What are they doing? : Collective Activity Classification Using Spatio-Temporal Relationship Among People.

Wongun Choi
University of Michigan
Ann Arbor, USA
wgchoi@umich.edu

Khuram Shahid
University of Michigan
Ann Arbor, USA
kshahid@umich.edu

Silvio Savarese
University of Michigan
Ann Arbor, USA
silvio@eecs.umich.edu

Abstract

In this paper we present a new framework for pedestrian action categorization. Our method enables the classification of actions whose semantic can be only analyzed by looking at the collective behavior of pedestrians in the scene. Examples of these actions are waiting by a street intersection versus standing in a queue. To that end, we exploit the spatial distribution of pedestrians in the scene as well as their pose and motion for achieving robust action classification. Our proposed solution employs extended Kalman filtering for tracking of detected pedestrians in 2D 1/2 scene coordinates as well as camera parameter and horizon estimation for tracker filtering and stabilization. We present a local spatio-temporal descriptor effective in capturing the spatial distribution of pedestrians over time as well as their pose. This descriptor captures pedestrian activity while requiring no high level scene understanding. Our work is tested against highly challenging real world pedestrian video sequences captured by low resolution hand held cameras. Experimental results on a 5-class action dataset indicate that our solution: i) is effective in classifying collective pedestrian activities; ii) is tolerant to challenging real world conditions such as variation in illumination, scale, viewpoint as well as partial occlusion and background motion; iii) outperforms state-of-the art action classification techniques.

1. Introduction

Consider a video sequence capturing a number of individuals located in an indoor environment such as coffee shop. Imagine an algorithm that is able to process the video and answer questions such as: Are these people talking? Are they in a queue waiting to order food or drink? By just looking at each single person it may be challenging to design an algorithm that is able to address these questions (Fig.1). In this paper we introduce a new paradigm for recognizing human actions: rather than classifying individuals in isolation, we analyze their collective behavior so as to reinforce the recognition of each individual's actions. This paradigm is inspired by recent contributions in computer vision where semantic or geometrical contextual information is used to help recognize objects in complex scenes [14]. In this work, action classification is enhanced by taking advantage of contextual information that comes from the position, pose and the actions of multiple individuals in the surrounding area. Unlike many previous methods of human action recognition, we aim at working under unrestrictive conditions such as dynamic cluttered background, variations in illumination and viewpoint, intra class variability in the human appearance and non-static cameras.

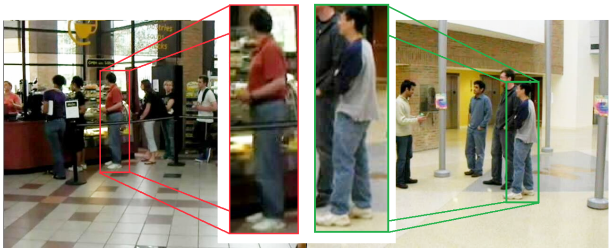


Figure 1. Example of queueing (left) and talking (right) actions. By just looking at one individual, it is very hard to classify whether this person is in a queue or talking. However, by looking at what the surrounding people are doing, the actions can be disambiguated. We aim to solve this problem by capturing videos using unstabilized cameras under generic viewing conditions.

Our algorithm is built upon the robust detection of humans using deformable part based detector[11] and HOG descriptor [7] for classifying human poses. We introduce a new algorithm based on the Extended Kalman filter that enables robust tracking of each detected human for a number of frames. Our algorithm incorporates into the feedback loop the estimation of rough camera parameters, the scene horizon line and 2D 1/2 location of each tracked individual. This makes the recovery of the each person's trajectory in parameter space robust with respect to camera shaking, viewpoint, scale changes, and background motion. Action recognition is eventually performed by introducing a new descriptor that captures the spatial temporal distribution of the surrounding people (position, pose and movement) using a binning scheme similar to the shape context descriptor [2]. Such descriptors enable the classification of the action performed by each detected person by using the dynamic behavior (location, pose and movement) of the surrounding people.

