Texture-Based Crowd Detection and Localisation

Stefano Ghidoni[1], Grzegorz Cielniak[2], and Emanuele Menegatti[1]

[1] Intelligent Autonomous Systems Laboratory (IAS-Lab)
Department of Information Engineering
The University of Padova
via Gradenigo, 6/B – 35131 Padova – Italy
{ghidoni,emg}@dei.unipd.it
[2] School of Computer Science
University of Lincoln
Brayford Pool, Lincoln, LN6 7TS, UK
gcielniak@lincoln.ac.uk

Abstract. This paper presents a crowd detection system based on texture analysis. The state-of-the-art techniques based on co-occurrence matrix have been revisited and a novel set of features proposed. These features provide a richer description of the co-occurrence matrix, and can be exploited to obtain stronger classification results, especially when smaller portions of the image are considered. This is extremely useful for crowd localisation: acquired images are divided into smaller regions in order to perform a classification on each one. A thorough evaluation of the proposed system on a real world data set is also presented: this validates the improvements in reliability of the crowd detection and localisation.

Keywords: Crowd detection, intelligent video surveillance.

1 Introduction

Crowd monitoring is a very important topic in the field of video surveillance. Crowds represent a potentially dangerous environment, due to the difficulty of controlling a high number of people. Crowds often set up during events like football matches and concerts, that take place inside large venues, under the control of a security staff. Such a large public is constantly monitored by security officers that look at the whole crowd from a distant point of view, and try to detect unusual or potentially dangerous situations. They coordinate a number of stewards that operate in the field in order to prevent dangerous situations and keep under control crowd density, that is crucial for safety considerations. Crowds can create and grow also unexpectedly, under several exceptional conditions; this second case is extremely dangerous, because of the lack of management of such large groups of people.

Over the years, technology has been used to help security officers: nowadays, several active cameras are installed in the venues and the control room of a football stadium is usually full of monitors and camera controls. This virtually enables security staff to analyse every part of the venue, inside and outside, so that the whole crowd can be observed. However, the huge amount of data provided by such surveillance systems would require several operators to be completely analysed, leading to very high costs.

S. Lee et al. (Eds.): Intelligent Autonomous Systems 12, AISC 193, pp. 725–736.
springerlink.com

On the other hand, if just one or two people are devoted to the task of analysing video streams provided by all cameras, they will likely miss several situations of interest.

To cope with this issue, a system capable of analysing multiple video streams and automatically detecting situations that require human intervention would be extremely effective. Such a system should be able to detect the crowd inside a venue, analyse its behaviour, and detect situations that could be dangerous: in that case, a security officer could be warned, in order to adopt proper countermeasures. In other words, such an automatic system is asked to select suspicious situations, and let security staff focus on them. The first step in developing a system capable of understanding dangerous situations inside large venues is detection and localisation of the crowded areas, together with crowd density estimation.

In this paper, we present a crowd monitoring system that is able to detect and localise crowded areas in the image. The core of our approach is based on a machine learning classifier which can categorise texture features extracted from image regions into a set of classes corresponding to different crowd density. We extend the feature set proposed by [1] to improve the classification performance in small image areas, demonstrate the ability of the method to localise the crowd and present a thorough evaluation of the system on a challenging data set acquired during a real crowded event. We first present state of the art in the area of crowd monitoring, then describe different components of our system, followed by experiments and finally briefly conclude and discuss possible future work.

2 Related Work

The first thorough study on crowd analysis dates back to the mid '90s: [2] discusses the importance of such topic in order to enhance public safety, and proposes a way for describing crowd at a high level that is inspired by gas dynamics laws. At a lower level, the paper deals with several computer vision techniques for detecting crowds.

Over the years, several approaches have been developed to detect crowd. For instance, some methods are based on motion analysis: in [3] a technique based on the observation of the motion of particles is described; such particles are initially evenly placed over the image domain, and then moved according to optical flow. In [4] a technique based on motion heat maps is presented, together with a set of indicators for measuring motion entropy and, finally, classifying motion as normal or abnormal. Finally, in [5] a method is presented, that is capable of detecting the precise contour of a crowd.

