3 Classifiers.....	6
1.3.4 Features	7
1.4 Overview of Our Method	7
CHAPTER 2	9
BACKGROUND SUBTRACTION AND SHADOW REMOVAL	9
2.1 Background Modeling and Updating	10
2.2 Shadow Removal.....	11
2.2.1 Necessity for Shadow Removal	11
2.2.2 HSV.....	12
2.2.3 RGB to HSV conversion.....	13
2.2.4 Condition of Shadow Detection.....	14
CHAPTER 3	16
HEAD-SHOULDER DETECTION	16
3.1 Foreground Curve Analysis.....	16
3.1.1 Introduction to Curve Analysis	16
3.1.2 Implementation	17
3.2 HOG features.....	20
3.3 Bag of Words	22
3.4 K-means clustering.....	23
3.5 Support Vector Machines (SVM)	24

3.6 Training Patches Specification	26
CHAPTER 4	32
TEMPORAL ANALYSIS FOR PEDESTRIAN COUNTING	32
4.1 Counting Rules	32
4.2 Implementation.....	33
CHAPTER 5	37
EXPERIMENTAL RESULTS	37
5.1 Experimental Dataset	37
5.2 Discussion	42
CHAPTER 6	45
CONCLUSIONS AND FUTURE DIRECTION	45
REFERENCES	46

List of Figures

Figure	Page
Figure 1: An example of difficulty in background subtraction in training video.	3
Figure 2: An example of two person too close to each other in testing video.	3
Figure 3: The work flow of person counting system.	8
Figure 4: HSV model is illustrated as a cylindrical.	12
Figure 5: Result of Shadow Removal.	15
Figure 6: An overview of head-shoulder detection procedure.	16
Figure 7: An example for Curve Analysis. The rectangle template scan	17
Figure 8: One person passing through counting zone.	17
Figure 9: Image of one person pass through counting zone after processing.	18
Figure 10: Plot of gray pixels percentage from a person walking in counting zone. ..	18
Figure 11: Four people clamped together.	19
Figure 12: Image of four people case after processing.	20
Figure 13: Plot of gray pixels percentage for four people case.	20
Figure 14: Variants of HOG descriptors. (a) A rectangular HOG (R-HOG) descriptor with 3×3 blocks of cells. (b) Circular HOG (C-HOG) descriptor with the central cell divided into angular sectors as in shape contexts. (c) A C-HOG descriptor with a single central cell [36].	21
Figure 15: Maximum margin hyper plane and margins for an SVM trained	26

Figure 16: Average edge maps (like omega) of head-shoulder samples in the training set.	27
Figure 17: Typical head-shoulder samples with edge map.	28
Figure 18: Some samples of positive training set.	28
Figure 19: Some samples of negative training set.	29
Figure 20: The first quartile is 26.75. The third quartile is 30. The mean is 28.2. The median is 28.5.	29
Figure 21: The first quartile is 34.5. The third quartile is 40. The mean is 37.05. The median is 36.5.	30
Figure 22: The first quartile is 51. The third quartile is 60.25. The mean is 55.5. The median is 53.5.	30
Figure 23: Result for one person.	30
Figure 24: Result for 4 person.	31
Figure 25: Position matrix plot for the first 2000 frame.	34
Figure 26: Position matrix plot for frame 740 to 810.	34
Figure 27: Position matrix plot for frame 1540 to 1700.	36
Figure 28: Some samples look vague to distinguish it is with head or without head. .	38
Figure 29: EBW1 Original frame	38
Figure 30: EBW1 frame after background subtraction.	38
Figure 31: There is a frame with 3 person passing through.	39
Figure 32: Curve analysis plot for the frame with 3 person passing.	39
Figure 33: The result for EBW2	40

Figure 34: The result for EBW3	40
Figure 35: The result for test video on vegetable market.	41
Figure 36: The curve analysis plot for test video.....	42
Figure 37: Wrong detected patches in test video	44

List of Tables

Tables	Page
Table 1: Interesting companies and their web sites.	1
Table 2: Frame 7706 to 7721 Position Matrix.	32
Table 3: Frame 763 to 781 Position Matrix.	34
Table 4: Frame 1631 to 1649 Position Matrix.	35
Table 5: Test training data for dataset on EBW.	37
Table 6: Test video EBW2 and EBW3.	37
Table 7: Test training data for dataset 2.	41
Table 8: Table 7: Test another video with similar scene.	41
Table 9: Patches TP, TN, FP, FN.	42

ABSTRACT

Pedestrian detection and counting have important application in video surveillance for entrance monitoring, customer behavior analysis, and public service management. In this thesis, we propose an accurate, reliable and fast method for pedestrian detection and counting in video surveillance. To this end, we first develop an effective method for background modeling, subtraction, update, and shadow removal. To effectively differentiate person image patches from other background patches, we develop a head-shoulder classification and detection method. A foreground mask curve analysis method is to determine the possible position of persons, and then use a SVM (Support Vector Machine) classifier with HOG (Histogram of Oriented) feature and bag of words to detect the head-shoulder of people. Based on the foreground detection and head-shoulder classification at each frame, we develop a person counting algorithm in the temporal domain to analyze the frame-level classification results. Our experiments with real-world surveillance videos demonstrate the proposed method has achieved accurate and reliable pedestrian detection and counting.

CHAPTER 1

INTRODUCTION AND BACKGROUND

1.1 Research Topic and Objectives

People counting and human detection have been an important problem in visual surveillance. During recent years, people counting systems have been widely used in security applications and marketing research, including customer behavior analysis, visitors flow estimation, and pedestrian traffic management [1, 3, 6, 11]. Such systems provide valuable information not only for optimizing the working hours but also for evaluating the security situation of a monitoring area. Moreover, several companies are developing products to provide such services. Table 1 lists a few of these companies and their services in the field of pedestrian tracking and surveillance.

Table 1: Commercial people counting products and services.

Name of Company	Web Site	Field
Acorel	http://www.acorel.com	People counting systems in public transport and shopping center
Video Turnstile	http://videoturnstile.com	People counting systems based on video processing
Infodev	http://www.infodev.ca	Counting people in building
SPSL	http://customercount.com/	People counting, people behavior analysis
Irisys	http://www.irisys.co.uk/people-counting	People counting systems based on video processing

This thesis targets at the problem of counting the number of pedestrians walking through a door or corridor. The system will record the total number of visitors at the entrance of a building. Our approach has three major steps: 1) to extract certain features

which encode frames from videos; 2) to develop a detector, i.e., a classifier, to determine whether a certain region in a frame is a person, and 3) to develop a people counter in using fuzzy matrices.

1.2 Challenges

Different illumination levels at different times of day and based on both the outdoor and indoor environment have a serious effect on the image captured. One notable effect is the image's pixel intensity even when the scene is always stationary and in the same location. The image captured by the camera at such a location will be different depending on the illumination level. Therefore, a stable illumination level is very important. It is very difficult to process the image with a large variation of the illumination levels for the environment that will also making the variation of the image captured by the camera become very large. Apart from the illumination condition of the environment, another factor that has a very serious effect on the image processing is the color similarity between the background and the clothes of people are wearing as shown in Figure 1. The third challenging issue we need to address in person counting is the difficulty in segmentation when two people are too close to one another with not enough space in the front and rear area between them, which Figure 2 shows.