We have demonstrated the strength of our algorithm on a number of video sequences portraying 5 different human action categories: walking, crossing, waiting, queueing and talking. Our video sequences were acquired by using home consumer hand held un-stabilized cameras capturing images of human activities in cluttered and busy environments. Experimental results show that our algorithm can successfully recognize different action categories in such challenging conditions. Also, our analysis demonstrates that the added contextual information provided by the collective behavior of individuals is critical for the coherent understanding of complex scenes. We see our work as a promising starting point for a number of applications such as autonomous vehicles, video surveillance, topic level video summarization and assistive technologies for impaired users.

2. Related Works

The classification of human actions in video sequences has received a large amount of interest in the computer vision community. This challenging problem is closely related to that of human detection and tracking. In past years, a number of methods have been proposed for accurately detecting [7] [26] [27] [29], tracking [23] [3] [17] and estimating the pose [12] [24] of multiple humans in cluttered environments. Since this problem is highly challenging in its most general formulation (moving cameras, dynamic background, etc..), researchers have proposed to make the tracking process more stable by combining detection and tracking [29] or using additional information such as stereoscopy [9] and motion cues [27]. In this work, we follow a similar philosophy. However, unlike [9] [30] where the knowledge of extrinsic (location, pose) and intrinsic (focal length) camera parameters play a critical role in detecting stable tracks, but similar to [14], we assume a simple camera model (camera height) and scene model (horizon line). Thus, camera and scene parameters can be estimated during the tracking process and, in turn, these parameters can be used to make the human detection more stable, without the need for solving the full 3D reconstruction problem.

Stable detection and robust tracking of humans in the scene is a critical building block if one wants to design robust algorithm for human action classification. The computer vision literature boasts a large body of works which can be coarsely divided in two groups depending on whether the goal is to recognize a simple action or a complex one. Researchers often refer to the latter case as the problem of recognizing and understanding human activities. A recent survey by [25] presents an excellent summary of recent and past methods on action/activity recognition. Among these, of particular interest are those based on volumetric and contour based representations[4][31], spatial-temporal filtering[32], distributions of parts [8][10][21][22] sub-volume matching[15] and tensor-based representations [16]. Different methods make different assumptions regarding the camera, background, and number of people in the scene. For instance, part based methods [8] [10] [21] [22] are flexible in modeling self-occlusions and articulated movements, but can hardly cope with crowded scenes.

One common assumption in human behavior classification methods is that actions are recognized in isolation - that is, the behavior of a human in the scene is recognized independently of what other humans are doing in the same scene. Our work introduces a new paradigm where actions are recognized in relationship with what other humans are doing in the scene. Unlike [13], our goal is not the one of classifying complex activities (e.g. sequence of actions involving one or multiple interacting humans), but rather disambiguating atomic actions (*e.g.*, queueing before a shop or waiting by a traffic intersection) which can hardly be characterized without analyzing the collective behavior of individuals. Also, unlike many of the works involving activity recognition we remove the assumption of static camera [4] [5]. Finally, our work is similar in spirit to [20]; however we focus more on the supervised multi-class activity classification rather than anomaly detection.

3. Method

3.1. System Overview

This section gives a brief overview of our method. More details will be discussed in subsequent sections. The basic idea is that instead of considering only a single individual's behavior, some human activities can be better inferred by considering location and motion of nearby individuals collectively. To show this, we begin by detecting humans in image sequences as well as classifying their poses [Sec. 3.2]. Given a group of detected individuals, the camera parameters, camera height and horizon position are estimated using a probabilistic model [Sec. 3.3]. Using these estimated camera parameters together with detection results, Extended Kalman Filtering(EKF) is applied onto each individual in order to estimate their 2D 1/2 trajectory [Sec. 3.4]. A spatio-temporal descriptor is constructed using these tracking results [Sec. 3.5] and employed for the ensuing classification stage where activities of individuals

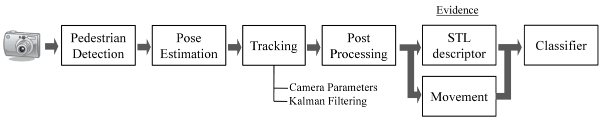


Figure 2. Overall process of our system.

are recognized [Sec. 3.6].