2.1 Texture Analysis

The main approach is based on the evaluation of GLDM (Gray Level Dependency Matrix): this is a method, dating back to the '70s, that aims at measuring texture content, and was originally developed for satellite image processing [6]. This first paper presenting GLDM discussed also a set of features for characterising it, and consequently measure image texture. The proposed feature set is quite large, but only few of the features were exploited when the co-occurrence matrix was employed for detecting crowds: for example in [1] four features called contrast, homogeneity, energy, and entropy were used as input to a neural network, developed for classifying crowd density. In [7] the same indicators were exploited, again for measuring crowd density.

Beside optical flow-based and GLDM-based methods, some works tackle the problem in other ways, like edge density [8,9], HOG descriptor [10], SIFT feature density [5]. In [11], a number of texture classification techniques is considered.

2.2 Crowd and Groups

A point that needs to be highlighted is the difference between a crowd and a group. Usually, people detectors (or pedestrian detectors) focus on detecting every person, that is supposed to be fully visible. However, people often walk in groups, that are composed of a number of people that are fully visible, and just side by side, or cause little occlusion [12,13,14]. A crowd is, on the other hand, a group in which people are so close that the occlusion level is sensibly high, and a few people (or even none) are fully visible [11]. This definition is coherent with computer vision techniques that are used in these two scenarios: when dealing with groups, a common people detector can be used to analyse the scene, while when dealing with crowds, such techniques fail and specific algorithms need to be created in order to understand the scene. Of course, intermediate scenarios also exist, like in [10].

3 Crowd Detection System

The developed system aims at detecting crowd, defined, as previously described, as a large group of people characterised by a high level of occlusion, that makes it impossible to see every single human body. The system has been developed starting from the GLDM-based texture analysis technique already proposed in the literature; however, deeper studies on the GLDM revealed that such matrix contains a great amount of information, that is only partially extracted by features commonly used in other systems. Therefore, we proposed a novel method for crowd analysis.

Our system is based on the GLDM approach since this offers several advantages over other methods. First, it does not require the crowd to generate sensible motion cues, as it is needed by systems based on motion observation; moreover, it runs faster than methods based on feature density: in this case, performance depends on the computational complexity for extracting each feature; in general, however, this time is not negligible and performance tends to drop when high concentrations of features are found in the images, that is, just when crowd is present in the scene. This makes features-based approaches not best-suited for real-time applications. On the other hand, GLDM-based approaches suffer from the complexity of extracting salient features once such histogram is available: this one of the main topic addressed in this paper.

3.1 GLDM Indicators

The GLDM can be seen as a two-dimensional histogram, that is created by setting up a grid of 256×256 locations (in the common case of 8-bit image depth), one for each greyscale value. The image for which the GLDM should be evaluated is then analysed by considering pixel couples, that are usually two pixels that are adjacent in a row, column or along a diagonal. For each couple, the two grey-scale values are used as the coordinates of the GLDM, and the corresponding bin value is increased of one unity. A GLDM depends on two parameters: d – the distance between the pair of pixels and the orientation (horizontal, vertical or diagonal). Once the whole image has been scanned and each pixel couple considered, the GLDM represents how pixels change: if only smooth variations can be found in the image, the GLDM will be concentrated towards the diagonal, while abrupt changes will lead to peaks that have a certain distance from the diagonal.

Since abrupt pixel changes increase the bins that are far from the diagonal, several indicators exploited so far are ways of performing a weighted sum of the bins. The weights depends on the distance to the diagonal; this is the case, for example, of contrast and homogeneity. Other indicators measure how even the distribution among the bins is, as it is the case of energy and entropy. These four features were exploited in [1].