Figure 1: An example of difficulty in background subtraction.

Figure 2: An example of two persons too close to each other.

1.3 Background and Related Work

People counting is a crucial and challenging problem in visual surveillance [16, 51, 53, 55]. It is very useful in the video surveillance field where a high accurate automatic counting of pedestrian flow through a gate or a door is essential to surveillance, traffic control and sales projection based on walk-in statistics (i.e., number of customers). The early automatic counting methods, such as turnstiles, rotary

bars, and light beams had one inherent problem: they only allowed one person through the gate at one time. To overcome this limitation, many methods based on video content analysis were proposed. [14, 18, 33, 38, 50] to address the problem of estimating crowd density and Marana [38] used fractal dimension to characterize levels of people congestion in images of crowds as well as counting the precise number of pedestrians. Bartolini [36] and Paragios [12] considered some specific situation such as getting in or out of a bus and train. However, the occlusion problem creates low accuracy be low. Weina Ge [25] proposed a marked point process model to detect and count people in crowds. Rossi [37] and Chen [56] mounted a camera vertically with respect to floor plane, so the multiple occlusions could be avoided.

1.3.1 Background Subtraction

A number of efficient background subtraction algorithms [32, 49, 52] have been developed in the literature [28]. For background modeling and subtraction, models based on non-parametric kernel density estimation [2], mixture of Gaussians [10], least median of squares (LMeds) [13], eigen-backgrounds [8] and illumination distribution analysis [30] have been proposed.

1.3.2 Shadow Removal

Existing methods for shadow detection include pixel-level classification in a HSV color space [30], gradient filtering [58], edge analysis [54], etc. A rather novel method [29] detects shadows and moving objects based on sound physical models with good efficiency. All four assumptions (such as a light source causing a cast shadow has a certain extent) are used at the same time to detect image region changes indicated by movement of the cast shadows [27]. Because the chromaticity information is not

affected by the change of illumination in some cases, a shadow region can be detected by selecting a region that is darker than its neighboring regions but has similar chromaticity information. According to this illumination invariant property of chromaticity, several efficient methods [23, 24] have been developed to detect shadows for color images efficiently. Tian *et al.* [41] used image information theory to deduce a tricolor attenuation model and employed blackbody irradiance to estimate its parameters. Shadows could then be detected shadows based on their new model. Tsai [42] presented an automatic property-based approach for the detection and compensation of shadow regions with shape information preserved in complex urban color aerial images for solving problems caused by cast shadows in digital image mapping. Lu et al. [43] first sampled shadow pixels based on the estimated shadow direction, afterwards, three shadow attributes were calculated based on sampled shadow pixels, but this involves significant processing time. In [44], moving shadows were detected by using their motion characteristics and the underlying physics based on the color segmentation; however, this method had poor accuracy since it could not distinguish shadows from black objects. Cucchiara et al. [20] proposed an approach which exploits color information for both background subtraction and shadow detection to improve object segmentation and background update.

Overall, shadow removal the existing shadow removal methods can be divided into two categories: model-based method and feature-based method. The model-based method generally uses some prior knowledge such as scene, object, illumination condition, and so on, through which can be matched using the angles of the moving

objects in order to detect shadows. Unfortunately, the prior knowledge is usually very difficult to obtain. This kind of method also has a long processing time and it is minimally used in practice. The feature-based method as proposed in references prefers using a shadow's brightness, color information, saturation, texture, and geometric features directly to detect shadows. This kind of method is commonly used, although it also has drawbacks. For example, if the color feature method is adopted, when the object and shadow have a similar color, then the invariant color features are not applicable, and the shadows cannot be differentiated. To avoid drawbacks of the feature-based method, this thesis proposes a shadow detection approach with elimination based on shadow position and edge attributes after HSV shadow removal.

1.3.3 Classifiers

Recently, learning based people detection methods using different classifiers have become more and more popular. The basic idea is to use a classifier to group the features extracted from the vehicle images into a positive class (person region) or negative class (non-person region). Various techniques [4, 17, 19] have been constructed as fast and reliable person detection techniques for surveillance applications. Classification techniques can for example be applied to decide if a given image region contains a person.

SVM has also been widely applied for object detection recently [40, 45]. SVM is a pattern classification algorithm which minimizes an upper bound on the generalization error, while other classifiers are trying to minimize the average training error. Tests have also proved that SVM can work well even in high dimensional space.

Nakajima et al. [19] adopted Support Vector Machines to classify RGB color histogram, shape histogram and local shape features to detect people. Gravrila [4] used a tree based classifier to represent possible shapes of pedestrians. Griebel et al. [17] used dynamic point distribution models. An alternative to modeling the appearance of an entire person is to design detectors for specific body parts and combine the detector results. The idea of learning part detectors using Ada-Boost and a set of weak classifiers is presented in [39]. A learning approach is then being used to combine the set of weak classifiers with body detectors, which are further combined using a probabilistic person model. All these approaches require a fair amount of training data to learn the parameters of the underlying model. Although these classifiers are robust to limited occlusions, they are not suitable to segment a group of people into individuals.

1.3.4 Features

A performance comparison of the HOG (Histogram of Oriented Gradients) feature and another two other popular features: the Haar feature [21] and SIFT descriptor [22] for head and shoulder detection have been defined in [57]. In their work, the HOG [31] feature performed much better than the other two features. The method developed in this research, the SIFT feature, is typically used for matching local regions while Gist is typically computed over the entire image. Therefore, HOG features are much more suitable for our system.

1.4 Overview of Our Method

As shown in Figure 3, our system consists of three main components: background subtraction, person detection, and pedestrian counting. All three components are combined into a tightly coupled framework. The background

subtraction includes the background modeling and updating. Shadow removal will be discussed in Chapter 2. Our person detection is restricted to head and shoulder detection, and will be explained in Chapter 3. The person detector is the most significant component in this system. As for the final component, the pedestrian counting, it performs as a partial person tracking. In the Chapter 4, we will explain this in more detail.

Figure 3: The work flow of person counting system.

CHAPTER 2

BACKGROUND SUBTRACTION AND SHADOW REMOVAL

Background subtraction, namely, segmenting objects from a background, identifies moving objects from the portion of a video frame that differs significantly from a background model. It is always the first and enabling step for many high-level analysis tasks, such as people tracking and activity recognition [3, 7, 9, 15, 26]. For background modeling and subtraction, the segmenting process needs to be accurate and robust, and it also need to immediately reflect sudden scene changes such as objects being moved.

To design an efficient background subtraction algorithm for outdoor living environments, there are three obvious external factors that need to be carefully addressed. Firstly, changes in lighting conditions can be caused by changing sunlight. To handle changing lighting conditions, adaptive background models are needed [5, 47]. Secondly, both in indoor environments and in outdoor environments, a strong shadow is always a big problem. In an outdoor environment, the sunshine will cast strong shadows of a moving person on the floor. However, in an indoor environment, the room is often illuminated by few lighting sources, such as light from a window. Thus, shadow removal becomes a significant challenge and a necessary task to be handled. Section 2.2 will discuss a simple and efficient method for shadow removal based on the HSV color space. Thirdly, another important challenge is caused by human activities. For

accurate background subtraction, it is necessary to update the background model for incorporating these changes.

