Detecting Individual in Crowd with Moving Feature's Structure Consistency

Yuanhao Yu Zhen Lei Dong Yi Stan Z. Li*
Center for Biometrics and Security Research & National Laboratory of Pattern Recognition
Institute of Automation, Chinese Academy of Sciences
95 Zhongguancun Donglu, Beijing 100190, China
{yhyu,zlei,dyi,szli}@nlpr.ia.ac.cn

Abstract

In this paper, we present a method for detecting individuals in crowd by clustering a group of feature points belonging to the same person. In our approach, a feature point is considered to contain three attributes: the motion trajectory in video sequence, the sparse local appearance around point in current frame, and the structure relationship with body center related with local appearance. We exploit these attributes to cluster them appearing on the same individual to achieve detection purpose. The algorithm does not require observing entire human body and could discriminate different individuals under overlap. Our experiments show that this approach advances the performance of detecting individuals in crowds.

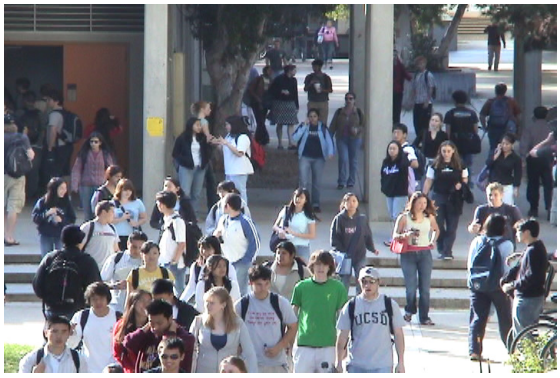

Figure 1. Example of a dense crowd. Our goal is to detect individual in video sequences like this.

1. Introduction

This paper addresses the problem of detecting individuals in real world dense crowds. The topic is a fundamental to further high-level visual analysis and some applications in video surveillance, such as people counting and abnormal event detection.

The phenomenon of crowding presents numbers of challenges for visual analysis. Occlusion and complex scene are the most important two factors. When dense crowd occurs, moving objects usually fill the scene, which precludes the traditional techniques based on background subtraction [17, 18, 5, 19, 4]. And, high occurrence of occlusion makes it impossible that all the parts of an individual are observed all the time in the video sequence. In consequence, traditional model-based techniques[15, 10, 11, 3] also fail to achieve robust and accurate result.

In contrast with traditional techniques, a moving objects detection framework[2, 8, 12] has been proposed, which only makes use of motion characteristics. In the framework, feature points are tracked in video sequence to generate motion trajectories. Then, each pair's similarity of feature points is measured according to two attributes, the average of space distances on each frames and the maximal variation of these distances. Finally, the detection task is translated to a problem of clustering those feature points using their similarities. Since background subtraction and observing all portions are not required, this framework is more robust to occlusion and gets better performance in crowd.

However, in real world scene, the framework fails when objects move closely and in the same direction. The reason is trajectories tend to be extremely similar and objects can not be segmented from crowd correctly. Recently, Daisuke uses the consistency of local color to measure the similarity of features in order to overcome this problem [16], which assumes that local color of the space between objects is continuously changing in video sequence. In ideal scene, it can deal with the situation correctly. Nevertheless, the assumption is not always satisfied, such as the situation that background seems the same color. Besides, the technique requires feature points tracked very accurately which is hard to meet in practice.

In addition, all pervious approaches can not detect individual actually but just segment moving objects from

*Stan Z. Li is the corresponding author.

2011 IEEE International Conference on Computer Vision Workshops
978-1-4673-0063-6/11/$26.00 ©2011 IEEE

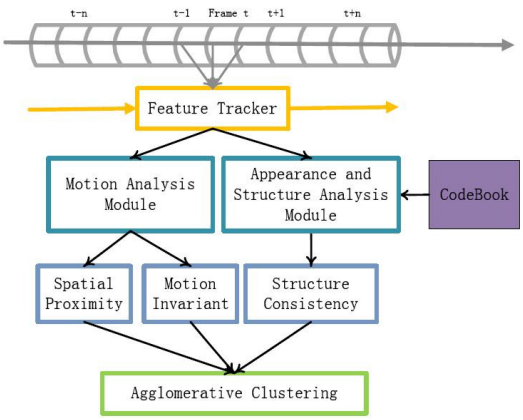

Figure 2. The Framework of Our Algorithm

crowd. Because the clustering-based framework does not contain appearance information. Using it to detect individual, people have to assume all moving objects in the scene are human.