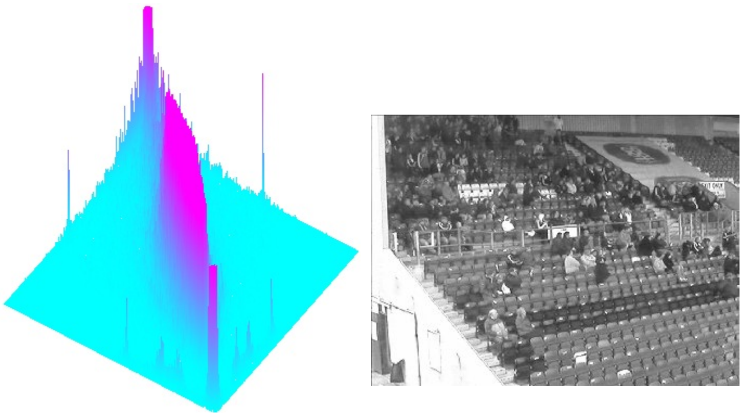

Fig. 1. An example of GLDM and the frame it is evaluated from

In Figure 1 a three-dimensional plot of a GLDM is shown: it presents several peaks of different size and shape: it is clear that such histogram is only partially measured when discussed indicators are evaluated. For example, a high contrast indicates that a sensible amount of the histogram is localised at some distance to the diagonal, but it is then impossible to know if this component is concentrated in some small area, or it develops over a wide region. For example, consider Figure 2 where two frames with substantially different texture contents are represented, together with the corresponding GLDMs that appear different as well. However, by considering the four mentioned indicators, it is possible to observe that just one of them (i.e. energy) undergoes significant change in value when comparing the two histograms.

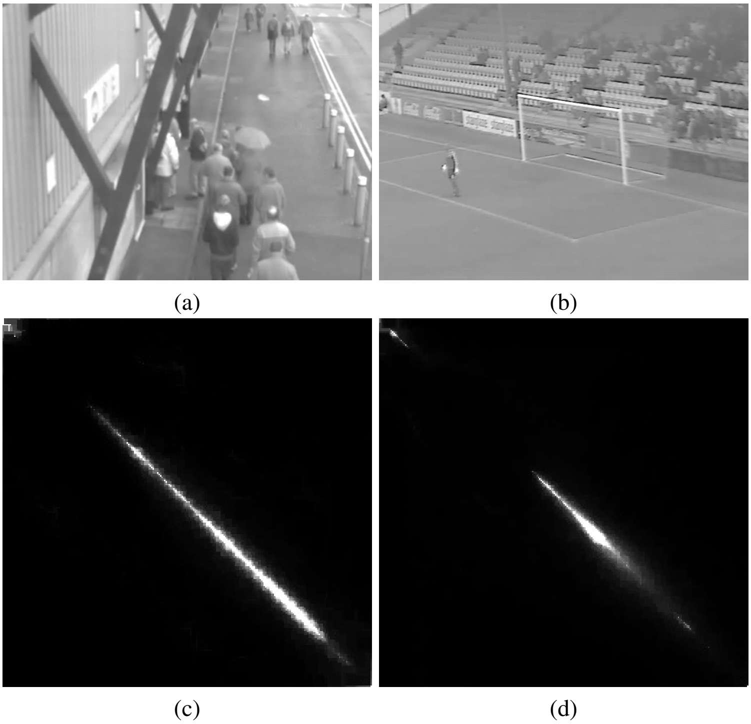

Fig. 2. Example of two different frames (a) and (b) characterised by different GLDMs (c) and (d). However, values of indicators commonly used in the literature for the two images are similar. GLDMs are drawn in two dimensions: the height of each bin is coded by the grey-level of the corresponding pixel.

3.2 GLDM Shape Analysis

To obtain a better characterisation of GLDMs, a set of new features has been developed, whose aim is to describe in more detail the shape of the histogram. From a detailed shape description it is then possible to obtain much more data than that provided by commonly used indicators.