2.1 Background Modeling and Updating

Frames 1 to 20 are selected for use in background modeling. More specifically we extract features in the RGB color space [48] and calculated the average for each RGB component.

In a dynamic video scene, it is important to continuously update the background model by incorporating background changes. The most common used method used to update a background first determines whether or not the background of an object or image area remained stationary for a certain period of time. For adaptive background updating, past Δ frames are used for background modeling. At each pixel location, one must compute the average values of its RGB components in the past Δ frames and denote them as:

$$avgR = \frac{avgR \times \Delta + R}{\Delta + 1} \quad (2.1.1)$$

$$avgG = \frac{avgG \times \Delta + G}{\Delta + 1} \quad (2.1.2)$$

$$avgB = \frac{avgB \times \Delta + B}{\Delta + 1} \quad (2.1.3)$$

One must also calculate the standard deviations of the color components at each pixel as:

$$sd = (R - avgR)^2 + (G - avgG)^2 + (B - avgB)^2 \quad (2.1.4)$$

The next step is to define a threshold based on standard deviation and establish the following decision rules for foreground and background: (1) if the standard deviation is larger than threshold, it will be viewed as foreground and it will not update the average

values with the past Δ frames. (2) If the standard deviation is smaller than the threshold, it will be viewed as a background, and it will update the average values of its RGB components with the past Δ frames. (3) If a pixel has been viewed as a foreground, it will be checked as to whether it is a shadow pixel. However, no matter it is a shadow pixel or not, it will not update the average values with the past Δ frames.

At this point, past Δ frames are used to update the background model. For accurate silhouette extraction, Δ must be small so that the background can be quickly updated. However, error can occur when Δ is small, which can lead to the human body being mistakenly updated as background if the person does not move for a while as when sitting still on a chair for a few minutes.

2.2 Shadow Removal

2.2.1 Necessity for Shadow Removal

In the current method of moving target detection, the most serious problem is that shadow can be easily mistaken as moving targets.

In a video surveillance, shadow pixels travel together with the moving target, so all the moving points in the video surveillance and shadows that accompany with the movement are likely to be regarded as moving targets. Moreover, shadows and objects connect to each other, which can cause serious shape distortion of moving objects. Due to this, the research's newly developed method generated strong interference forcing reliance on geometric properties of the object classification and determining the position of moving object. Therefore, if the shadow connects with two different objects, two different objects will be viewed as one object in this situation.

2.2.2 HSV

This thesis proposes to use HSV color space to detect and remove shadow as a way to manage this problem. HSV defines a type of color space and stands for hue, saturation and value. In HSV, hue represents color. In this model, hue is an angle from 0 degrees to 360 degrees. Saturation indicates the range of grey in the color space. It ranges from 0 to 100% and is calculated from 0 to 1. When it is '0', the color is grey and it is '1', the color is a primary color. A faded color is due to a lower saturation level, which means the color contains more grey. Value is the brightness of the color and varies with color saturation. It ranges from 0 to 100%. When it is equal to 0, the color space will be totally black. With an increase in value, the color space brightens up and shows various colors.

Figure 4: HSV model is illustrated as a cylindrical.

Figure 4 shows the HSV model to be a cylindrical or conical object. When it is represented in geometry, hue is represented by angular dimension, starting at the red primary range at 0°, passing through the green color range at 120° and the blue primary color range at 240°, and then wrapping back to red at 360°. The central vertical axis is comprised of the neutral, achromatic, or gray colors, ranging from black at value 0 on the bottom, to white at value 1 on the top.

2.2.3 RGB to HSV conversion

From the GRB triplet, one must first find the maximum and minimum values to convert from RGB to HSV. Saturation ,S, is then:

$$S = \frac{(\max - \min)}{\max} \quad (2.2.1)$$

and V, is:

$$V = \max \quad (2.2.2)$$

The hue, H, is then calculated as follows. First calculate R', G', and B' :

$$R' = \frac{\max - R}{\max - \min} \quad (2.2.3)$$

$$G' = \frac{\max - G}{\max - \min} \quad (2.2.4)$$

$$B' = \frac{\max - B}{\max - \min} \quad (2.2.5)$$

If saturation, S is 0 then the hue is undefined. In this case, the color has no hue because it is monochrome. Otherwise:

If R = max and G = min

$$H = 5 + B' \quad (2.2.6)$$

Otherwise, if R = max and G $\neq$ min

$$H = 1 - G' \quad (2.2.7)$$

Or if G = max and B = min

$$H = 1 + R' \quad (2.2.8)$$

Or if G = max and B $\neq$ min

$$H = 3 - B' \quad (2.2.9)$$

Or if B = max and G = min

$$H = 3 + G' \quad (2.2.10)$$

Or if $B = \max$ and $G \neq \min$

$$H = 5 - R' \quad (2.2.11)$$

Hue, H , is then converted to degrees by multiplying by 60 giving HSV with S and V

between 0 and 1 and H between 0 and 360. Finally, we calculate HSV as follows;

$$v = \max \quad (2.2.12)$$

$$s = \begin{cases} 0 & \text{if } \max = 0 \\ \frac{\max - \min}{\max} = 1 - \frac{\min}{\max} & \text{otherwise} \end{cases} \quad (2.2.13)$$

$$H = \begin{cases} 0 & \text{if } \max = \min \\ 60 \times \frac{G - B}{\max - \min} + 0 & \text{if } \max = R \text{ and } G \geq B \\ 60 \times \frac{G - B}{\max - \min} + 360 & \text{if } \max = R \text{ and } G < B \\ 60 \times \frac{B - R}{\max - \min} + 120 & \text{if } \max = G \\ 60 \times \frac{R - G}{\max - \min} + 240 & \text{if } \max = B \end{cases} \quad (2.2.14)$$

2.2.4 Condition of Shadow Detection

We define the conditions of shadow detection for each pixel. Those pixels in accordance with the conditions will be treated as shadow, and the others will be viewed as moving target.

$$bg(i, j) = \begin{cases} 1 & \text{if } \frac{g^1(v)}{g(v)} > \theta_1 \text{ and } \frac{g^1(v)}{g(v)} < \theta_2 \text{ and } g^1(s) - g(s) < \theta_3 \text{ and } g^1(h) - g(h) < \theta_4 \\ 0 & \text{otherwise} \end{cases} \quad (2.2.15)$$

In this formula, $0 < \theta_1 < \theta_2 < 1$, and $\theta_1, \theta_2, \theta_3$ and θ_4 are all experienced parameters.

The $g(h)$, $g(s)$, $g(v)$ correspondingly represent the HSV components of the HSV model frame, and $g^1(h)$, $g^1(s)$, $g^1(v)$ correspondingly represent the HSV components of a new frame. To achieve optimal results, it is necessary to adjust those parameters according to different video surveillance systems. The result is shown in Figure 5.