To improve performance, not only motion characteristic but also appearance and structure information are used in our algorithm. We present a novel measurement of moving features called feature's structure consistency (FSC), which makes use of appearance and structure priori as the cue. It's known to all that object's local appearance contains structural relationship with object center. For example, if the position of individual's head is already known, we could infer body's center on a image. So using appearance priori, we can recognize the local appearance around feature on individual and obtain its structural relationship with center. According relationship and position of feature, a possible position of individual center can be gained. Because possible centers inferred by each features tend to be proximity when these moving features are belonging to the same individual. FSC could makes use of this characteristic by measuring features through calculating distances of those centers. By utilizing FSC, our algorithm is more robust to the situation that individuals walk closely in the same direction. Furthermore, using appearance and structure priori, we discriminate human with other objects in crowd.

In our approach, we have used three kinds of measures. Two kinds of measures for motion are obtained through analyzing features' trajectories. The other FSC measure is obtained via recognizing appearances around features using priori. We design a automatical process to get the priori information before detection. All kinds of measures are used as the similarities of clustering method. We implement agglomerative clustering method to cluster these features.

The main contribution of this work is we introduce appearance and structure priori into the framework for the first time by proposing a novel measurement of moving features called feature's structure consistency. The algorithm's merits include: 1)it is more robust to dense crowd for detecting individual; 2)Using FSC to discriminate human with others, we make the framework detect individuals in crowd firstly; 3)the system can deal with the situation more effectively that individuals walk closely and in the same direction; 4)we design a automatical training process to obtain priori.

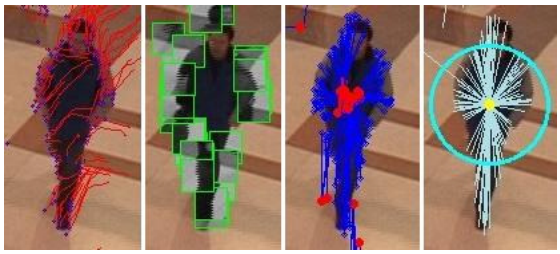

Figure 3. Middle Results and Final Result: Tracking features to generate trajectories; Using CodeBook entries to replace the extracted patches; Link features with their hypothetical centers; The final detection result.

2. Clustering Framework to Detect Individual

In the approach, detecting individual in crowd is achieved by clustering moving features by their similarities. The system can be divided in three steps: tracking feature points, calculating similarities of each pair of features, and clustering features, which is shown in Figure 2.

First of all, KLT algorithm is used to track features in video sequence[13, 14]. At each frame, new feature points would be detected to make sure every moving individuals contain enough feature points. We abandon the features generating fractured and violently changing trajectories which could be considered as noises in our system. The features which do not move in a long term are also not utilized as most of them are on the background.

Then, according to trajectories and appearances around every features, analysis modules generate three similarities for each pair of features. The motion analysis module achieves Spatial Proximity Similarity and Motion Invariant Similarity; the appearance and structure analysis module obtains Structure Consistency Similarity. Note that the latter is also used to judge a feature belonging to an individual or not and we abandon other objects' features to just detect individuals. The details of analysis modules are given in next section. In our approach, three similarities are combined together using equation(1), where c_i is one of features, $S_{sp}(c_i, c_j)$ is the Spatial Proximity Similarity, $S_{di}(c_i, c_j)$ is the Motion Invariant Similarity, $S_{fsc}(c_i, c_j)$ is the Structure Consistency Similarity and $S(c_i, c_j)$ is the combined similarity. The similarities are illustrated in Fig-