To analyse the shape of a GLDM, a technique based on its contour lines at several heights has been developed: such lines are evaluated, and some parameters are exploited to measure the properties of their shapes; finally, parameters referring to a single line are compared to those evaluated for contour lines at other heights, and from the observed pattern, information is gathered about the shape of the GLDM.

3.2.1 Single Contour Line Description

A single contour line usually appears as a set of curves, one of which is clearly larger than the others, and is therefore called the principal component. This is the only part to be analysed, as smaller components are usually similar for every image. Each contour line is evaluated by intersecting the GLDM with a horizontal plane at a given height: by choosing several planes at different heights it is possible to provide a detailed description of the two-dimensional histogram.

Once the principal component is found, the least squares fitting algorithm is used to find the best fitting ellipse; the axes lengths are then evaluated. The ellipsis was chosen because it is the best suited shape for representing the typical contour line provided by GLDMs.

In all cases of interest, the ellipse has its major axis placed along the diagonal of the GLDM; its length tends to be maximum when the level curve is at a low height, unless the original image is not extremely uniform.

3.2.2 Multilayer Shape Analysis

The GLDM is analysed at several heights; ten height values were enough to precisely characterise the shape in all cases we observed. Values used for the actual heights depend on the size of the original image, as larger image produce higher histograms, due to the higher number of pixels; however, this problem can be solved by evaluating the normalised GLDM.

Once the result of the ellipse fitting is available for every contour line, some indicators are derived from the comparison of all of them. Such indicators are:

- **Evenness** depends on the ellipse minor axis at different heights. If w_i represents the minor axis at the i-th layer, N the total number of layers, and $\overline{\Delta w}$ is the average of the minor axis difference between ellipses belonging to consecutive layers, evenness is then defined to be:

$$\mathrm{EV} = \sqrt{\frac{1}{N} \frac{\sum_{i=1}^{N-1} \left[(w_i - w_{i+1}) - \overline{\Delta w} \right]^2}{i}} . \tag{1}$$

  Evenness is a measure of how evenly the ellipse minor axis shorten its length as long as the height at which the contour line is found is increased; the division by i is necessary to cope with the fact that the difference between subsequent height values does not increase linearly.
- **Minor axis spread** measures how steep the GLDM is, and is evaluated as the difference between the maximum and minimum lengths of the minor axis over all contour lines:

$$\mathrm{MAS} = w_{max} - w_{min} . \tag{2}$$

- **Minimum width** of the minor axis over all contour lines, which provides information about the smoothness of the upper part of the GLDM.
- **Total volume** included under the analysed level curves measures how spread the GLDM is outside the main component. If h_i and w_i are the major and minor axis of the ellipse found at the i-th height h_i, it is evaluated as:

$$\mathrm{TV} = w_1 h_1 l_1 + \sum_{i=2}^{N} w_i h_i \left(l_i - l_{i-1} \right) . \tag{3}$$

- **Width-height ratio** measures the mean of the width-height ratio of all ellipses, describing how spread over dark and light regions the texture is. It is evaluated as:

$$\mathrm{WHR} = \frac{1}{N}\sum_{i=1}^{N}\frac{w_i}{h_i}\,. \tag{4}$$

- **Volume of the peak** is the volume of the portion of the GLDM that is above the highest level curve, and represents another descriptor of the GLDM shape in its peak. If L is the highest level at which the GLDM is analysed, and there are M bins of height $H_i \geq L$, the peak volume is then evaluated as:

$$\mathrm{PV} = \sum_{i=1}^{M}(H_i - L)\,. \tag{5}$$

  This is a way of taking into consideration the part of the histogram that is neglected by the analysis of the contour lines.
- **Maximum-minimum area ratio** is the ratio between the area of the largest ellipse, and that of the smallest one, and describes the gradient of the GLDM; this can appear similar to the minor axis spread, but the major axis of each ellipse is also involved in this case. Since ellipses found at lower levels are larger than the ones found at higher levels, this ratio is evaluated as:

$$\mathrm{MMAR} = \frac{\mathrm{Area}_1}{\mathrm{Area}_N}\,. \tag{6}$$

- **Minimum area** is the area of the smallest ellipse, another descriptor of the GLDM peak.
- **Blank GLDM locations** is the number of empty bins in the GLDM. This is not directly related to the shape of the GLDM; however, this indicator is useful because it describes the histogram at the lowest level, as if a level curve at height 1 would be considered. The number of blank locations has been chosen instead of such level curve, because this second solution would have neglected a high number of blank locations that are inside a region composed of bins that are low, but anyway greater than zero. This situation is often the case in the region where the GLDM is very low.