There are four picture in Figure 5: The picture in the top left corner is the original

picture; the picture in the top right corner is the picture after background subtraction in which there is an obvious shadow; the picture in the bottom left is the picture after shadow removal; the picture in bottom right is the picture without shadow and filled with original pixels.

Figure 5: Result of Shadow Removal.

CHAPTER 3

HEAD-SHOULDER DETECTION

Our head-shoulder detection follows the procedure shown in Figure 6. In this chapter, the details of each step will be discussed.

Figure 6: An overview of head-shoulder detection procedure.

3.1 Foreground Curve Analysis

Foreground curve analysis is the method we used in order to count the objects in one frame after the input frame is processed by background and foreground subtraction. This method was developed in [34].

3.1.1 Introduction to Curve Analysis

A template is developed to scan the counting zone of the segmented image. The template is shown at the most left hand side of the counting zone, where the rectangle is found. The template will scan the counting zone of the image from left to right. When the template moves from the most left hand side to the most right hand side, the percentage of the area of the template covered by gray pixels is recorded and will represent the y-axis of a curve and the position of the template will be represented as the x-axis of the curve. As a result, when the template scans through the counting zone of the image, a curve is plotted where x- axis shows the position of

the template and y-axis shows the percentage of the template areas covered by the gray pixels.

Figure 7: An example of foreground curve Analysis. The rectangle template scans the counting zone from left to right.

Obviously, in Figure 7 when an object is in the counting zone, a curve that rises and then drops will be plotted. This means that a loop will be plotted and the object in the image is represented by the loop. We observed that the shape of the curve will not change a lot if some gray pixels that represent the object in the segmented image are missed unless almost all gray pixels are missed.

3.1.2 Implementation

For example, in Figures 8 and 9, when the template is at position 295 where the object detected, it can be seen that a certain percentage of the area of the template is covered by the gray pixels of the object.

Figure 8: One person passing through counting zone.

Figure 9: Image of one person pass through counting zone after processing.

Figure 10: Plot of gray pixels percentage from a person walking in counting zone.

In Figure 10, it can be seen that when $x=295$, the y-axis will mark at a certain value indicating the percentage of the area of the template which is covered by black pixels. As a result, by counting the number of loops and analyzing the width of the loop, it can teach the computer to recognize and count the target object in the image. In the author's project, the width of the loop where y-axis = 0.1 was calculated. The width of the loop of the curve at position y-axis = 0.1 is $387-295 = 92$. In Figure 11, it can be seen that four people are walking closely together in a shoulder-to-shoulder parallel approach. After processing the image in Figure 12 and plotting the curve in Figure 13, it can be seen that there is one loop with three wave crests rather than three loops. But the computer can recognize that there are three people rather than one

person. The priority here was to set a suitable threshold value for the width to detect and recognize the number of people represented by the loop. After processing more sample images, we see where the width of the loop representing a typical obese person is about 80-90 based on a threshold value where the width of one person is set at 70 in the system. The number of people are counted based on the width of the loop divided by the width of one person which then taken as the integer portion. For example, for Fig.3.2.3, the width of the loop is 92, the computer will calculate it as $92/70 = 1.32$. If you take the integer portion for 1.32, you will get 1 as the final result. Therefore, the computer will recognize that there is one person. The same principle can be applied to the case of four people walking shoulder-to-shoulder together as shown in Fig. 3.2.4. In Fig. 3.2.6, it can be seen that there are two loops totally, and the width of first loop is very large, that is 242. Now, the computation is done by $242/70 = 3.46$ rounding off the integer portion to 3. One must then do the same thing to the second loop, computing $81/70 = 1.16$ and then round off the integer portion to 1. Add the number of first loop and second loop together and the computer will successfully recognize four people represented by the plot.

Figure 11: Four persons standing side by side.

Figure 12: Image of four-people case after processing.

Figure 13: Plot of gray pixels percentage for four-people case.

3.2 HOG features

Detecting and counting people in the video of a crowded scene is a challenging problem as described in Section 1.2. A robust feature set should allow each person to be discriminated cleanly even in a cluttered background with poor lighting conditions. It has been shown that the locally normalized Histogram of Oriented Gradient (HOG) descriptor provides excellent performance in object detection over other existing feature sets.

The HOG descriptors have many advantages in person detection. They can extract the edge and gradient information, which is very characteristic of local shapes, and perform with a controllable degree of invariance to local geometric and

photometric transformations. Variants of HOG descriptors have been proposed such as Rectangular HOG (R-HOG), Circular HOG (C-HOG), Bar HOG, Centre-Surround HOG. The sketch maps of R-HOG, C-HOG and Centre-Surround HOG are shown in Fig.4.1. As R-HOG is the easiest to implement among all these descriptors, only R-HOG will be adopted and discussed in this thesis.

Figure 14: Variants of HOG descriptors. (a) A rectangular HOG (R-HOG) descriptor with 3×3 blocks of cells. (b) Circular HOG (C-HOG) descriptor with the central cell divided into angular sectors as in shape contexts. (c) A C-HOG descriptor with a single central cell [36].

Different block normalization schemes can be applied. Four normalization schemes were evaluated in [35].

- L2 – norm : $\mathbf{v} \leftarrow \frac{\mathbf{v}}{\sqrt{\|\mathbf{v}\|_2^2 + \varepsilon^2}}$
- L2 – Hys : L2-norm followed by clipping (limiting the maximum values of $\mathbf{v}$ to 0.2) and renormalizing
- L1 – norm : $\mathbf{v} \leftarrow \frac{\mathbf{v}}{\|\mathbf{v}\|_1 + \varepsilon}$
- L1 – sqrt : $\mathbf{v} \leftarrow \sqrt{\frac{\mathbf{v}}{\|\mathbf{v}\|_1 + \varepsilon}}$

Where $\mathbf{v}$ is the normalized descriptor vector, $\|\mathbf{v}\|_k$ is its k -norm for $k = 1, 2$, and ε is a small normalization to avoid division by zero. It was shown in [35] that L2 – Hys, L2 – norm and L1 – sqrt normalization achieved equally good performance while

L1 – norm reduced performance by 5% compared with the other three. Each 32×32 pixel sample is divided into 4 cells with the size of 16×16 pixels. For each cell in each block, a histogram of nine orientation bins in 0- 180 is calculated and normalized within this block to represent the local features. The final descriptor is then a vector with the dimension of 36.

3.3 Bag of Words

In computer vision, the bag-of-words model can be applied to image classification by treating image features as words. In document classification, a bag of words is a sparse vector of occurrence when counting words: that is, a sparse histogram over the vocabulary. To represent an image using Bag of Words model, an image can be treated as a document. Similarly, “words” in images need to be defined too. To achieve this, it usually includes the following three steps: Feature detection, feature description and codebook generation. A definition of the Bag of Words model can be the “histogram representation based on independent features.” Content-based image indexing and retrieval appear to be early adopter of this image representation technique. After feature detection, each image is abstracted by several local patches. Feature representation methods deal with how to represent the patches as numerical vectors. These vectors are called feature descriptors. A good descriptor should have the ability to handle intensity, rotation, scale and affine variations to some extent.