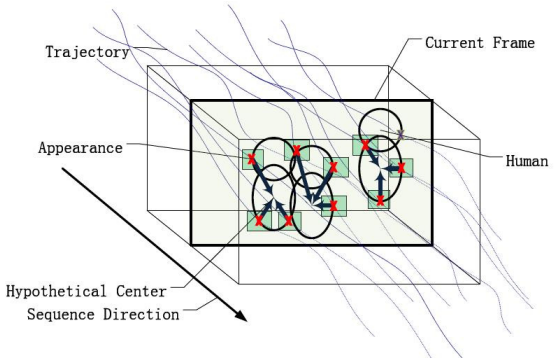

Figure 4. The illustration of three similarities in our algorithm: Spatial Proximity Similarity and Motion Invariant Similarity are generated by trajectories shown and Structure Consistency Similarity are obtained by appearance and priori.

ure 4.

$$S(c_i, c_j) = S_{sp}(c_i, c_j) \cdot S_{di}(c_i, c_j) \cdot S_{fsc}(c_i, c_j) \quad (1)$$

Using features' similarities, we design a cluster method based on agglomerative clustering [1] ensuring that system has a repeatable result. We do not initialize a set with one feature but a group features with high similarities to speed up the criterion and make the system more robust to shared features. At each iteration, two closet sets are merged if their distance is smaller than an threshold and the criterion stops when all sets are considered and no pair is merged. As a result, each group of features represent a individual. Figure 3 shows the middle and final results.

3. Similarity between Features

Three kinds of similarities between each pair of features are calculated, called the Spatial Proximity Similarity, the Motion Invariant Similarity and the Structure Consistency Similarity. We would describe the Structure Consistency Similarity firstly.

3.1. Priori and CodeBook

To attain the Structure Consistency Similarity, we need collect appearance and structure priori information before detection. An automatical training process is designed to obtain the priori and we store the priori in a structure, called CodeBook.

Before training process, a video sequence is prepared that there are sparse individuals moving in the scene so that they can be segmented by background subtraction. Using background modeling technique [9], we cut a set of individual images from the sequence. Then, the rest process can be divided in two steps. For each image, a DoG interest point operator is applied to extract fixed size image patches. An clustering scheme [6] is used to cluster these patches and makes the result cluster centers form a compact representation of local appearance. We store these cluster centers representing appearance as appearance prior. In the next step, we perform a second iteration over the collected images to learn the structure priori for each cluster center. Patches are extracted from images again and matched with cluster centers using Normalized Greyscale Correlation (NGC) [6]. Once a cluster center could be matched with a patch with similarity higher than α, it records the relationship of location between the patch and individual center. Using all of relationships recorded, an cluster center learn a relative distribution for individual center, which represents a structure priori information actually. Finally, we put a cluster center and its distribution together and store them. We call a pair of cluster center and distribution an CodeBook entry $(I_k, p(\lambda|I_k))$, where I_k is the cluster center, $p(\lambda|I_k)$ is the distribution, and λ is a location $\lambda = (\lambda_x, \lambda_y)$. And all of CodeBook entries construct an CodeBook. In fact, a CodeBook entry could express a local appearance with the cluster center and its structure relationship with the distribution. Because we collect enough CodeBook entries in the process mentioned, the CodeBook could express most local appearances appearing possibly on an individual's body and their relationships with body's center.

3.2. Hypothetical Center

According to appearance around moving feature point at each frame, a possible individual center called hypothetical center can be inferred for each feature.

$$p(\lambda|e, l) = \sum_k p(\lambda|I_k, l) p(I_k|e) \quad (2)$$

Given a location on a frame, the possible position of individual center could be inferred in a probabilistic voting procedure. The same patch extraction method mentioned is used on the location. We use the patch to match with CodeBook, and the matching CodeBook entries cast votes for the possible position on the image plane based on learned distribution and the patch's location. The voting result can be considered as a conditional distribution of center $p(\lambda|e, l)$, where e is the observation, the extracted image patch, and l is the observation location. The whole procedure could be formulated in equation (2). We translate the matching similarity to the probability $p(I_k|e)$ to weight the matching CodeBook entry. In the CodeBook, the $p(\lambda|I_k)$ describes a stored relative distribution for center. According to the location of patch l, $p(\lambda|I_k, l)$ presents the distribution of center on the current frame.

