

# Robust Object Tracking based on Detection with Soft Decision

Bo Wu, Li Zhang, Vivek Kumar Singh, and Ram Nevatia  
University of Southern California  
Institute for Robotics and Intelligent Systems  
Los Angeles, CA 90089-0273  
{bowu|zhang11|viveksin|nevatia}@usc.edu

## Abstract

*This paper presents a detection based object tracking method that forms object trajectories by associating detection responses. Discriminative classifiers of objects of a known class are learned and applied to the video sequence frame by frame. The output of the detection module is a “soft decision”, which consists of a set of detection responses of different confidence levels. Responses of different confidence levels are generated by classifiers with different complexities. The cheap classifiers are applied to the whole image first, while the expensive classifiers are only applied to the region accepted as object by the cheap classifiers. Object trajectories are initialized from the responses of higher confidence; hypothesized objects are tracked by associating with all the responses in the order of their confidence levels. The proposed approach is applied to the problems of human tracking in indoor meeting videos and outdoor surveillance videos. The system is evaluated on two public video corpora and compared with some previous methods.*

## 1. Introduction

Object detection and tracking is a fundamental problem of computer vision research and very important for many real-life applications, such as visual surveillance and human computer interaction. There are three main types of tracking methods: 1) 2-D image region tracking, 2) moving blob tracking, and 3) object detection based tracking.

2-D region tracking algorithms focus on the problem of tracking *after* initialization by assuming that the position of the object is given in the first frame. The algorithms first try to extract some characteristic properties of the object, could be color [17] or salient points [18], from the initialization and then use these properties to track the object in the new frames. These types of trackers are not fully automatic and usually sensitive to object deformation, occlusion, and

initialization. The moving blob tracking algorithms focus on the detection and tracking of moving objects. First motion segmentation is applied and then the moving blobs are tracked based on their appearance and motion, *e.g.* [14]. The motion segmentation algorithms are sensitive to abrupt illumination changes, shadows, and reflections. The main limit of these types of methods is that they do not have a discriminative model of the object class; hence, they can not really detect and track specific objects.

Detection based tracking methods attempt to overcome these limitations by using discriminative methods. Recently, fast development of object detection techniques has resulted in many promising methods for detection of particular object classes, *e.g.* faces [4, 16] and pedestrians [3, 2, 5, 10, 15]. These object detectors produce good observations for the detection based tracking algorithms. By associating the frame by frame object detection responses, we can answer the question of when to start or terminate an object trajectory. However, the detectors are not perfect: the detection performance is usually a tradeoff between the detection rate and the false alarm rate. The missed detections and false alarms provide misleading information to the tracking algorithms. To improve the robustness of object tracking, we propose a method that makes “soft decisions” at the detection stage by producing detection responses of different confidence levels and uses these confidence levels for associating the detection responses to form trajectories. We apply our approach to the problems of human detection and tracking in meeting videos and surveillance videos.

### 1.1. Related Work

Detection based tracking algorithms obtain object hypotheses by applying a discriminative object model to images. The discriminative object models are learned off-line from labeled training data. The models used are of great variety, *e.g.* edge template matching [11], boosted ensemble classifiers with cascade decision strategy [16, 5, 4, 3], SVM classifiers [10, 2], salient feature based constellation type of models [13, 7], and random field based models [8]. Given

a new sequence, discriminative object models are used to search the object in each frame. The detection responses include rough segmentation, *e.g.* bounding boxes, of the objects. The main errors of the detection methods are false alarms and missed detections.

To track the objects, the detection responses at different frames are linked together to form the object trajectories. Wu and Nevatia [3] define an affinity measure between detection responses based on cues from position, size, and color and use the Hungarian algorithm [19] to associate object hypotheses and detection responses. New object trajectories are initialized whenever the detection responses do not match with any existing trajectories for a certain number of frames; old trajectories are terminated when they are lost by the detector for a certain number of frames. Li *et al.* [1] and Okuma *et al.* [12] use particle filter methods to associate the detection responses of an unknown number of objects. The detection responses are used to generate new particles and evaluate existing particles. By particle filtering, the tracker can maintain multiple hypotheses. However, increasing the number of particles requires more computational cost. To improve the computational efficiency, Li *et al.* [1] use multiple detectors (observers) to form a cascade particle filter. The order in which the detectors are applied is determined based on their computational costs: the faster the earlier. The detection based tracking methods are fully automatic. They do not need outside initialization.