The initial step for the Bag of Words model is to convert vector represented patches to “code words,” which also produces a “codebook.” A codeword can be considered as a representative of several similar patches. One simple method is performing k-means clustering over all the vectors. Code words are then defined as the

centers of the learned clusters. The number of clusters is the codebook size. Computer vision researchers have developed several learning to leverage methods for the Bag of Words model relating to image related tasks, such as object categorization. These methods can roughly be divided into two categories, generative and discriminative models. For multiple label categorization problems, the confusion matrix can be used as an evaluation metric.

3.4 K-means clustering

K-means is one of the simplest unsupervised learning algorithms used to solve the well-known clustering problem. K-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster. Given an initial set of k means $m_1^{(1)}, m_2^{(1)}, m_3^{(1)}, \dots, m_k^{(1)}$; hence, the algorithm proceeds by alternating between two steps:

(1) Assignment step: Assign each observation to the cluster whose mean yields the least within-cluster sum of squares. Since the sum of squares is the squared Euclidean distance, this is intuitively the “nearest” mean.

$$S_i^{(t)} = \left\{ x_p : \left\| x_p - m_i^{(t)} \right\|^2 \leq \left\| x_p - m_j^{(t)} \right\|^2 \forall 1 \leq j \leq k \right\} \quad (3.4.1)$$

where each x_p is assigned to exactly one $S_i^{(t)}$, even if it could be assigned to two or more of them.

(2) Update step: Calculate the new means to be the centroids of the observations in the new clusters.

$$m_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i^{(t)}} x_j \quad (3.4.2)$$

Since the arithmetic mean is a least-squares estimator, this also minimizes the

within-cluster sum of squares objective.

The algorithm has converged when the assignments no longer change. Since both steps optimize the within-cluster sum of squares objective, and there only exists a finite number of such partitioning, the algorithm must converge to a local optimum. There is no guarantee that the global optimum is found using this algorithm. The algorithm is often presented as assigning objects to the nearest cluster by distance. This is slightly inaccurate: The algorithm aims at minimizing the within-cluster sum of squares objective, and thus, assigns by “least sum of squares.” Using a different distance function other than Euclidean distance may stop the algorithm from converging. It is correct that the smallest Euclidean distance yields the smallest squared Euclidean distance and, thus, also yields the smallest sum of squares. The K-means clustering is used for features in bag of words. As shown in the bag of words in Figure 3, each feature is taken as a word into a bag. After that, K-means clustering is applied in the bag of words.

3.5 Support Vector Machines (SVM)

In machine learning, support vector machines (SVM) are supervised learning models with associated learning algorithms that analyze data and recognize patterns. SVMs are used for classification and regression analysis. The basic SVM takes a set of input data and predicts, for each given input, which of two possible classes forms the output thereby making a non-probabilistic binary linear classifier. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other. An

SVM model is a representation of examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.

The Linear SVM could be defined by the following equations. Given some training data, D , a set of n points of the form

$$D = \{(x_i, y_i) | x_i \in R^P, y_i \in \{-1, 1\}\}_{i=1}^n \quad (3.5.1)$$

where y_i is either 1 or -1, indicating the class to which point x_i belongs. Each x_i is a p -dimensional real vector. We want to find the maximum-margin hyper plane that divides the points having $y_i=1$ from those having $y_i = -1$. Any hyper plane can be written as the set of points x satisfying

$$w \cdot x - b = 0 \quad (3.5.2)$$

where $\cdot$ denotes the dot product and w the normal vector to the hyper plane. The parameter $\frac{b}{\|w\|}$ determines the offset of the hyper plane from the origin along the normal vector w .

If the training data are linearly separable, we can select two hyper planes in a way that they separate the data and there are no points between them, and then try to maximize their distance. The region bounded by them is called “the margin”. These hyperplanes can be described by the equations

$$w \cdot x - b = 1 \quad (3.5.3)$$

and

$$w \cdot x - b = -1 \quad (3.5.4)$$

By using geometry as in Figure 15, we find the distance between these two hyper

Figure 15: Maximum margin hyper plane and margins for an SVM trained with samples from two class.

planes as $\frac{2}{\|w\|}$, so we want to minimize $\|w\|$. As we also have to prevent data points from falling into the margin, we add the following constraint: For each i either

$$w \cdot x_i - b \geq 1 \text{ for } x_i \text{ of the first class} \quad (3.5.5)$$

or

$$w \cdot x_i - b \leq -1 \text{ for } x_i \text{ of the second class} \quad (3.5.6)$$

can be rewritten as :

$$y_i(w \cdot x_i - b) \geq 1 \text{ for all } 1 \leq i \leq n \quad (3.5.7)$$

3.6 Training Patches Specification

The most reliable feature for head-shoulder detection in surveillance scenes is its omega-like shape shown in Figure 16. HOG (Histograms of Oriented Gradients) features have been proven to be good shape descriptors [35] for human detection. We evaluated local feature descriptors for head-shoulder detection in the experiment section, and determined that HOG gives the best results. As illustrated in Figures 16 and 17, it is easy to see that typical head-shoulder samples have a similar shape, which is a curve-like omega. Figure 16 shows the average edge maps of head-

shoulder areas in training sets, and we found an omega-like curve. Typical head-shoulder samples can be seen in Figure 17. They vary in ethnicities, view angles, appearances and scenes. For negative samples, we also selected from the datasets we created. For diversity, we added some human-body images without head-shoulders to the negative set, which are shown in Figure 19.

Figure 16: Average edge maps (like omega) of head-shoulder samples in the training set.

Figure 17: Typical head-shoulder samples with edge map.

We created a head-shoulder dataset, which includes 9,100 positive samples with the size of 70×150 and 25,000 negative samples. Due to different videos with different angles and different horizons, it is necessary to resize the patches in a uniform standard. Figure 18 shows some samples of a positive training set and Figure 19 shows some samples of a negative training set.

Figure 18: Samples of positive training set.

Figure 19: Samples of negative training set.

To detect the omega curve successfully, it is essential to enlarge the negative training set as large as possible, and thereby zoom in on arm, leg, haversack, vehicle wheel, and so on. Using statistical analysis, we respectively deliberated the data range based on the width of a head, the length of a head and the width of shoulder. As illustrated in Figure 20 for width of a head, Figure 21 for length of head and Figure 22 for the width of a shoulder, we could find each range correspondingly for testing patches.

Figure 20: The first quartile is 26.75. The third quartile is 30. The mean is 28.2. The median is 28.5.

Figure 21: The first quartile is 34.5. The third quartile is 40.
The mean is 37.05. The median is 36.5.

Figure 22: The first quartile is 51. The third quartile is 60.25.
The mean is 55.5. The median is 53.5.

We employed different videos as training patches to source and test the training patches individually. The results are shown in Figures 23 and 24.

Figure 23: Result for one person.

Figure 24: Result for 4 persons.

CHAPTER 4

TEMPORAL ANALYSIS FOR PEDESTRIAN COUNTING

We can see that the head-shoulder detection achieved a fairly acceptable performance. After that classification, we applied a pedestrian counting method to count how many people passed through the counting zone. As mentioned before in Figure 3, the pedestrian counting includes two parts: recording procedure and analyzing procedure.