$$p(\lambda|c_i^t) = \sum_j p(\lambda|e_j, l_j)p(e_j, l_j|c_i^t) \quad (3)$$

$$= \sum_k \sum_j p(\lambda|I_k, l_j)p(I_k|e_j)p(e_j, l_j|c_i^t) \quad (4)$$

$$= \sum_k \sum_j p(\lambda|I_k, l_j)p(I_k|e_j)p(e_j|c_i^t) \quad (5)$$

In our approach, we want to detect every individual center for each moving feature based on the appearance around feature. Because there are much noise in real world crowd. We do not extract one patch on the location of feature at current frame c_i^t ($c_i^t \in \{\lambda\}$) but use all the observations around the position to infer the center's probabilistic distribution $p(\lambda|c_i^t)$, which is shown in Figure 5. In a small area around feature, interest point detector is utilized. For every point, the same probabilistic voting procedure is implemented and a uniform voting space is constructed for each feature. It could be expressed by the equation (3), where we weight every detected point by $p(e_j, l_j|c_i^t)$ which is proportional to the distance between the point and feature's location. Taking equation (2) into the formula, we could get expression (4). Because on every location l_j, a single observation e_j can be obtained. The $p(e_j, l_j|c_i^t)$ is equal to $p(l_j|c_i^t)$ and we obtain equation (5). In this way, we get a center's probabilistic distribution for each moving feature. Thus, an possible individual center could be founded at maxima in voting space [7]. Because we infer the center according to the appearance around a feature, we call it the moving feature's Hypothetical Center O_{c_i} ($O_{c_i} \in \{\lambda\}$).

$$O_{c_i} = \arg\max_{\lambda} p(\lambda|c_i^t) \quad (6)$$

3.3. Features on Individuals

In our algorithm, we discriminate human and others by differing the features on them based on appearance and structure priori. If a feature belongs to an individual, the appearance observation should be recognized and the hypothetical center should be near to the truth individual center. These requirement are expressed by two constraints (7).

$$\begin{cases} \omega(O_{c_i}) \cdot p(O_{c_i}|c_i^t) > \beta \\ \sum_j NGC(I_k, e_j) \cdot p(l_j|c_i^t) > \alpha \end{cases} \quad (7)$$

The upper constraint expresses feature's hypothetical center should appear at the location where real individual center occurs in a high confidence. A weight function $\omega(\lambda_i)$ expressing the possibility of individual center's appearing is used as a standard to measure the displacement of feature's hypothetical center. We calculate the weight function by combining all distributions of moving features shown in (8). It is based on the fact that using all information tends to weaken the influence of inaccuracy and a group features on same person get similar distribution.

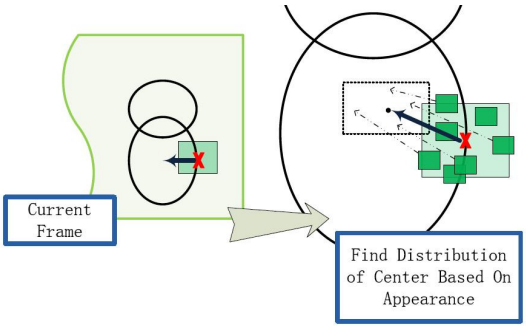

Figure 5. The process of getting distribution of center: The green rectangles are the patches, the thin arrows stand for $p(\lambda|I_k, l_j)$ and the thick arrow represents $p(\lambda|c_i^t)$.

$$\omega(\lambda) \sim \sum_i p(\lambda|c_i^t) \quad (8)$$

The below one represents appearance restriction. It means that the average similarity between appearance and priori should be high enough. In the equation, e_j is the observation patch and I_k is the CodeBook entry matched with e_j.

3.4. Feature's Structure Consistency

As mentioned, we already obtain a hypothetical center for each features. In our approach, the distance of hypothetical centers is used to measure the similarity of two features (9), where ξ_{fsc} is a weight factor. Since, the hypothetical center is expected around the truth center, the measurement makes every features appearing on the same individual tend to be very similar. And, as we make use of a spare local appearance around feature mentioned, the similarity is robust to occlusion in dense crowd.

$$S_{fsc}(c_i, c_j) = \frac{1}{1 + \xi_{fsc}||O_{c_i} - O_{c_j}||_2} \quad (9)$$

3.5. Spatial Proximity and Motion Invariant

The Spatial Proximity and Motion Invariant are also utilized in our approach to cluster features. Since, the trajectories tend to be remain in close proximity in sequence when they belong to the same one, the maximum displacement of trajectories is considered as the Spatial Proximity Similarity which has been used in [12]. And, we define the Motion Invariant by the average variation of distance between trajectories like [2].