Most existing detection methods output hard decisions *i.e.* binary value predictions. Hard decisions extract high level information (for detection the position and size of the objects) from raw data (images or sequences of images). This greatly reduces the problem space for further processing; however, there is also a tradeoff between detection rate and false alarm rate. To achieve good tracking performance, we need both a low false alarm rate to avoid false object trajectories, and a high detection rate to track the objects most of the time. Although hard decisions at intermediate stages make the system less robust, we can not avoid making any decisions. Recovering trajectories directly from image sequences or some real-valued probability field is not feasible.

## 1.2. Outline of Our Approach

Fig. 1 gives a schematic diagram of our system. Our approach is a detection based tracking method. Our detection module outputs a set of responses with different confidence levels. The detection module consists of three classifiers: 1) an edgelet based boosted classifier with a cascade decision strategy [3], 2) an edgelet based boosted classifier with a single-threshold decision strategy, and 2) a Histogram of Oriented Gradients (HOG) descriptors [10] based SVM classifier. In terms of computational cost, the single-threshold boosted classifier is the most expensive one, as it requires evaluating hundreds of edgelet features before a

detection decision is reached; the cascade boosted classifier is the fastest one, as its cascade decision strategy throws off most negative samples at very early stage of the cascade.

The order in which the classifiers are applied is determined based on their computational costs. Basically, the faster a classifier, the earlier it is used. Given a new frame, first the cascade boosted classifier is used to scan objects at all possible positions and of various sizes. The positive responses of this stage are sent to the SVM classifier for verification. When an existing trajectory can not be found by the first and second level detection responses, the single-threshold boosted classifier is activated around the predicted position of the trajectory; this significantly reduces the search space of the single-threshold boosted classifier.

In terms of accuracy, the outputs of the three classifiers can be seen as three points on a Receiver Operating Characteristic (ROC) curve. The first level has the lowest false alarm rate but also the lowest detection rate; the third level has the highest detection rate but also the highest false alarm rate. The three levels of detection responses together form a soft decision.

The basic idea of improving tracking robustness is to initialize trajectories with a detection decision strategy of low false alarm rate and track the objects with a decision strategy of high detection rate. In our system, we initialize trajectories from the first level detection responses; and track the objects from the second and third level detection responses. Although the detection rate of the first level is relatively low, when the object is present for a large period, the probability that it is detected at at least some point is high. Missed detection may result in some delay in trajectory initialization, but this can be compensated for by tracking in both forward and backward directions.

The rest of the paper is organized as follows: section 2 and section 3 describe our detection and tracking methods respectively; section 4 shows the experimental results; some conclusions and discussions are given in the last section.

## 2. Detection Module

Following previous work, we learn the edgelet based boosted classifiers [3] and the HOG based SVM classifier [10] for an object class. We choose these two types of features, because they are complementary. The edgelet features are designed to encode the local silhouette explicitly but relatively sensitive to small transforms (translation, rotation, *etc.*). The HOG descriptor encodes the statistics of a sub-region and is robust to small transforms, but does not maintain information about which pixels contribute to the histogram bins. Very different shapes could result in the same histogram.

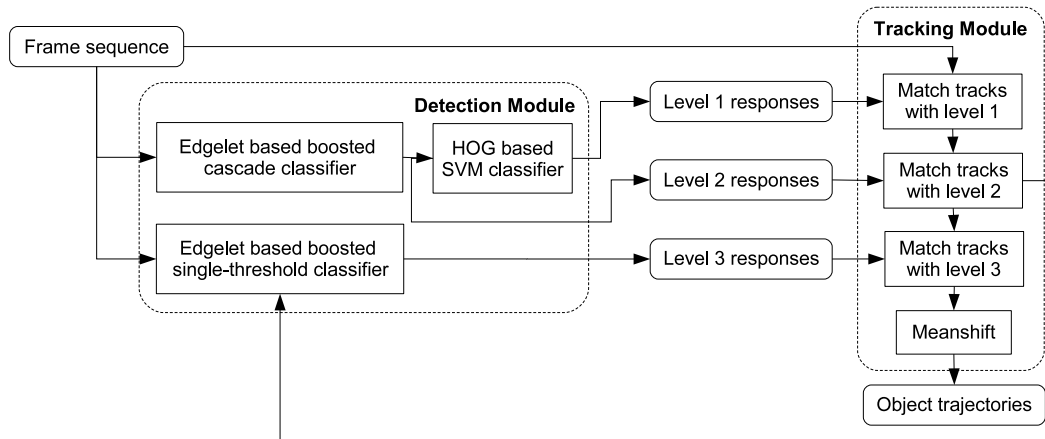


Figure 1. System diagram.