3.2. Human Detection and Pose Classification

The detection stage of our method employs the multiscale deformable part based detector developed by Felzenszwalb *et al.* [11]. This detector uses a Histogram of Oriented Gradients(HOG)[7] descriptor to learn a part based model for humans. Given a fixed number of parts to learn, the algorithm identifies the most common yet descriptive HOG features across the training set and learns these features as well as their positions relative to the bounding box provided in training. During detection, if the cost of the deformation necessary to make a candidate resemble the learned model is lower than a threshold, the system declares the candidate to be a positive match for a human. As a result, the detector is able to handle partial occlusions while still providing an accurate bounding box due to learned knowledge of relative position of parts.

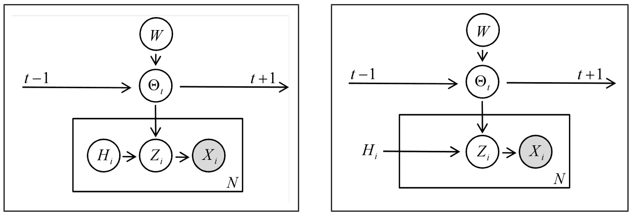
As far as the pose estimation is concerned, we use the HoG descriptor [7] and a linear SVM classifier in order to classify different poses : front, left, right, and back. Final pose classification is achieved using a 1 vs. all classification regime on the bounding boxes obtained from the detector.

3.3. Camera Parameter Estimation

As previous works show, in order to obtain stable and robust tracking results, it is desirable to estimate the camera parameters and the 3D location of the target. However, in cluttered and busy environment scenes captured by hand held cameras, 3D scene reconstruction using structure from motion (SFM) is problematic in that : i) the reconstruction is noisy and unreliable due to small base-line changes, ii) dynamic scene elements violate the SFM assumption of static background, iii) the procedure is computationally expensive and can be hardly implemented in real time. Also unlike [9], we do not want to take advantage of binocular systems. Thus, our challenge is to estimate a coarse 3D location (2D 1/2, x and depth information) of target along with camera parameter for robust tracking without using either stereo system or SFM. Inspired by Hoiem *et al.* [14], we propose to highly simplify the "SFM" problem by assuming that all people are standing on a flat ground plane and camera tilt is approximately zero. Under these assumptions, the relationship between the position of the feet, height of a person, position of the horizon in the image plane and the camera height can be expressed as follows

$$h = y_c \frac{s_i}{v_i - v_0} \tag{1}$$

where h is the height of person (meters), (v_i,s_i) is bottom position and vertical size of detection bounding box in image plane (pixels), and (y_c, v_0) is camera height and horizon position in image plane (meters and pixels).



(a) (b)

Figure 3. Θ_t is the camera parameters in time t, H_i is actual height of each person, Z_i is the true position and height in image plane, and X_i is measurement of their state in image plane. In this diagram, only the measurement X_i is observable whereas all the others are hidden variables. (a) is the exact model, and (b) is a simplified model.

Given this equation, we can design a probabilistic model for each person in the image plane. This model can be explained by an analogy to the generative model. Once the camera parameters $\theta_t = \{y_c, v_0\}$, and height h_i of each person are sampled from the prior distributions, a set of points Z_i in image plane space can be generated, that follow a linear constraint satisfying Eq.1. After sampling a point Z_i = (v_i, s_i) randomly from given set, a measurement X_i will be obtained by applying additive gaussian noise onto true state values. The variance of this noise is set relative to the height of people in the image plane. Since the relationship between each variable is known, we estimate the camera parameters given X_i in image plane. Similar assumptions to [14] were used for the initial prior distribution of camera parameters. In our implementation, the posterior distribution for the camera parameters in time t was used as the prior distribution for the next time t +1. However, Gaussian noise is added to the posterior before it is used as the prior to model the noise added due to camera shaking.