(a) (b)

Figure 6. Results on PETS2001: (a) is result of pervious method and (b) is our result.

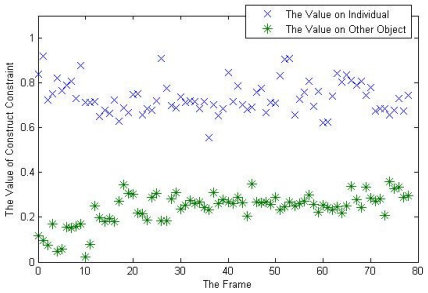

Figure 7. The histogram illustrates the ability of discriminating individuals and others: the value of y axis is structure constraint. We could observe the obvious difference on the value between individual and vehicle.

$$S_{sp}(c_i, c_j) = \frac{1}{1 + \xi_{sp} \max_{t' \in Time} ||c_i^{t'} - c_j^{t'}||_2} \quad (10)$$

$$S_{di}(c_i, c_j) = \frac{1}{1 + \xi_{di} Var(c_i, c_j)} \quad (11)$$

4. Experimental and Analysis

In order to demonstrate the robustness and effectiveness of our approach, we implement our algorithm on four datasets including different densities of crowds and different moving objects in scene. We also tests pervious techniques for comparison.

4.1. Effectiveness of differing individuals with other objects

To examine effectiveness of discrimination ability, we test our algorithm and pervious techniques on PETS2001, where individuals and vehicles are both appearing in the scene. A nearest distance tracking method based on the detection algorithm at each frame is used to illustrate the detection performance in a long term. Figure 6 a and b are the same frames chosen from different results. In our result, individuals' moving features are sifted for clustering and the system only detect individual. In pervious method, both individual and car are considered as human. As it uses a space constraint according to human size, the car's features are clustered into two parts. To better explain discrimination ability, we choose 70 frames from video sequence and calculate the value of structure restraint, which is shown in Figure 7. Because the value of weight is usually large and does not contain probabilistic meaning, we translate the restraint value to range form 0 to 1 for easy observation. The blue points are average values of individual's features, the green ones are car features and they are easily to be differed.

(a) (b)

Figure 8. Results on PETS2006: (a) shows Vincent's and (b) shows ours

4.2. Robustness in discrimination of individuals

The situation that people are walking closely in the same direction is a tough problem for the framework. In this subsection, we would like to show that our algorithm can deal with the situation well. Since, Daisuke uses the assumption that local color between individuals is constantly changing to improve the framework [16], we also compare it on a open dataset UCSD.

To explain the ability of our algorithm for handling the mentioned problem, the algorithm is implemented on PETS2006 and Vincent's citeCVPR06C02 which only does not use FSC is tested for comparison too. PETS2006 is a dataset where sparse individual walking on simple background and people usually walking equidirectionally and closely. Figure 8 shows two key frames from different algorithms, where we find our system segment the crowd effectively which can not be achieved by Vincent's algorithm. Figure 10 shows the similarity of different clusters in both approaches: a point stands for the value of average similarities of different clusters for one algorithm at one frame. We could observe that features on different persons become more dissimilar by using FSC.

Approach	Precision	Recall
Vincent's	0.664	0.353
Daisuke's	0.883	0.591
Ours	0.935	0.598

Table 1. The Recall and Precision Rates of experimental results on UCSD of three approaches.

As UCSD is an open dataset where dense crowds fill the scene and always used to test robustness for pervious approaches, we use it to compare the performance of three algorithms, ours, Vincent's and Daisuke's. The precision and recall rates of the result is shown in Table 1 (the detail of two rates described in next subsection). Because we and Daisuke both try to overcome the hard problem, the two approaches performs better than Vincent's. Moreover, our precision is 5% higher than Daisuke's and the recall is higher too. We reason that when color between individuals is not changing our system works more effectively.