## 2.1. Edgelet based Boosted Classifier

Edgelets are one type of local shape features. One edgelet can be seen as a small edge template. For each edgelet in a big feature pool, one weak classifier is built to distinguish objects from background. Then a boosting algorithm [4] is used to learn tree structured classifiers for multi-view objects. Each node of the tree is an ensemble classifier with the cascade decision strategy [16] which makes the detection process efficient. In our work, following [3], we learn a two level tree classifier for our object classes.

During the training of the tree classifier, a cascade decision strategy is learned for each branch of the tree. For each node along a branch, a threshold is chosen to accept most positive samples (a positive passing rate of 0.998 in our experiments) while rejecting as many negative samples as possible. When a target overall false alarm rate ( $10^{-6}$  in our experiments) is reached, the training procedure is terminated. Because of the cascade decision strategy, most sub-windows examined in the image can be discarded by computing only the first few features in the tree. Although this is an efficient strategy, it is aggressive in terms of discarding negative samples. To obtain a prediction of high detection rate, we ignore the series of thresholds of the cascade decision strategy and learn an overall threshold for each branch of the tree. The threshold is chosen to accept *all* positive samples and reject as many negative samples as possible. However, as the decision is made at the leaf nodes of the tree, all features in the classifier need to be computed to classify one sub-window. The classification results of the boosted classifier with the cascade decision strategy are the second level responses; the results of the single-threshold boosted classifier are the third level responses. Fig. 2(b) and Fig. 2(c) show an example of each of these two levels for the problem of meeting room human (head-shoulder) detection.

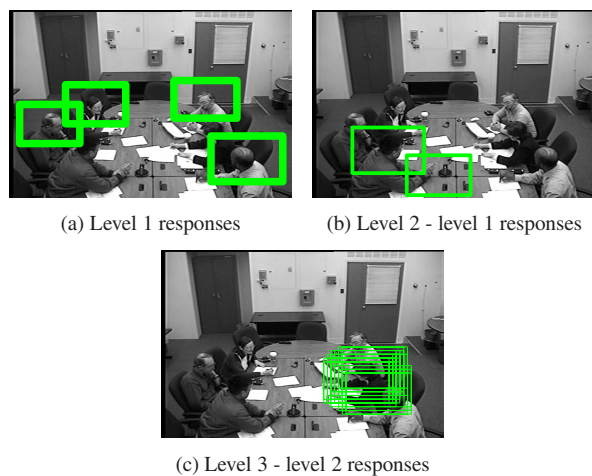


Figure 2. Examples of the three levels of detection responses for the problem of meeting room human detection and tracking.

## 2.2. HOG based SVM classifier

HOG descriptor, proposed by Dalal and Triggs [10], is another local shape feature. It encodes the statistics of the edge orientation within a small neighborhood. Following [10], we learn SVM classifiers for object classes based on HOG descriptors. We use the HOG based SVM classifier as post verification for the edgelet based boosted classifier. During training, the false alarms and the successful detected samples from the cascade boosted classifier are collected and used to learn the SVM classifier. During detection, first the cascade boosted classifier is applied to the whole image, then the positive responses are sent to the SVM classifier for verification. The threshold of the SVM classifier is chosen to remove 90% false alarms from the cascade boosted classifier and accept as many positive samples as possible. We use the SVM classifier as the verifier after the cascade

boosted classifier, because the boosted classifier is more efficient. The positive responses of the SVM classifier are the first level responses. Fig.2(c) shows an example of this level.

### 2.3. Discussion about Detection Module Design

Our detection module is one implementation of the “soft decision” idea based on existing detection techniques. There are other choices. For example, we could remove the thresholding process from the detection stage and use some continuous probability distribution representation as the soft decision, instead of the discrete confidence levels. However, this would greatly increase the problem space of the tracking algorithm. Searching for an unknown number of object trajectories in a real-valued spatial-temporal space directly is unfeasible. Also, thresholding is necessary for the cascade boosted classifier for efficiency. To achieve multiple confidence levels, we could learn multiple thresholds for one classifier, instead of combining multiple heterogeneous classifiers. However, for a single classifier the reasonable range of the threshold is likely to be small. Enforcing a high detection rate may result in a too high false alarm rate, and *vice versa*.