These features represent a novel way for analysing GLDMs, that takes into consideration a high number of geometrical characteristics. It is then possible to use them as input for a classifier, together with previously used features, in order to obtain more accurate results.

3.3 Classification

Features described in the previous section form input data for a machine learning classifier. We use AdaBoost [15] to categorise these features into classes corresponding to different density of the crowd. AdaBoost combines so called "weak classifiers" into one final strong classifier that performs better than any of the weak classifiers alone (see [15] for details of the training algorithm). We use the Real AdaBoost variant of the algorithm [16] that provides a lower error rate by allowing weak classifiers to vote by their individual degree of certainty instead of making simple binary decision. We use binary decision trees [17] as weak classifiers. A single decision tree is constructed recursively based on the weighted training examples and in result only the relevant input features are selected. Therefore such a classifier can be considered as a feature selector.

The standard AdaBoost algorithm performs the binary classification only; to perform multi-class classification, we adopt the AdaBoost.MH algorithm [16] which creates a separate strong classifier for each class. This method requires the training examples to be presented individually for each class with amended labels indicating their affiliation to this specific class.

3.4 Crowd Localisation

A standard GLDM captures the global properties of the entire image, and is therefore unable to localise the detected crowd. In order to obtain localisation information, we applied a pyramidal image division into a set of regions: first, the entire image is divided into four equal parts, then each part is divided into four regions. After splitting, a separate GLDM is calculated and a set of the proposed texture features is extracted from each region. Each splitting step increases the number of regions but at the same time reduces the region area. In result, the localisation ability of the algorithm improves, while its discriminative power (i.e. ability to categorise) decreases. This trade-off should be considered when deciding for the optimal number of the splitting steps. In our work, we analyse results obtained for 0, 1 and 2 splitting steps which corresponds to division into 1, 4 and 16 regions.

The pyramidal division is a key factor of the developed system, since it enables a GLDM-based method to provide localisation information. It has been observed that when the GLDM is evaluated over a portion of the image, the new descriptors based on multilayer shape analysis provide better classification results with respect to the ones used in the literature: the new features are therefore required in order to let GLDM provide strong result with localisation information.

4 Experiments

4.1 Experimental Data

The presented system has been tested on a set of challenging sequences recorded during the real football matches at the local stadium arena. The acquisition system consisted of the four best placed PTZ cameras installed at the venue: three cameras directed at the spectator areas and one placed outside the stadium. Data collection started one hour before the match and ended half an hour after the game was over.

The collected video sequences include people getting into the venue, watching the game and leaving the stadium. Available scenes include several scenarios: the public framed at several distances watching the game, the game field with players, people queuing up for coffee and crowds flowing into and out from the venue. The crowd itself does not always present the same texture density due to the different zoom levels and camera orientations used during the acquisition. As it can be seen, the system is required to correctly face large variety of different situations, with several crowd densities, scales and viewpoints; performed tests are therefore able to measure the generality of the proposed method.

The video resolution was set to 640×480 pixels. From all available data, we have selected 22 different video sequences resulting in 901 images in total that were used for our experiments.

The ground truth data were collected by manually annotating the number of persons in each image region corresponding to image division into 16 parts. These examples were later categorised into a set of classes representing different crowd density according to the criteria presented in Table 1. Ground truth for other image divisions (i.e. into 4 and 1 region) was calculated in a similar way.