The equation for this model is as below

$$\begin{aligned} P(\theta|\bar{X}) &\propto P(\bar{X}|\theta)P(\theta) \qquad (2) \\ &= \sum_{\bar{h}} P(\bar{X}|\bar{h},\theta)P(\bar{h})P(\theta) \qquad (3) \end{aligned}$$

$$= \sum_{\bar{h}} \prod_i P(X_i|h_i,\theta)P(h_i)P(\theta) \qquad (4)$$

$$\approx \prod_i P(X_i|\tilde{h},\theta)P(\theta) \qquad (5)$$

Marginalization over hidden variable h_i requires extensive amount of computation. We further simplified the model by assuming all h_i are just parameters set to the mean height of people ($\tilde{h}$ =1.7 m). Then the generative model reduces to the fig 3.(b). Though it seems to be an oversimplification, this method worked well in practice. In order to compute $P(X_i|\tilde{h},\theta)$, instead of marginalizing out z every time, we compute $P(X_i|\tilde{h},\theta)$ using a sampling method, prior to camera parameter estimation. For every possible value of (y_c, v_o), we randomly sampled a point Z from the line satisfying equation 1, and sampled X_i from Z and added gaussian noise. We iterated 500,000 times and generated an approximated probability. Finally, given $P(\theta|\bar{X})$ we use a maximimum likelihood estimate for the camera parameters that are to be used in Kalman Filtering.

3.4. Filtering and Matching

To make the STL descriptor robust to viewpoint, we propose to construct the STL descriptor in 2D 1/2 coordinates by first estimating the position of the individuals in the scene [Sec.3.5].

Since the projection onto the image plane is inherently a nonlinear function, We use a first order Extended Kalman Filter (EKF) in our system. The state models the position of each person's feet (x, z), their velocity (v_x, v_z) measured based on the feet position, as well as their height h. Though the height of each individual might seem irrelevant to the tracking task, this additional variable will help the Kalman filtering in providing a more stable estimate given a noisy set of bounding boxes from the detector. The actual height of a person is closely related not only to the vertical size of a bounding box but also to the depth z in the actual world. Thus, if we assume that actual height is constant over time, we are able to put a constraint on the possible location of an individual's feet which is especially helpful if the detection results are poor. As the Kalman filtering measurement, we simply use the feet position of each person (u, v) and the vertical size s of the bounding box in the image plane, both of which are obtained from the detector. Our state, X, and measurement, Z variables then become : X=(x,z,h,v_x,v_z) and Z=(u,v,s).

We assumed that the movement of each person in 2D 1/2 space is locally linear, thus the transition matrix A will have the following simple expression :

$$X_{k+1} = AX_k + W_k \quad A = \begin{bmatrix} 1 & 0 & 0 & dt & 0 \\ 0 & 1 & 0 & 0 & dt \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad (6)$$

where X_k is the state in time k and W_k is the process noise, and A is the transition matrix.

The measurement vector Z_k will be related to the state vector via the measurement function $Z_k = h(X_k + V_k)$ where V_k is a measurement noise. Under the same assumption we made in section 3.3 the measurement function can be defined as : $h(X;\tilde{\theta}) = [\frac{fx_i}{z} + u_c, \frac{fy_c}{z} + v_0, \frac{fh}{z}]^T$. Here, $\tilde{\theta}$ represents the camera parameters (f, y_c, u_c, v_0), f is the focal length, y_c is the height of the camera, u_c is the horizontal center of the image plane, and v_0 is the position of horizon in image plane. Given these equations, we were able to implement the actual EKF for each person. For more details on EKF see [28].

The Kalman filter is essentially applicable onto only a single moving object, which is clearly not ideal for our datasets which contain multiple people. To address this issue, we proposed to find the correspondence between detection results between frames. We used two different types of cues in order to find the matches between frames : appearance and dynamics. Given observations at t, EKF will produce a predicted location of the tracked object in $t + 1$. Also by computing color histograms for each of the detected bounding boxes and averaging them over [0 to t], we can find the best match between tracked objects and new detections. We assumed these two cues are independent. Since the detector has a high false-negative rate, and it is often the case that people suddenly enter the field of view of the camera, we use an empirical threshold to reject false matching. Fig.6 shows the benefit of using EKF in 2D 1/2 coordinates. Fig.5 reports performance of our tracker on the [9] dataset. Notice that unlike [9], we did not use explicit 3D information (from a stereo system) to help boost tracking accuracy.