To achieve a low false alarm rate, we could reduce the final target false alarm rate of the cascade boosted classifier, instead of applying an SVM classifier for post verification. But learning a cascade boosted classifier includes a bootstrapping step for each layer of the cascade to collect new negative samples. When the current cascade is already operating at a very low false alarm rate, it is hard to collect enough negative samples. While boosting requires a large training set to get good generalization property, SVM can function well with smaller training sets.

The third level responses are mainly for a high detection rate. This level could be replaced with the responses of some early stage in the cascade boosted classifier. However, we find that at the same detection rate, the false alarm rate of the single-threshold boosted classifier is much smaller than that of the early-stage output of a cascade classifier, because the former one evaluates more features in making a decision.

The order in which the classifiers are applied is determined based on their speeds. Our experiments show that the speed ratio, between the cascade boosted classifier, the single-threshold boosted classifier, and the SVM classifier, is about 1:10:200. Note that the third level detection is applied *after* tracking based on the first two levels of detection. We use a second level response, in-between the first and the third levels, to reduce the search space of the third level for efficiency. The single-threshold boosted classifier is only applied around the predicted positions of the existing trajectories can not be tracked by associating with the detection responses of the first and the second levels.

## 3. Tracking Module

Our tracking algorithm has three main components: trajectory initialization, termination, and growth. In the three components we use the detection responses with different confidence levels in different ways based on their detection rates and false alarm rates.

### 3.1. Detection Response Association

Given a new frame from a video sequence, first the boosted classifier with the cascade decision strategy is applied and then the positive responses are sent to the SVM classifier to verify. This process provides the first and second level responses. One detection response is represented by a 4-tuple,  $\mathbf{r} = \{\mathbf{p}, s, \mathbf{c}, f\}$ , where  $\mathbf{p}$  is the image position,  $s$  is the size,  $\mathbf{c}$  is an appearance model, and  $f$  is a real-valued classification confidence. In our implementation,  $\mathbf{c}$  is a color histogram. The classification confidence  $f$  is the weighted sum of all weak classifiers’ outputs for the boosted classifier or the distance to the classification boundary for the SVM classifier. The object hypotheses have the same representation as the detection responses.

Similar to [3], we define the affinity between two detection responses,  $\mathbf{r}_1$  and  $\mathbf{r}_2$ , by

$$A(\mathbf{r}_1, \mathbf{r}_2) = A_{pos}(\mathbf{p}_1, \mathbf{p}_2)A_{size}(s_1, s_2)A_{appr}(\mathbf{c}_1, \mathbf{c}_2) \quad (1)$$

where  $A_{pos}$ ,  $A_{size}$ , and  $A_{appr}$  are affinity measure based on position, size, and appearance respectively. In practice,  $A_{pos}$  and  $A_{size}$  are modeled by Gaussian functions, and  $A_{appr}$  is modeled by Bhattachayya distance. Suppose that at the current frame we have  $n$  hypothesized trajectories,  $H_1, \dots, H_n$ , whose predictions at the current frame are  $\hat{\mathbf{r}}_1, \dots, \hat{\mathbf{r}}_n$ , and  $m$  first level detection responses  $\mathbf{r}_1, \dots, \mathbf{r}_m$ . We compute an  $n \times m$  affinity matrix  $\mathbf{A}$  of all  $(\hat{\mathbf{r}}_i, \mathbf{r}_j)$  pairs, *i.e.*  $\mathbf{A}(i, j) = A(\hat{\mathbf{r}}_i, \mathbf{r}_j)$ . If  $\mathbf{A}(i, j)$  is larger than a threshold, then we say  $\hat{\mathbf{r}}_i$  and  $\mathbf{r}_j$  is a potential association. The Hungarian algorithm [19] is used to find the best match between the hypotheses and the responses. After association with the first level responses, all remaining hypotheses are matched with the second level responses with the same algorithm.

After association with the first and second level responses, we apply the single-threshold boosted classifier around the predicted positions of the remaining unmatched hypotheses. This process gives the third level responses. The search space for this stage is much smaller than that for the first and second level responses. As the third level responses may include many false alarms, we add one more measure to the affinity function in Equ.1 to reduce the match ambiguity:

$$A'(\hat{\mathbf{r}}, \mathbf{r}) = A(\hat{\mathbf{r}}, \mathbf{r})A_{shape}(\mathbf{r}) \quad (2)$$

where  $A_{shape}(\mathbf{r}) = f$  is the classification confidence of  $\mathbf{r}$ . All hypotheses unmatched with the first and second level responses are matched with the third level responses.