Table 1. Experimental data: crowd categories, the corresponding number of persons per image region and a number of examples per class

Category	No crowd	Low	Med	High
Persons	0-1	2-4	5-9	10 and more
Examples	9954	2072	1489	901

4.2 Classification Results

4.2.1 Training and Testing

To train AdaBoost we have used 300 training examples for each class. The number of weak classifiers was set to 50 and the depth of the decision tree set to 3. Each experiment was cross-validated 20 times. The performance of the classifier was verified on 400 testing examples for each class and compared to the ground truth information.

4.2.2 Classification Performance

We have run several tests to investigate the main characteristics of our system. Table 2 presents the classification results obtained when using the original and proposed feature sets alone and using both sets together. In addition, to check the ability of the system to discriminate between different crowd categories we have run the tests for different number of classes including 4 original crowd categories, 3 categories where the Mid and

Table 2. Classification results for different feature sets and different number of classes. The feature sets include: MAR - original feature set, MD - the proposed feature set, MAR&MD - combination of all features.

Classes	MAR	MD	MAR&MD
4	$84.0 \pm 1.0\%$	$84.3 \pm 1.2\%$	$89.4 \pm 0.8\%$
3	$83.2 \pm 1.3\%$	$83.6 \pm 0.9\%$	$88.4 \pm 1.0\%$
2	$85.6 \pm 2.0\%$	$85.2 \pm 1.5\%$	$89.8 \pm 1.4\%$

Table 3. Classification results for different number of image regions

Regions	MAR	MD	MAR&MD
16	84.0 ± 1.0%	84.3 ± 1.2%	89.4 ± 0.8%
4	95.9 ± 0.9%	96.0 ± 0.8%	97.3 ± 0.9%
1	98.7 ± 0.6%	98.7 ± 0.6%	98.3 ± 1.1%

Fig. 3. Six frames presenting different situations of crowd-no crowd. These six frames are correctly classified when they are considered as one region or divided into four regions. However, when divided into 16 regions, the system correctly classifies most of the blocks (as shown by the black blocks in the lower part of the images), correctly locating the crowd. Only few blocks are problematic and misclassified (the grey blocks in the lower part of the images – brighter the block, larger the difference between the classifier output and the ground truth). Each block also contains two words: the classification output of the system (upper row) and the ground truth (lower row).

High class were combined into one and 2 categories with Low, Mid and High classes combined into one.

While performance for these two features sets is comparable (∼84%) combination of all features results in a significant improvement (∼89%). We can also see that this trend is similar for different number of classes and that merging different crowd classes did not significantly affect the results.

To test the trade-off between the system's ability to localise the crowd and accurately estimate its density we run the classification tests for different number of splitting steps (i.e. image regions): default division into 16 regions, division into 4 regions and for the entire image. The results presented in Table 3 indicate that the classification performance drops significantly (by $\sim$8%) when increasing the number of regions from 4 to 16. We can also see that differences in performance for different feature sets are less pronounced for lower number of regions and almost the same when analysing the entire image.

The main sources of misclassification were caused by rich texture content of non-crowded regions, significant image blur and the rough division into image regions. Some of the problematic situations are presented in Fig. 3, where the analysed images are coupled with the classification output, shown as a image that is black when the classification is correct, and that becomes lighter for increasing error entity. The results of the classification provided by our system and the ground truth are also reported, in the first and second row of each block, respectively.

We have also investigated the importance of each input feature, by analysing the weights of individual weak classifiers. In our tests, the features were ranked in the following order, starting from the most important one: homogeneity, entropy, total volume, contrast, blank locations, energy, minimum width, minor axis spread, volume of the peak, max-min area ratio, width-height ration and evenness. Two of the most important features turned out to be from the original set, while the most relevant features from the proposed set include total volume and blank locations. The evenness and width-height ratio were the last in the rank; however these features were also contributing to the final outcome of the classifier.