3.5. Spatio-Temporal Descriptor

Our spatio-temporal local (STL) descriptor is inspired by the popular Shape Context developed by Belongie *et al.* [2]. Our histogram-based STL descriptor is centered on an individual(the anchor), and captures the histograms of number of people, and their pose, in different bins surrounding the anchor. Each descriptor is oriented according to the anchor's pose (Fig.4). Such histograms, calculated per frame, are concatenated to capture the temporal evolution of the activities being performed in the scene - this forms an STL descriptor. As there may be numerous individuals present in a single scene, we construct STL descriptors, centered around

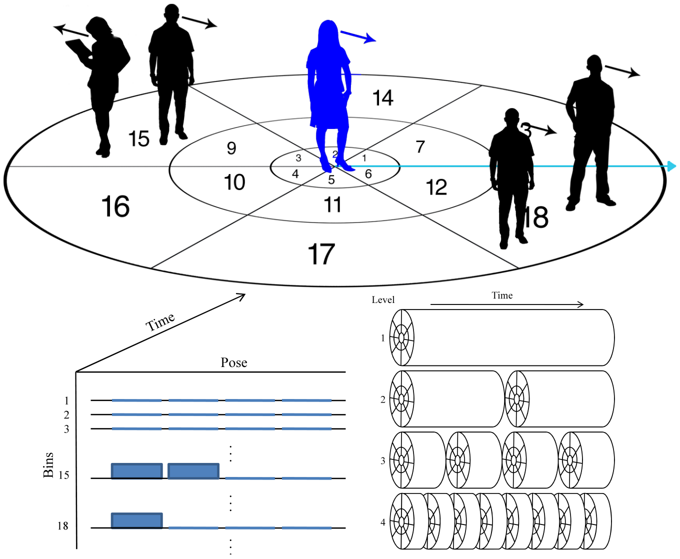


Figure 4. Spatio-Temporal Local Descriptor. (a) Space around anchor person (blue) is divided into multiple bins. The pose of the anchor person (blue arrow) locks the "orientation" of the descriptor which induces the location of the reference bin "1". (b) Example of STL decriptor - the descriptor is a histogram capturing people and pose distribution in space and time around the anchor person. (c) Classification of STL descriptor is achieved by decomposing the histogram in different levels along the temporal axis.

each person in the scene. Ultimately, we gather a collection of STL descriptors, one per individual being tracked.

Since the STL descriptor captures spatial variation over time, the relative motion of each human in the scene is implicitly embedded in the descriptor. Furthermore, since tracking is performed in 2D 1/2 scene coordinates, we are able to apply the STL descriptor to the bird's eye view of the scene. This helps the descriptor to be robust to perspective as well as view-point changes and to implicitly capture the motion and velocity of each individual with respect to that of the anchor in 2D 1/2 scene coordinate.

3.6. Classification

Our system classifies each person in the video sequence at every N = 10 frames by choosing the class that best explains the evidence (observation) arsing from STL descriptors as well as the velocity of an anchor person. We assume that these observations are independent, thus the classification step can be expressed as follows,

$$\hat{C} = \arg\max_{C} P(C|e_S, e_V) \tag{7}$$

$$\begin{aligned} P(C|e_S, e_V) &\propto P(e_S, e_V|C) && (8)\\ &= P(e_S|C)P(e_V|C) && (9)\end{aligned}$$

where e_S , e_V indicate the evidence brought by the STL descriptor and velocity descriptor respectively, and C indicates an activity class.