If after this stage there are still unmatched hypotheses, we use a color based mean-shift tracker [17] to track them. However we find that this part is not critical for the whole system, as the detection rate of the third level responses is close to 100%. Even if we leave the finally unmatched hypotheses at their original positions, there is not a big difference in the performance.

### 3.2. Trajectory Initialization and Termination

The unmatched first level detection responses are used to initialize new trajectories. If at one frame, a first level detection response does not correspond to any existing trajectory, we start a new *potential trajectory*  $H$  from it. If in the succeeding  $T$  consecutive frames,  $T$  first level responses are matched with  $H$ , we compute an initialization confidence of  $H$  [6]. If the confidence is larger than a threshold, we say  $H$  becomes a *confident trajectory*. Because the false alarm rates of the second and third level responses are relatively high, the unmatched second and third level responses are not used to start new trajectories.

The trajectory termination criterion is similar to that of initialization. If in  $T$  consecutive frames we can not find the matched first or second level responses for one object hypothesis, then we compute a termination confidence. If the confidence is larger than a threshold, the trajectory is ended and we call it a *dead trajectory*, otherwise an *alive trajectory*. Our initialization and termination strategies are very similar to that of [6], but we make use of detection responses of multiple confidence levels.

The full detection and tracking algorithm is given in Fig.3. In practice, in order to compensate the initialization delay, after a trajectory becomes confident, we track the object in both forward and backward directions.

## 4. Experimental Results

We apply our approach to two applications: human tracking in indoor meeting scenarios and outdoor surveillance scenarios. We evaluate our system on two public video corpora and compare with previous methods quantitatively.

### 4.1. Evaluation Metrics

Recently, there have been many efforts towards quantitative evaluation of tracking algorithms. To compare with previous methods, we use four metrics defined by the CLEAR 2007 evaluation protocol [22, 21] for 2-D multiple object detection and tracking tasks:

1. Multiple Object Detection Accuracy (MODA) is the

detection accuracy calculated from the number of false alarms and missed detections;

2. Multiple Object Tracking Accuracy (MOTA) is the tracking accuracy calculated from the number of false alarms, missed detections, and identity switches.
3. Missed Detections Per Ground-Truth (MISS\_PGT) is the number of missed objects normalized by the number of ground-truth objects at the tracking level;
4. False Alarms Per Frame (FA\_PF) is the number of false alarms per frame at the tracking level.

To compute these metrics, at each frame the detected object hypotheses are matched with the ground-truth by the Hungarian algorithm [19]. Denote the 2-D regions of a pair of matched detection and ground-truth at the  $t$ -frame by  $D_i^{(t)}$  and  $G_i^{(t)}$  respectively, the overlap ratio of them is computed by

$$\text{Overlap}(D_i^{(t)}, G_i^{(t)}) = \frac{|D_i^{(t)} \cap G_i^{(t)}|}{|D_i^{(t)} \cup G_i^{(t)}|} \quad (3)$$

The ground-truth objects which do not match with any detection responses (the overlap ratio with any detection response is smaller than a given threshold) are counted as missed objects, the number of which is denoted by  $m_t$  for the  $t$ -th frame, and the detection responses which do not match with any ground-truth are counted as false alarms, the number of which is denoted by  $fp_t$ . The MODA score is computed by

$$\text{MODA} = 1 - \frac{\sum_{t=1}^{N_{frames}} (c_m(m_t) + c_f(fp_t))}{\sum_{t=1}^{N_{frames}} N_G^{(t)}} \quad (4)$$

where  $N_{frames}$  is the number of frames,  $N_G^{(t)}$  is the number of ground-truth objects at the  $t$ -th frame,  $c_m$  and  $c_f$  are penalties for missed detection and false alarm respectively. For tracking, identity switch is another type of error need to be considered. The MOTA score is computed by

$$\text{MOTA} = 1 - \frac{\sum_{t=1}^{N_{frames}} (c_m(m_t) + c_f(fp_t) + \log(id_{switches}))}{\sum_{t=1}^{N_{frames}} N_G^{(t)}} \quad (5)$$

where  $id_{switches}$  is the number of identity switches. The definitions of MISS\_PGT and FA\_PF are straightforward:

Among the four metrics, MOTA is the most important one for tracking systems. These may not be the ideal measures for evaluating tracking algorithms, however, they cover most typical errors and an automatic scoring software is available.