4.3. Results on real world crowd

We examine the performance of our method in real world crowd on three datasets in this part, the PETS2006, the UCSD, and our surveillance sequence. Our dataset is an challenging one as there are high density crowd filling the scene, where 35 individuals are in crowds with heavy occlusion on average at each frame. We also test Vincent's approach in the three sequences for comparison (The comparison for Daisuke's is shown in 4.2). The test results are shown in Figure 9, 11-16 and Table 2. In each frame, the clustering center are linked with moving features and a circle is drawn using clustering center as circle center and average distance between center and features as radius.

We defined the true positive, that one cluster is formed for one person. The false positive contains two situations: 1) one individual is divided into more than one clusters, 2) multiple individuals are clustered as one cluster.

The result shows that there are more clusters containing multiple individuals while our performance is much better . Furthermore, lacking enough discrimination information makes Vincent's system generate extreme fractured clusters. Abandoning them would produce losing detection shown in Figure 12 d while remaining them could result in multiple clusters' appearing on single person shown in Figure 12 b. Since the structure information tends to group one person's features together, our results seem better. More results are shown in Figure 14-16.

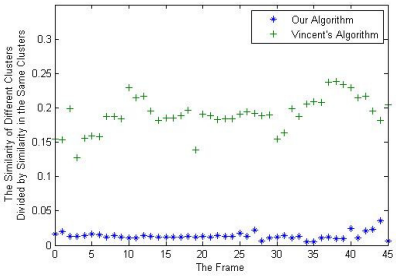

Figure 10. The histogram shows the similarity of different clusters in both approaches

From Table2 and Figure 13, we observe that our system is better than pervious approach on precision and recall rates especially in dense crowd. Although, the recall rate descends as the crowd density becomes higher, our algorithm's recall rate stays a relatively high level which is 20% higher than pervious in heavy crowds. And, our precision rate of our algorithm stays high in three sequences and still higher than other approaches. The result shows that our system is robust for detecting individual in crowds.

In observation, we find that our algorithm fails when an individual acts complex motion making the feature points move irregularly and an individual is observed a extremely special appearance which could not be recognized by appearance priori. Besides, little moving features would also weaken the performance of the system. We would like to dress the problems in our future work.

5. Conclusions

In this work, we introduce appearance and structure priori into the feature clustering framework by a novel mea-

Figure 9. Our result on UCSD

Figure 11. Vincent's result on UCSD

(a) (b) (c) (d)

Figure 12. a is the result of our algorithm on PETS2006, b is Vincent's result, c is output of our system on our dataset and d is the result of pervious method

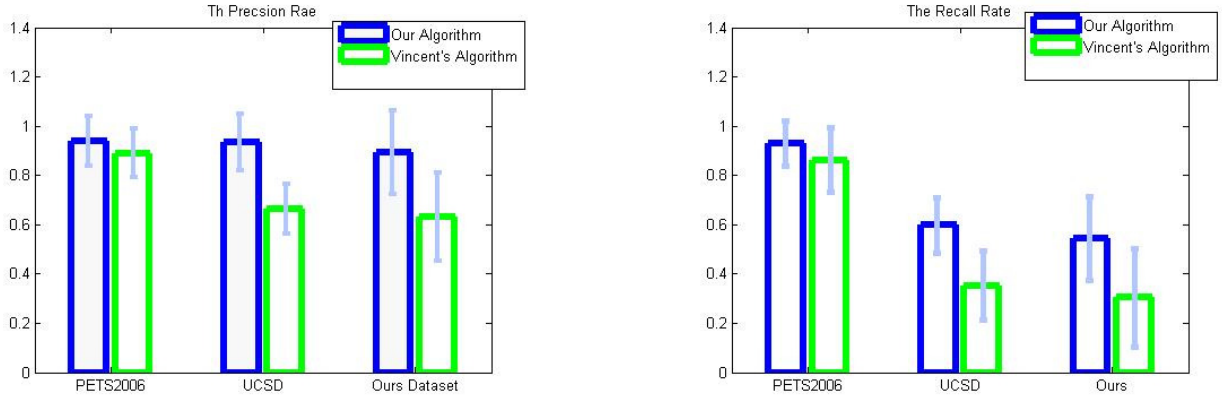

Figure 13. The left bar shows the average of precision rate for each datasets and the recall rate is shown on right