Evidence arising from STL descriptors is obtained as follows: descriptors are constructed for each tracker at time t for an empirically chosen fixed length duration $T = [t-31, t+32]$ (roughly 2 seconds $\approx$ 64 frames). Once the descriptor is built, libSVM toolbox [6] and a pyramid-like kernel[19][18] are used to classify each descriptor. Instead of dividing feature space or spatial coordinates into levels, we iteratively divided the temporal axis so as to obtain 4 levels in total (see Fig.4). The pyramid matching kernel is very useful in our framework, since it can capture various degrees of information about the distribution of people around each anchor person - the correlation across density distributions at the lower level; the relationship between people movements at the higher level. The likelihood $P(e_S|C)$ was provided by libSVM[6].

Since STL descriptor cannot capture the movement information of an anchor person, we considered evidence arising from per-person velocity. Average velocity (magnitude and direction aligned along the pose) of each person in each time segment (T) was estimated and discretized using a one-out-of-K coding scheme ($K = m * n$, with m bins in magnitude and n bins in angle). $P(e_V|C)$ is estimated by counting the occurence of such encodings in each activity class.

Each segment of frames is classified independently. However, additional temporal regularization can help the classifier to be more robust. We employed a Markov Chain so as to enforce temporal constraints between the same person's activities in different time segments (Eq.10). The transition probability $P(C_t|C_{t-1})$ was estimated by counting the occurences of each activity transition in each video of the training set.

$$P(C_t|e_S, e_V, C_{t-1}) \propto P(e_S, e_V|C_t)P(C_t|C_{t-1})P(C_{t-1}) \tag{10}$$

4. Experimental Results

4.1. Dataset

Our goal is to classify human activities based on collective behavior of individuals under general conditions. Since there is no existing dataset that can be used for evaluating our framework, we created our own dataset [1]. Unlike many existing datasets, our dataset is acquired under unconstrained real-world conditions. Over 40 short video clips of crossing, waiting, queueing, walking and talking action categories were recorded. The videos are 640x480 pixels in size and were recorded using a consumer hand held camera. See fig.9 to gain understanding of the complexity of the scenes. Every tenth frame of all video sequence was manually labeled with pose, activity and bounding box information. Only the pose label is required for learning purposes. The remaining labels are used for performance evaluation and dataset characterisation. Table.1 helps understand some of the properties of our dataset. Every property was estimated using manually annotated ground truth data.

Property	Crossing	Waiting	Queueing	Walking	Talking	Overall
Number of People	3.89	4.19	7.32	2.57	3.86	5.22
Number of Classes	1.42	1.39	1.15	1.46	1.49	1.37
Activity Clutter	0.35	0.31	0.14	0.41	0.49	0.33
Bounding Box Overlap	0.24	0.48	0.43	0.16	0.38	0.34
Camera Shake	18.55	13.76	23.30	19.53	12.88	18.35

Table 1. Dataset Characteristic : *Number of People* indicates the average number of people per frame performing a certain activity. *Number of Classes* indicates the average number of different activities in a short video sequence and hence represents the activity contamination of our dataset. *Activity Clutter* conveys the average number of people whose activity was too ambiguous. *Bounding Box Overlap (%)* indicates the average amount of overlap in each bounding box - 0 means completely visible and 1 means not visible - and conveys scene density along with partial occlusion. Finally the *Camera Shake* (in pixels) was estimated by applying our camera parameter estimation algorithm on ground truth labeled data and computing the mean squared difference between horizon positions in consecutive frames. This indicates the average amount of camera shaking within each action category.

4.2. Detection, Tracking and Camera Parameter Estimation

In order to test our tracking results we used the dataset proposed in [9]. Note that even though we used a video sequence from a single camera alone, the recall rate (table.2 and fig.5) was comparable to that of the algorithm in [9] where a full stereo system is used.

system	Seq 1	Seq 2	Seq 3
Ess [9]	0.44	0.43	0.44
Our system	0.50	0.34	0.31

Table 2. Recall at 0.5 FPPI. The columns show Recall performance for 3 sequences in [9]