---

### Multi-Level Detection based Object Tracking Algorithm

Let the set of object hypotheses be  $F$ , initially  $F = \Phi$ .

For each time step  $t$  (denote by  $F_t$  the set of all alive trajectories in  $F$  at time  $t$ )

1. Apply the cascade boosted classifier and the SVM classifier to frame  $t$ . Let the sets of the first and second level responses be  $R_{t,1}$  and  $R_{t,2}$ .
2. Data association with the first and second level responses:
  - (a) Associate hypotheses in  $F_t$  with responses in  $R_{t,1}$ . Let the set of matched hypotheses be  $\tilde{F}_{t,1}$ .
  - (b) Associate hypotheses in  $F_t - \tilde{F}_{t,1}$  with responses in  $R_{t,2}$ . Let the set of matched hypotheses be  $\tilde{F}_{t,2}$ .
  - (c) Build a new hypothesis  $H$  from each unmatched response in  $R_{t,1}$ , and add  $H$  into  $F$  and  $F_t$ .
3. Apply the single-threshold boosted classifier around the predicted positions of all hypotheses in  $F_t - \tilde{F}_{t,1} - \tilde{F}_{t,2}$ . Let the set of the third level responses be  $R_{t,3}$ .
4. Data association with the third level responses:  
Associate hypotheses in  $F_t - \tilde{F}_{t,1} - \tilde{F}_{t,2}$  with responses in  $R_{t,3}$ . Let the set of matched hypotheses be  $\tilde{F}_{t,3}$ .
5. Pure tracking:  
For each confident trajectory in  $F_t - \tilde{F}_{t,1} - \tilde{F}_{t,2} - \tilde{F}_{t,3}$ , grow it by meanshift tracking.
6. Model update:
  - (a) For each potential trajectory in  $\tilde{F}_{t,1}$ , update its initialization confidence.
  - (b) For each hypothesis in  $\tilde{F}_{t,1} + \tilde{F}_{t,2}$ , update its appearance model.
  - (c) For each trajectory in  $\tilde{F}_{t,1} + \tilde{F}_{t,2}$ , reset its termination confidence to 0.
  - (d) For each trajectory in  $F_t - \tilde{F}_{t,1} - \tilde{F}_{t,2}$ , update its termination confidence.

Output all confident trajectories in  $F$  as the final results.

---

Figure 3. Multi-level detection based object tracking algorithm.

## 4.2. Human Tracking in Meeting Videos

In meeting videos, only the upper-bodies of the humans are visible most of the time while the legs could be occluded by some scene objects, *e.g.* table. Hence we learn a head-shoulder detector and track the head-shoulder parts for this problem. (The definition of human body parts is same as that in [3].) The training/testing data are from the NIST meeting corpus [20]. The training set contains about 2,500 positive samples for frontal/rear view, 3,000 positive samples for profile view, and 650 background images of indoor scene without humans. The positive samples are normalized to  $36 \times 24$  pixels. The test set contains 50 sequences, overall about 204,000 frames, with a frame rate of 30 FPS and a frame size of  $720 \times 480$  pixels. The sizes of humans vary from 120 pixel wide to 300 pixel wide. The training set and test set are captured in the same meeting room with the same camera setting but different sitting arrangements and different attendants. This set was used in the VACE 2005 evaluation [21].

First, we evaluate the detection rates and the false alarm rates of the three level detection responses. 200 frames are randomly selected from the test videos and sent to our detection module. Note, to evaluate the actual detection performance, we apply the single-threshold boosted classifier to the whole image. Table.1 shows the scores of the three

levels. It can be seen that these three levels represent different favors in the decision tradeoff.

	Level 1	Level 2	Level 3
Detection rate (%)	75.65	88.19	99.95
False alarm (# per frame)	0.58	2.53	45.20

Table 1. Detection performance of the three level detection responses.

Second, we compare the end-to-end performance of our method with the system in [6], which is also a detection based tracking algorithm. However, the detection module in [6] outputs only one level of decision, whose accuracy is similar to our second level responses. Table.2 lists the scores of these two methods. It can be seen that our method achieves about 7.3% higher MOTA than the method in [6] and our system produces both fewer false alarms and fewer missed detections. Fig.4(a) shows an example result.