DataSet	*N*	*Our Algorithm*				*Vincent's Algorithm*			
		Recall Rate		*Precision Rate*		*Recall Rate*		*Precision Rate*	
		μ	σ	μ	σ	μ	σ	μ	σ
PETS2006	6	0.931	0.092	0.938	0.100	0.862	0.130	0.890	0.100
UCSD	21	0.598	0.112	0.935	0.116	0.353	0.140	0.664	0.102
Ours	33	0.545	0.170	0.891	0.170	0.303	0.200	0.631	0.181

Table 2. Detection results on three datasets: N denotes the average number of people in each frame, μ and σ presents the mean and the standard deviation.

surement of features called feature's structure consistency. Our approach could discriminate individuals with other objects, which enables the framework actually detect individuals in crowds for the first time. And, the situation can be better solved that individuals are walking closely in the same direction in crowds. Our experiments on four datasets show that the approach improves the performance of detecting individual in crowd obviously.

Acknowledgements

The authors would like to acknowledge the following funding sources: the Chinese National Natural Science Foundation Project #61070146, the National Science and Technology Support Program Project #2009BAK43B26, the AuthenMetric R&D Funds (2004-2011), and the TABULA RASA project (http://www.tabularasa-euproject.org) under the Seventh Framework Programme for research and technological development (FP7) of the European Union (EU), grant agreement #257289.

References

[1] P. Bickel, P. Diggle, S. Fienberg, U. Gather, I. Olkin, and S. Zeger. The elements of statistical learing. *Springer*, 2009. 3

[2] G. J. Brostow and R. Cipolla. Unsupervised bayesian detection of independent motion in crowds. *CVPR*, pages 594–601, 2006. 1, 4

[3] Felzenszwalb, McAllester, and Ramanan. A discriminatively trained, multiscale, deformable part model. *CVPR*, 2008. 1

Figure 14. Detection Results of our system on PETS2006

Figure 15. Detection Results of our system on UCSD

Figure 16. Detection Results of our system on Our Surveillance Sequence

[4] W. Ge and R. Gollins. Evaluation of sampling-based pedestrian detection for crowd counting. *PETS*, 2009. 1

[5] Z. Kim. Real time object tracking based on dynamic feature grouping with background subtraction. *CVPR*, 2008. 1

[6] B. Leibe and B. Schiele. Interleaved object categorization and segmentation. *BMVC*, 2003. 3

[7] B. Leibe, E. Seemann, and B. Schiele. Pedestrian detection in crowded scenes. *CVPR*, pages 878–885, 2005. 4

[8] Y. Li and H. Ai. Fast detection of independent motion in crowds guided by supervised learning. *ICIP*, pages 341–344, 2007. 1

[9] S. Liao, G. Zhao, V. Kellokumpu, M. Pietikainen, and S. Z. Li. Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes. *CVPR*, pages 1301–1306, 2010. 3

[10] Y. Liu and S. Shan. Spatial-temporal granularity-tunable gradients partition descriptors for human detection. *ECCV*, 2010. 1

[11] T. Malisiewicz, A. Gupta, and A. Efros. Ensemble of exemplar-svms for object detection and beyond. *ICCV*, 2011. 1

[12] V. Rabaud and S. Belongie. Counting crowded moving objects. *CVPR*, pages 705–711, 2006. 1, 4

[13] E. Rosten and T. Drummond. Fusing points and lines for high performance tracking. *ICCV*, pages 1508–1515, 2005. 2

[14] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. *ECCV*, pages 430–443, 2006. 2

[15] W. R. Schwartz, A. Kembhavi, D. Harwood, and L. S. Davis. Human detection using partial least squares analysis. *ICCV*, pages 24 – 31, 2009. 1

[16] D. Sugimura and K. M. Kitani. Using individuality to track individuals: Clustering individual trajectories in crowds using local appearance and frequency trait. *ICCV*, pages 1467 – 1474, 2009. 1, 5

[17] T.Zhao and R. Nevatia. Tracking multiple humans in complex situations. *IEEE Trans. Pattern Anal. Mach. Intell*, pages 26(9):1208–1221, 2004. 1

[18] T.Zhao and R. Nevatia. Tracking multiple humans in crowded environment. *CVPR*, pages 406–413, 2004. 1

[19] L. Wang and N. Yung. Bayesian 3d model based human detection in crowded scenes using efficient optimization. *WACV*, 2011. 1