4.1 Counting Rules

As people pass through the counting zone in small groups, it is critical to be able to robustly count the number of individuals at any given time. In the head-shoulder detection part, if one person passes through the counting zone, there will be a bounding box placed over the person. At same time, the system will record the frame number and left bottom position of the bounding box. If more than one person appears in the counting zone, the system will record the left bottom position of the bounding box and total number of people in the counting zone. Table 3 shows the recording results for frames 7706 to 7721. To record this data, it always follows two rules: First, when there is a continuous run of ten frames with no one in the counting zone, the system will stop the recording procedure and begin to analyze how many people passed through. Second, when it finishes analyzing the procedure, it will clear the matrix and wait for recording.

Table 2: Frame 7706 to 7721 Position Matrix.

Frame	7706	7707	7708	7709	7710	7711	7712	7713	7714	7715	7716	7717	7718	7719	7720	7721
Number	1	1	2	2	1	2	2	2	2	2	2	2	2	2	1	2
Position	143	192	143	143	143	144	144	144	145	144	144	144	144	144	184	144

Position	0	0	192	192	0	190	190	190	195	195	195	190	180	195	0	190
position	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
position	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

After recording the position, we start to process the data and estimate how many people pass through. We adopt the method tracking position matrix frame by frame. Based on a specific counting zone, the horizontal coordinate of the left bottom of bounding box represents the position of people, because the vertical coordinate never changes. As long as the near position occurs more than one time from a review of the continuous six frames, the system will view it as “continuous”. Moreover, in terms of the width of a patch, if the absolute difference of two positions from “continuous” frame is less than a threshold, it is reasonable to view that as one person. Here, we define the threshold as two-thirds in the width of the patch. In this way, we can easily find a tracking line in the position matrix. In a dynamic and unpredictable video, some people walk fast and some walk slow. But even if a person rides a motorcycle to pass through the counting zone, he cannot be too fast to appear in less than nine frames in a video. Therefore, we can define a threshold to clean some meaningless data, such as some positions only occur two times, and there is always some error detection. However, there are many people who walk very slowly in a video, so that they could appear in more than 60 frames. In this situation, it becomes difficult to separate two people who walk closely leaving space in the front and to the rear between each other.

4.2 Implementation

Figure 25 shows one example of position matrix with 2,000 frames. It can be seen that there is one person passing through or around frame 790, and there are four people passing through or around frame 1620. In Table 4, it can be seen that there is no

breakpoint from frame 763 to frame 781.

Figure 25: Position matrix plot for the first 2000 frames.

Table 3: Frame 763 to 781 Position Matrix.

Frame	763	764	765	766	767	768	769	770	771	772	773	774	775	776	777	778	779	780	781
Number	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Position	142	142	140	140	139	139	137	137	137	137	136	135	135	135	135	134	134	132	132
Position	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
position	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
position	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Furthermore, the maximum of absolute differences between two frames is 2, which is far away from the threshold, with a two-thirds width of patch. Figure 26 shows the plot from frame 740 to 810, and there is only one person passing by.

Figure 26: Position matrix plot for frame 740 to 810.

Table 5 shows another example from frames 1631 to 1649. This table looks a little more complex than Table 4. As mentioned in Section 4.1, which discussed counting rules, this searching should start from 162 with the first position in frame 1631. We keep searching until frame 1649, and all the positions for this first person have been dyed blue. All the positions in blue should be clear and the number in the second row of Table 5 should be reduced by one.

Table 4: Frame1631 to 1649 Position Matrix.

Frame	1631–1649																	
Number	3	3	4	4	4	4	4	4	4	3	3	4	4	3	4	4	4	4
Position	162	162	101	101	99	99	97	96	109	109	153	107	106	151	91	91	91	90
Position	212	212	151	151	164	149	147	146	159	159	203	157	156	201	141	141	141	140
Position	255	255	201	201	214	199	197	196	209	239	253	207	206	251	191	191	191	190
Position	0	0	253	252	264	249	247	246	259	0	0	257	256	0	241	241	241	240

The second searching step starts from 212, and the second position is in frame 1631. However, we find that there is a breakpoint in frame 1640. There is no connection point between 209 and 203, and there is a position 239 with no connection to any other position. In our counting rules, as long as the period between two positions is no more than six frames, we could view it as “continuous.” As for position 239, we call this a “dead position,” and it must, therefore, be clean by this rule searching and searching. In Figure 27, we could find that there are four people during these frames.

Figure 27: Position matrix plot for frame 1540 to 1700.

CHAPTER 5

EXPERIMENTAL RESULTS

5.1 Experimental Dataset

Because there is no open large dataset for head-shoulder detection, we created two. For the first dataset has three videos in Engineering Building West. There are 5,551 positive and 17,601 negative samples with a size of 75*150 as the training data. We retested the training data and obtained a 98.64% accuracy as shown in Table 5. We used the codebook and model from video EBW1 to test video EBW2 and EBW3 and obtained the results shown in Table 6. The estimated accuracy rate is around 89.7% over 33,315 patches.

Table 5: Test training data for dataset on EBW.

	Total Patch	Correct Patch	Incorrect Patch	Accuracy
Quantity	23,152	22,838	314	98.64%

Table 6: Test video EBW2 and EBW3.

	Detection Number	More Counting	Less Counting	Real Number	Accuracy	Error Rate
EBW1	65	1	4	68	95.59%	7.35%
EBW2	54	0	5	59	91.53%	8.47%
EBW3	60	0	4	64	93.75%	6.25%

Figure 28: Some samples look too vague to distinguish if there is a head or not.

Figure 29: EBW1 Original frame.

Figure 30: EBW1 frame after background subtraction

In video EBW1, the result after background subtraction showed before in

Figure 7 and Figure 9. Figure 29 and Figure 30 show the large changes in illumination and its impact on person counting. Therefore, if the video often changes the illumination, the background subtraction algorithm will not work well. There is a frame with three people passing through, as shown in Figure 31. The foreground curve analysis will produce results as shown in Figure 32. After the system captures the position range, the following algorithm will detect the specific position and will allow placement of the bounding box over the frame.

Figure 31: There is a frame with three people passing through.

Figure 32: Curve analysis plot for the frame with 3 people passing through.

In Figure 33, there are three pictures shown, including the frame after background subtraction, the curve analysis plot and the result frame. These pictures are obtained from video EBW2, which is a test video with the same scene and different resolutions compared with video EBW1. In Figure 34, these three pictures

were captured from video EBW3.

Figure 33: The results for EBW2

Figure 34: The results for EBW3

The second dataset includes two video in a farmer market. We use the first half video for training data and tested the other half and another video. There are 7228 positive patches and 13253 negative patches for training data. We retested the training data and obtained 98.64% accuracy as shown in Table 8. A codebook and model was used to test another video with similar scenes and we obtained the results shown in

Table 9.

Table 7: Test training data for dataset 2.

	Total Patch	Correct Patch	Incorrect Patch	Accuracy
Quantity	20,481	19737	744	96.37%

Table 8: Table 7: Test another video with similar scene.