5 Conclusions and Future Work

We have presented a texture-based crowd detection system that is able to categorise image regions into classes corresponding to different crowd density and to roughly localise the crowd in the image. The presented results demonstrate that the additional proposed features improve the classification performance in smaller image areas. We have also demonstrated a trade-off between the localisation and classification ability of the system and investigated the importance of individual features.

Possible extensions to the system include incorporation of motion cues (e.g. based on motion flow analysis) for detection of moving crowds, the use of additional features (e.g. Gabor filters), combining information from multiple sensors, tracking individuals and groups of people in crowds, etc. It should also be possible to apply regression models to obtain continuous density estimates instead of discrete density categories.

References

1. Marana, A.N., Velastin, S.A., Costa, L.F., Lotufo, R.A.: Estimation of crowd density using image processing. In: IEE Colloquium on Image Processing for Security Applications (Digest No.: 1997/074), pp. 11/1–11/8 (March 1997)

2. Davies, A.C., Hong Yin, J., Velastin, S.A.: Crowd monitoring using image processing. Electronics Communication Engineering Journal 7(1), 37–47 (1995)
3. Mehran, R., Oyama, A., Shah, M.: Abnormal crowd behavior detection using social force model. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, pp. 935–942 (June 2009)
4. Ihaddadene, N., Djeraba, C.: Real-time crowd motion analysis. In: 19th International Conference on Pattern Recognition, ICPR 2008, pp. 1–4 (December 2008)
5. Arandjelović, O.: Crowd detection from still images. In: Proc. British Machine Vision Conference (BMVC) (September 2008)
6. Haralick, R.M.: Statistical and structural approaches to texture. Proceedings of the IEEE 67(5), 786–804 (1979)
7. Rahmalan, H., Nixon, M.S., Carter, J.N.: On crowd density estimation for surveillance. In: The Institution of Engineering and Technology Conference on Crime and Security, pp. 540–545 (June 2006)
8. Kong, D., Gray, D., Tao, H.: Counting pedestrians in crowds using viewpoint invariant training. In: Proc. British Machine Vision Conference (BMVC) (September 2005)
9. Kong, D., Gray, D., Tao, H.: A viewpoint invariant approach for crowd counting. In: 18th International Conference on Pattern Recognition, ICPR 2006, vol. 3, pp. 1187–1190 (September 2006)
10. Gárate, C., Bilinsky, P., Bremond, F.: Crowd event recognition using hog tracker. In: 2009 Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS-Winter), pp. 1–6 (December 2009)
11. Marana, A.N., Costa, L.F., Lotufo, R.A., Velastin, S.A.: On the efficacy of texture analysis for crowd monitoring. In: Proceedings of SIBGRAPI 1998, International Symposium on Computer Graphics, Image Processing, and Vision, pp. 354–361 (October 1998)
12. McKenna, S.J., Jabri, S., Duric, Z., Rosenfeld, A., Wechsler, H.: Tracking groups of people. Computer Vision and Image Understanding 80(1), 42–56 (2000)
13. Gennari, G., Hager, G.D.: Probabilistic data association methods in visual tracking of groups. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004, vol. 2, pp. II–876–II–881 (June 2004)
14. Lau, B., Arras, K.O., Burgard, W.: Tracking groups of people with a multi-model hypothesis tracker. In: IEEE International Conference on Robotics and Automation, ICRA 2009, pp. 3180–3185 (May 2009)
15. Freund, Y., Schapire, R.E.: A Decision-Theoretic Generalization of On-line Learning and an Application to Boosting. In: Vitányi, P.M.B. (ed.) EuroCOLT 1995. LNCS, vol. 904, pp. 23–37. Springer, Heidelberg (1995)
16. Friedman, J.H., Hastie, T., Tibshirani, R.: Additive logistic regression: a statistical view of boosting. Technical report, Dept. of Statistics, Stanford University (1998)
17. Breiman, L., Friedman, J., Olshen, R., Stone, C.: Classification and Regression Trees, 1st edn. Wadsworth and Brooks, Monterey (1984)