	MODA	MOTA	MISS_PGT	FA_PF
System in [6]	0.7142	0.7139	0.1680	0.2334
This method	0.7815	0.7870	0.1011	0.1978

Table 2. System performance of human detection and tracking in meetings.

### 4.3. Human Tracking in Surveillance Videos

For street surveillance scenarios, we learned full-body detectors for pedestrians. The training/testing data are from the CLEAR-VACE surveillance corpus [22]. The training set contains about 1,700 positive samples for frontal/rear view, 1,120 positive samples for profile view, and 500 background images of outdoor street scene without humans. The positive samples are normalized to  $24 \times 58$  pixels. The testing set contains 50 sequences, overall about 121,000 frames, with a frame rate of 30 FPS and a frame size of  $720 \times 480$  pixels. This set was used in the CLEAR 2006 and 2007 evaluations [22]. The sizes of humans vary from 10 pixel wide to 80 pixel wide. However, as our detectors do not work on very low resolution, we modify the original ground-truth to label the humans smaller than  $20 \times 50$  pixels as “don’t care”.

We compare the performance of our method with the system in [9] which is a detection based tracking algorithm and has only one level of detection responses. Table.3 lists the scores of these two systems. It can be seen that our method achieves about 8.8% higher MOTA than the method in [9]. Note, the scores in Table.3 are computed by excluding very small humans, hence they are not directly comparable with the scores reported in the CLEAR evaluation workshops [22], which consider humans of all sizes. Fig.4(b) shows an example result.

	MODA	MOTA	MISS_PGT	FA_PF
System in [9]	0.5265	0.5252	0.4231	0.0381
This method	0.6157	0.6137	0.3147	0.0697

Table 3. System performance of human detection and tracking in surveillance videos.

The speed of the entire system is about 1 FPS for the meeting room human tracking problem and 0.4 FPS for the surveillance human tracking problem. The most computational part of our system is the detection module. The speed of our three level detection module is similar to that of the single level detection module [6, 9], because the expensive detectors of the second and third levels are only applied to a small search space. Our experimental machine is a 64-bit Intel Xeon 3.6GHz CPU PC with 4G RAM; the program is code in C++ using OpenCV functions without any parallel computing.

### 5. Conclusion and Future Work

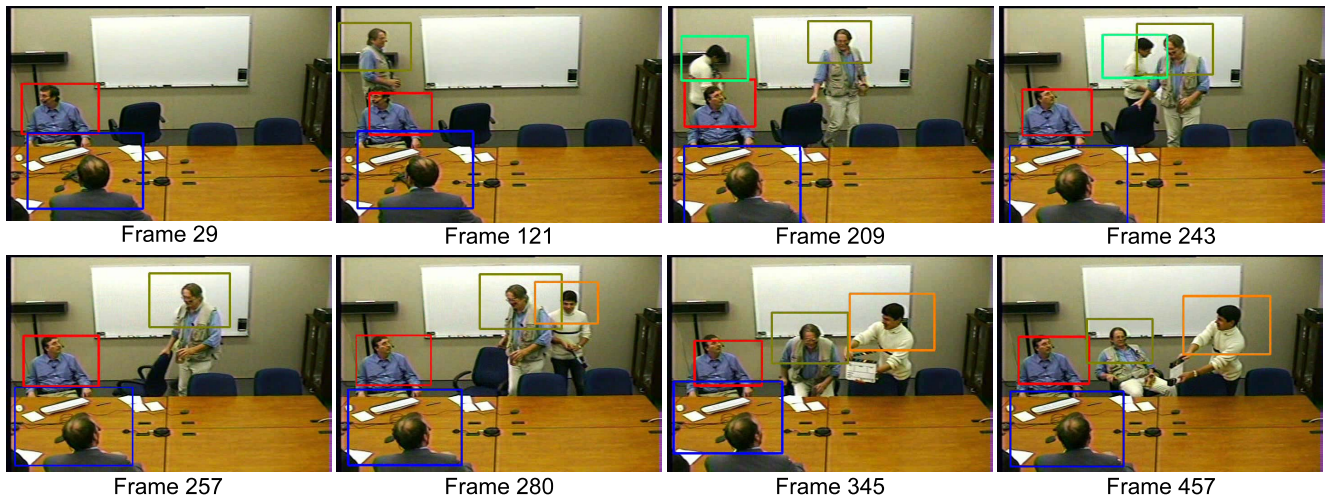
We described a fully automatic object detection and tracking system. Tracking is done by associating detection responses of multiple confidence levels. Experimental results show that our method is more robust than the tracking algorithms based on detection with only a single decision strategy.