	Detection Number	More Counting	Less Counting	Real Number	Accuracy	Error Rate
Test Video	76	2	11	85	89.41%	15.29%

In Figure 35, the frame after background subtraction and result frame are shown. In Figure 36, the curve analysis plot is shown. It is obvious that the background subtraction is not as good as the video from EBW. The reason is the distance from camera to counting zone is further than that from the EBW video. Therefore, the speed of people moving is much slower than before, which led to the background modeling appearing a little faster than before. So the plot for curve analysis also appears as somewhat inexact. However, the function for speedup is still working.

Figure 35: The result for test video on vegetable market.

Figure 36: The curve analysis plot for test video.

On the detection part, we did a specific study on this video. There are a total of 41,253 patches with a potential accuracy of more than 96%. Details are shown in Table 9.

Table 9: Patches TP, TN, FP, FN.

	True	False
Positive	4857	1420
Negative	132	34844

5.2 Discussion

In test video EBW2, one person purposely puts on his hat and the system did not detect him. In the test video from the farmer market, we found that as long as there was a person with a hat or helmet, it was very difficult to detect the person. Due to a fact that there was no one with hat or respirator in training data on both datasets, we concluded that besides the omega curve, which is what we wanted, there were several complex curves extracted. In test video EBW3, when one actor walked sideways, the system could not detect that person.

In the farmer market test video, there were several images, which could not be detected successfully as shown in Figure 37. Because this video was more realistic, it

was also more informative, and we found several failed detections. The system sometimes mistook some bags, frontal face of motorcycle and headlamp for head-shoulder combinations. The most important reason for these failed detections is due to a lack of negative training data. Some patches with head-shoulder combinations should have been classified as a positive set, but the system mistook them for negative patches. Take the right top patch in Figure 37 as an example: The background of this patch is an electro-tricycle. Due to the complexity of background, there are several edges with various kinds of angle. At this point, HOG features could not get the most important gradients, and some meaningless gradients held a large proportion. Moreover, the headlamp was truly easy to be mistaken as a head and shoulder. The reason is that the shape of a headlamp is close to a person's head and the edge of a car is close to person's shoulder. Due to the algorithm bag of words used in the system, the sequence in the "dictionary" has been disorganized. In this way, the omega curve could be decomposed into three parts: two curves bent to look more like horizontal lines and one curve extended into a shape more like a circle.

Figure 37: Wrongly detected patches in test video

CHAPTER 6

CONCLUSIONS AND FUTURE DIRECTION

A head-shoulder method using SVM classifier with HOG feature and bag of words was developed in this work. This system performed a robust result and obtained more than 90% accuracy.

However, further algorithm refinement is needed to achieve better performance. Most of the people counting systems used person detection from one camera, which we mentioned in Section 1.3 of Chapter 1. In [46], Danny B. Yang proposed an alternative approach that directly estimates the number of people. In their system, groups of image sensors segment foreground objects from the background, aggregate the resulting silhouettes over a network, and compute a planar projection of the scene's visual hall. This method is new and promising. There are some papers [56] roughly counting with the area of people projected on an image captured by a zenithal video camera. In our future work, we shall explore these types of ideas to further improve the system performance.

REFERENCES

- [1] Cope A, Doxford D, Probert C. Monitoring visitors to UK countryside resources The approaches of land and recreation resource management organisations to visitor monitoring[J]. Land Use Policy, 2000, 17(1): 59-66.
- [2] S.J. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler, “Tracking Groups of People,” Computer Vision and Image Understanding, vol. 80, no. 1, pp. 42-56, Oct. 2000.
- [3] Wren C R, Azarbayejani A, Darrell T, et al. Pfunder: Real-time tracking of the human body[J]. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 1997, 19(7): 780-785.
- [4] Gavrilu D M. Pedestrian detection from a moving vehicle[M]//Computer Vision—ECCV 2000. Springer Berlin Heidelberg, 2000: 37-49.
- [5] Spagnolo P, Orazio T D, Leo M, et al. Moving object segmentation by background subtraction and temporal analysis[J]. Image and Vision Computing, 2006, 24(5): 411-423.
- [6] Liu X, Tu P H, Rittscher J, et al. Detecting and counting people in surveillance applications[C]//Advanced Video and Signal Based Surveillance, 2005. AVSS 2005. IEEE Conference on. IEEE, 2005: 306-311.
- [7] R. Cucchiara, C. Granan, M. Piccardi, and A. Prati, “Detecting moving objects, ghosts, and shadows in video streams,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 25, no. 10, pp. 1337–1342, Oct. 2003.

- [8] N. M. Oliver, B. Rosario, and A. P. Pentland, "A Bayesian computer vision system for modeling human interactions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 8, pp. 831-843, Aug 2000.
- [9] Haritaoglu, I.; Harwood, D.; Davis, L. S. (2000). W4: real-time surveillance of people and their activities, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 8, Aug., pp.809–830.
- [10] C. Stauffer and W.E.L. Grimson, "Learning Patterns of Activity Using Real-Time Tracking," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 747-757, Aug. 2000.
- [11] Yu S, Chen X, Sun W, et al. A robust method for detecting and counting people[C]//Audio, Language and Image Processing, 2008. ICALIP 2008. International Conference on. IEEE, 2008: 1545-1549.
- [12] Paragios N, Ramesh V. A MRF-based approach for real-time subway monitoring[C]//Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on. IEEE, 2001, 1: I-1034-I-1040 vol. 1.
- [13] L. Wang, T. Tan, H. Ning and W. Hu, "Silhouette Analysis-Based Gait Recognition for Human Identification", *IEEE Trans. Pattern Analysis and Machine Intelligence*, pp.1505-1518, December 2003.
- [14] Hou Y L, Pang G K H. People counting and human detection in a challenging situation[J]. *Systems, Man and Cybernetics, Part A: Systems and Humans*, *IEEE Transactions on*, 2011, 41(1): 24-33.

- [15] Magee D R. Tracking multiple vehicles using foreground, background and motion models [J]. *Image and vision Computing*, 2004, 22(2): 143-155.
- [16] Yang D B, González-Baños H H, Guibas L J. Counting people in crowds with a real-time network of simple image sensors[C]//*Computer Vision*, 2003. Proceedings. Ninth IEEE International Conference on. IEEE, 2003: 122-129.
- [17] Giebel, Jan, Darin M. Gavrilu, and Christoph Schnörr. "A bayesian framework for multi-cue 3d object tracking." *Computer Vision-ECCV 2004*. Springer Berlin Heidelberg, 2004. 241-252.
- [18] Regazzoni C S, Tesei A. Distributed data fusion for real-time crowding estimation[J]. *Signal Processing*, 1996, 53(1): 47-63.
- [19] Nakajima C, Pontil M, Heisele B, et al. People recognition in image sequences by supervised learning[R]. MASSACHUSETTS INST OF TECH CAMBRIDGE MA CENTER FOR BIOLOGICAL AND COMPUTATIONAL LEARNING, 2000.
- [20] R. Cucchiara, C. Granan, M. Piccardi, and A. Prati, "Detecting moving objects, ghosts, and shadows in video streams," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 10, pp. 1337–1342, Oct. 2003.
- [21] Viola P, Jones M. Rapid object detection using a boosted cascade of simple features[C]//*Computer Vision and Pattern Recognition*, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on. IEEE, 2001, 1: I-511-I-518 vol. 1.