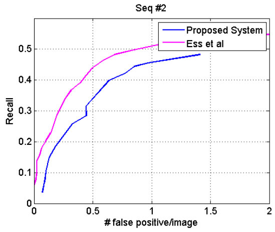
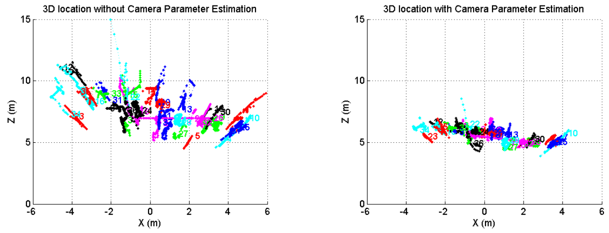


Figure 5. Comparison of our tracking results with those obtained by [9] on the dataset provided by [9]. Our algorithm only uses a single camera instead of the stereo system as in [9].

We also evaluated to what degree our tracking algorithm is robust in presence of significant camera shakes. The overall standard deviation of 2D 1/2 people location estimation computed without camera parameter estimation was $((\sigma_X, \sigma_Z) = (1.13, 2.84)$, where σ_X is an average standard deviation in x coordinate and σ_Z is an average standard deviation in z coordinate). These values were computed on static activities videos (*e.g.* waiting, queueing and talking). After parameter estimation, however, the location estimation shows significantly less variation $((\sigma_X, \sigma_Z) = (0.15, 0.66))$. A qualitative example of such improvement is reported in fig.6. This stabilization helps improve the accuracy of subsequent classification steps.



(a)

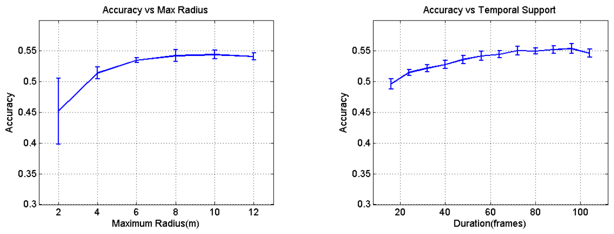
(b)

Figure 6. Examples of 2D 1/2 trajectory estimation for the queueing (static) action. Ideally trajectories should nearly collapse to one point. Instead, because of camera shakes, the trajectories become much noisier. By taking advantage of camera parameter estimation, the computed trajectories (right) are more stable than those obtained without camera estimation (left).

4.3. Activity Classification

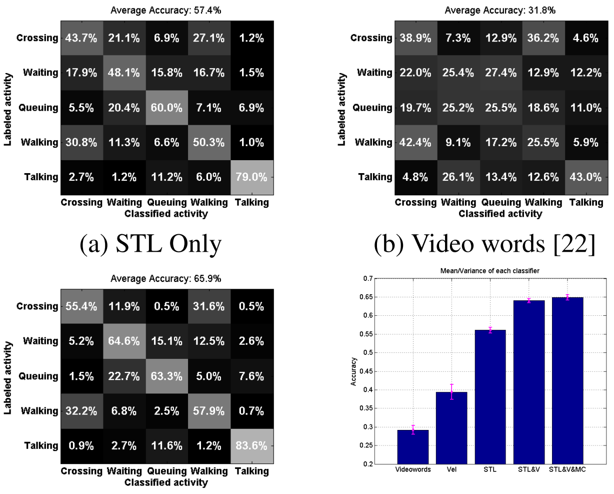
A leave-one-out scheme was used to assess the performance of our system. When classifying one video, we removed this video from our training set, and sampled a uniform random subset of the other videos to serve as the training set. The amount and relevance of the data collected in the STL descriptor is controlled by the spatial and temporal parameters. As the radial support distance increases, the STL is able to capture the dynamic property of more individuals in the scene and can hence provide richer information regarding the activity of the anchor (Fig7). Similarly, a longer temporal support helps disambiguate activities that might share similar micro-motions and supress noise. However, this does increases the risk of classification failure if there is a activity transition (Fig7).