The affinity measure is one of the key parts in our system. The current affinity only encodes spatial and color information. In our future work, we plan to integrate other types of information, such as the motion consistency based on the Lucas-Kanade feature tracker [18].

**Acknowledgements:** This research was funded, in part, by the U.S. Government VACE program.

### References

- [1] Y. Li, H. Ai, T. Yamashita, S. Lao, and M. Kawade. Tracking in Low Frame Rate Video: A Cascade Particle Filter with Discriminative Observers of Different Lifespans. CVPR 2007. 2
- [2] O. Tuzel, F. Porikli, and P. Meer. Human Detection via Classification on Riemannian Manifolds. CVPR 2007. 1
- [3] B. Wu, and R. Nevatia. Detection and Tracking of Multiple, Partially Occluded Humans by Bayesian Combination of Edgelet based Part Detectors. International Journal of Computer Vision, 2007. 1, 2, 3, 4, 6
- [4] C. Huang, H. Ai, Y. Li, and S. Lao. High Performance Rotation Invariant Multiview Face Detection, IEEE Transactions on PAMI, 29(4): 671-686, 2007. 1, 3
- [5] Q. Zhu, S. Avidan, M.-C. Yeh, and K.-T. Cheng. Fast Human Detection Using a Cascade of Histograms of Oriented Gradients. CVPR 2006. 1
- [6] B. Wu, and R. Nevatia. Tracking of Multiple Humans in Meetings. In V4HCI workshop, in conjunction with CVPR 2006. 5, 6, 7
- [7] A. Opelt, A. Pinz, and A. Zisserman. A Boundary-Fragment-Model for Object Detection. ECCV 2006. 1
- [8] A. Kapoor, and J. Winn. Located Hidden Random Fields: Learning Discriminative Parts for Object Detection. ECCV 2006. 1
- [9] B. Wu, X. Song, V. K. Singh, and R. Nevatia. Evaluation of USC Human Tracking System for Surveillance Videos. In CLEAR Evaluation Campaign and Workshop, in conjunction with FG 2006. 7
- [10] N. Dalal, and B. Triggs. Histograms of Oriented Gradients for Human Detection. CVPR 2005. 1, 2, 3
- [11] L. Zhao, and L. S. Davis. Closely Coupled Object Detection and Segmentation. ICCV 2005. 1
- [12] K. Okuma, A. Taleghani, N. D. Freitas, J. J. Little, and D. G. Lowe. A Boosted Particle Filter: Multitarget Detection and Tracking. ECCV 2004. 2
- [13] B. Leibe, A. Leonardis, and B. Schiele. Combined Object Categorization and Segmentation with an Implicit Shape Model. Workshop on Statistical Learning in Computer Vision, in conjunction with ECCV 2004. 1
- [14] T. Zhao, and R. Nevatia. Tracking Multiple Humans in Complex Situations, IEEE trans. on PAMI, 26(9): 1208-1221, 2004. 1
- [15] P. Viola, M. J. Jones, and D. Snow. Detecting Pedestrians Using Patterns of Motion and Appearance. ICCV 2003. 1
- [16] P. Viola, and M. Jones. Rapid Object Detection using a Boosted Cascade of Simple Features. CVPR 2001. 1, 3



(a) Example result of human tracking in meeting videos



(b) Example result of human tracking in surveillance videos

Figure 4. Example results of object tracking.

- [17] D. Comaniciu, V. Ramesh, and P. Meer. The Variable Bandwidth Mean Shift and Data-Driven Scale Selection. *ICCV* 2001. 1, 5
- [18] C. Tomasi, and T. Kanade. Detection and Tracking of Point Features. Technical Report CMU-CS-91-132, Carnegie Mellon University, 1991. 1, 7
- [19] H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2: 83-87, 1955. 2, 4, 5
- [20] J. Garofolo, C. Laprum, M. Michel, V. Stanford, and E. Tabassi. The NIST Meeting Room Pilot Corpus. In: *Proc. of Language Resource and Evaluation Conference*. 2004. 6
- [21] R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, M. Boonstra, and V. Korzhova. Performance Evaluation Protocol for Face, Person and Vehicle Detection & Tracking in Video Analysis and Content Extraction (VACE-II) CLEAR - Classification of Events, Activities and Relationships. 5, 6

- [22] <http://www.clear-evaluation.org/> 5, 7