- [22] Lowe D G. Distinctive image features from scale-invariant keypoints[J]. International journal of computer vision, 2004, 60(2): 91-110.
- [23] P-M Jodoin, M Mignotte, J Konrad, Statistical background subtraction using spatial cues. IEEE Trans. Circuits Syst. Video Technol. 17(12), 1758–1763 (2007)
- [24] R Cucchiara, C Grana, M Piccardi, A Prati, S Sirotti, Improving shadow suppression in moving object detection with HSV color information, in Proceedings of the IEEE Intelligent Transportation Systems Conference (Oakland, CA), 25-29 Aug 2001, pp. 334–339
- [25] Ge W, Collins R T. Marked point processes for crowd counting[C]//Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009: 2913-2920.
- [26] D. Comaniciu, V. Ramesh, and P. Meer, “Real-time tracking of non-rigid objects using mean shift,” in Proc. CVPR, pp. 142-151, 2000.
- [27] J Stauder, R Mech, J Ostermann, Detection of moving cast shadows for object segmentation. IEEE Trans. Multimed. 1(1), 65–76 (1999)
- [28] Piccardi M. Background subtraction techniques: a review[C]//Systems, man and cybernetics, 2004 IEEE international conference on. IEEE, 2004, 4: 3099-3104.
- [29] S Nadimi, B Bhanu, Physical models for moving shadow and object detection in video. IEEE Trans. Pattern Anal. Mach. Intell. 26(8), 1079–1087 (2004)

- [30] N. Ohta, "A Statistical Approach to Background Suppression for Surveillance Systems," Proc. IEEE Int'l Conf. Computer Vision, pp. 481-486, 2001.
- [31] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, 2005, pp. 886-893 vol. 1.
- [32] Stauffer C, Grimson W E L. Adaptive background mixture models for real-time tracking[C]//Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on. IEEE, 1999, 2.
- [33] Antic B, Letic D, Culibrk D, et al. K-means based segmentation for real-time zenithal people counting[C]//Image Processing (ICIP), 2009 16th IEEE International Conference on. IEEE, 2009: 2565-2568.
- [34] Lim Eng Aik and Zarita Zainuddin, "Real-Time People Counting System using Curve Analysis Method" in International Journal of Computer and Electrical Engineering, Vol. 1, No. 1, April 2009, 1793-8198
- [35] N. Dalal, "Finding people in images and videos," PhD, Institut National Polytechnique De Grenoble, 2006.
- [36] Bartolini F, Cappellini V, Mecocci A. Counting people getting in and out of a bus by real-time image-sequence processing[J]. Image and vision computing, 1994, 12(1): 36-41.
- [37] Rossi M, Bozzoli A. Tracking and counting moving people[C]//Image Processing, 1994. Proceedings. ICIP-94., IEEE International Conference.

- IEEE, 1994, 3: 212-216.
- [38] Marana A N, da Fontoura Costa L, Lotufo R A, et al. Estimating crowd density with Minkowski fractal dimension[C]//Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on. IEEE, 1999, 6: 3521-3524.
 - [39] Mikolajczyk K, Schmid C, Zisserman A. Human detection based on a probabilistic assembly of robust part detectors[M]//Computer Vision-ECCV 2004. Springer Berlin Heidelberg, 2004: 69-82.
 - [40] E. Osuna, R. Freund, and F. Girosit, "Training support vector machines: an application to face detection," in Computer Vision and Pattern Recognition, 1997. Proceedings. 1997 IEEE Computer Society Conference on, 1997, pp. 130-136.
 - [41] JD Tian, J Sun, YD Tang, Tricolor attenuation model for shadow detection. IEEE Trans. Image Process. 18(10), 2355–2363 (2009)
 - [42] VJD Tsai, A comparative study on shadow compensation of color aerial images in invariant color models. IEEE Trans. Geosci. Remote Sens. 44(6), 1661–1671 (2006)
 - [43] YH Lu, HJ Xin, J Kong, BB Li, Y Wang, Shadow removal based on shadow direction and shadow attributes, in Proceedings of the IEEE International Conference on Computational Intelligence for Modeling Control and Automation (Sydney, NSW), Nov. 28 2006-Dec. 1, 2006, pp. 37–41.
 - [44] X Pan, Moving shadow detection based on color information and edge

- features. J. Zhejiang Univ. (Engineering Science). 38(4), 389–391 (2004)
- [45] Y. Zhong and A. K. Jain, "Object localization using color, texture and shape," presented at the Proceedings of the First International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition, 1997.
- [46] Yang D B, Gonz ález-Bañ os H H, Guibas L J. Counting people in crowds with a real-time network of simple image sensors[C]//Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on. IEEE, 2003: 122-129.
- [47] R. Cucchiara, C. Granan, M. Piccardi, and A. Prati, "Detecting moving objects, ghosts, and shadows in video streams," IEEE Trans. Pattern Anal. Mach. Intell., vol. 25, no. 10, pp. 1337–1342, Oct. 2003.
- [48] Horprasert T, Harwood D, Davis L S. A statistical approach for real-time robust background subtraction and shadow detection[C]//IEEE ICCV. 1999, 99: 1-19.
- [49] Elgammal A, Harwood D, Davis L. Non-parametric model for background subtraction[M]//Computer Vision—ECCV 2000. Springer Berlin Heidelberg, 2000: 751-767.
- [50] Ma R, Li L, Huang W, et al. On pixel count based crowd density estimation for visual surveillance[C]//Cybernetics and Intelligent Systems, 2004 IEEE Conference on. IEEE, 2004, 1: 170-173.

- [51] Serné H, Stehouwer C D A, ter Maaten J C, et al. Microvascular function relates to insulin sensitivity and blood pressure in normal subjects[J]. *Circulation*, 1999, 99(7): 896-902.
- [52] Stinebrickner-Kauffman T. Counting Matters: Prison Inmates, Population Bases, and one Person, One Vote[J]. *Va. J. Soc. Pol'y & L.*, 2003, 11: 229.
- [53] Cutler R, Davis L S. Robust real-time periodic motion detection, analysis, and applications[J]. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2000, 22(8): 781-796.
- [54] Dong Xu, Jianzhuang Liu, Xuelong Li, Zhengkai Liu, and Xiaoou Tang, "Insignificant Shadow Detection for Video Segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 8, pp. 1058-1064, 2005.
- [55] Cutler R, Davis L S. Robust real-time periodic motion detection, analysis, and applications[J]. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2000, 22(8): 781-796.
- [56] Chen T H, Chen T Y, Chen Z X. An intelligent people-flow counting method for passing through a gate[C]//*Robotics, Automation and Mechatronics*, 2006 IEEE Conference on. IEEE, 2006: 1-6.
- [57] Li M, Zhang Z, Huang K, et al. Estimating the number of people in crowded scenes by mid based foreground segmentation and head-shoulder detection[C]//*Pattern Recognition*, 2008. ICPR 2008. 19th International Conference on. IEEE, 2008: 1-4.

- [58] M. Hu. Visual pattern recognition by moment invariants. IRE Trans. Information Theory, vol. 8, no. 2, pp. 179-187, 1962.