As a bechmark, we also report a classification result achieved by the popular video codewords method [22]. Histograms of video codewords are extracted as described in [22] and they are classified by using a SVM classifier equipped with a kernel based on histogram intersections. As shown in Fig.8.(b), due to camera-shakes, clutters and intrinsic class ambiguities, it is hard to perform classification just based on video codewords alone. On the contrary, STL descriptor and combination of STL and velocity descriptors



(a) (b)

Figure 7. Impact of changing STL parameters. a) Any individuals that are farther than a set distance from the anchor are ignored. This radius defines the spatial support distance of the STL descriptor. b) The temporal support correlates with the number of frames used to construct the STL and hence impacts the activity classification. Note: Only STL descriptor is used for classification in this stage.



(a) STL Only (b) Video words [22]

(c) Full implementation (d) Mean/Var

Figure 8. Confusion tables for different classification method. Our method based on STL descriptor (a) outperforms method based on video words (b). (c) shows results by using combination of STL, velocity descriptor and a Markov Chain. (d) indicates that the STL descriptor is reliable and shows only a small variation in classification accuracy. Notice, the plot indicates clear improvement over activity classification using video-words alone. Notice that actions such as walking and crossing or queueing and talking can be successfully discriminated using the information captured by the STL descriptors.

provided promising results. As suggested in the introduction, actions such as walking vs crossing or queueing vs talking, can not be disambiguated by looking at individual person only. As Fig.8 (a) and (c) show, the contribution of STL descriptor help disambiguate these actions.

5. Conclusion

In this paper, we demonstrated that collective correlation among multiple people can help to improve classification of human activities by using STL descriptor as well as other cues. We anticipate a significant performance improvement if a better human pose classifier and additional scene level semantic cues are provided.

References

[1] Activity recognition dataset, 2009. Dataset will be available at http://www.eecs.umich.edu/vision/activity-dataset.html.

[2] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, 24(4):509–522, 02.

[3] J. Berclaz, F. Fleuret, and P. Fua. Robust people tracking with global trajectory optimization. In *CVPR*, volume 1, June 2006.

[4] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *ICCV*, volume 2, Oct. 2005.

[5] A. Bobick and J. Davis. The recognition of human movement using temporal templates. *PAMI*, 23(3):257–267, Mar 2001.

[6] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[7] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, volume 1, June 05.

[8] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on*, pages 65–72, Oct. 2005.

[9] A. Ess, B. Leibe, K. Schindler, and L. Van Gool. A mobile vision system for robust multi-person tracking. In *CVPR*, pages 1–8, June 2008.

[10] C. Fanti, L. Zelnik-Manor, and P. Perona. Hybrid models for human motion recognition. In *CVPR*, volume 1, pages 1166–1173 vol. 1, June 2005.

[11] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, pages 1–8, June 2008.

[12] V. Ferrari, M. Marin-Jiminez, and A. Zisserman. Progressive search space reduction for human pose estimation. In *CVPR*, pages 1–8. IEEE, June 2008.

[13] A. Hakeem and M. Shah. Learning, detection and representation of multi-agent events in videos. *Artif. Intell.*, 171(8-9):586–605, 07.

[14] D. Hoiem, A. Efros, and M. Hebert. Putting objects in perspective. In *CVPR*, volume 2, pages 2137–2144, 2006.

[15] Y. Ke, R. Sukthankar, and M. Hebert. Efficient visual event detection using volumetric features. In *ICCV*, volume 1, pages 166–173 Vol. 1, Oct. 2005.

[16] T. Kim, S.-f. Wong, and R. Cipolla. Tensor canonical correlation analysis for action classification. In *CVPR*, June 2007.

[17] O. Lanz. Approximate bayesian multibody tracking. *PAMI*, 28(9):1436–1449, Sept. 2006.

[18] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, pages 1–8, Anchorage, Alaska, June 2008.

[19] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, volume 2, pages 2169–2178, 2006.

[20] J. Li, S. Gong, and T. Xiang. Scene segmentation for behaviour correlation. In *ECCV*, pages 383–395, Berlin, Heidelberg, 2008. Springer-Verlag.

[21] J. Niebles and L. Fei-Fei. A hierarchical model of shape and appearance for human action classification. In *CVPR*, pages 1–8, June 2007.

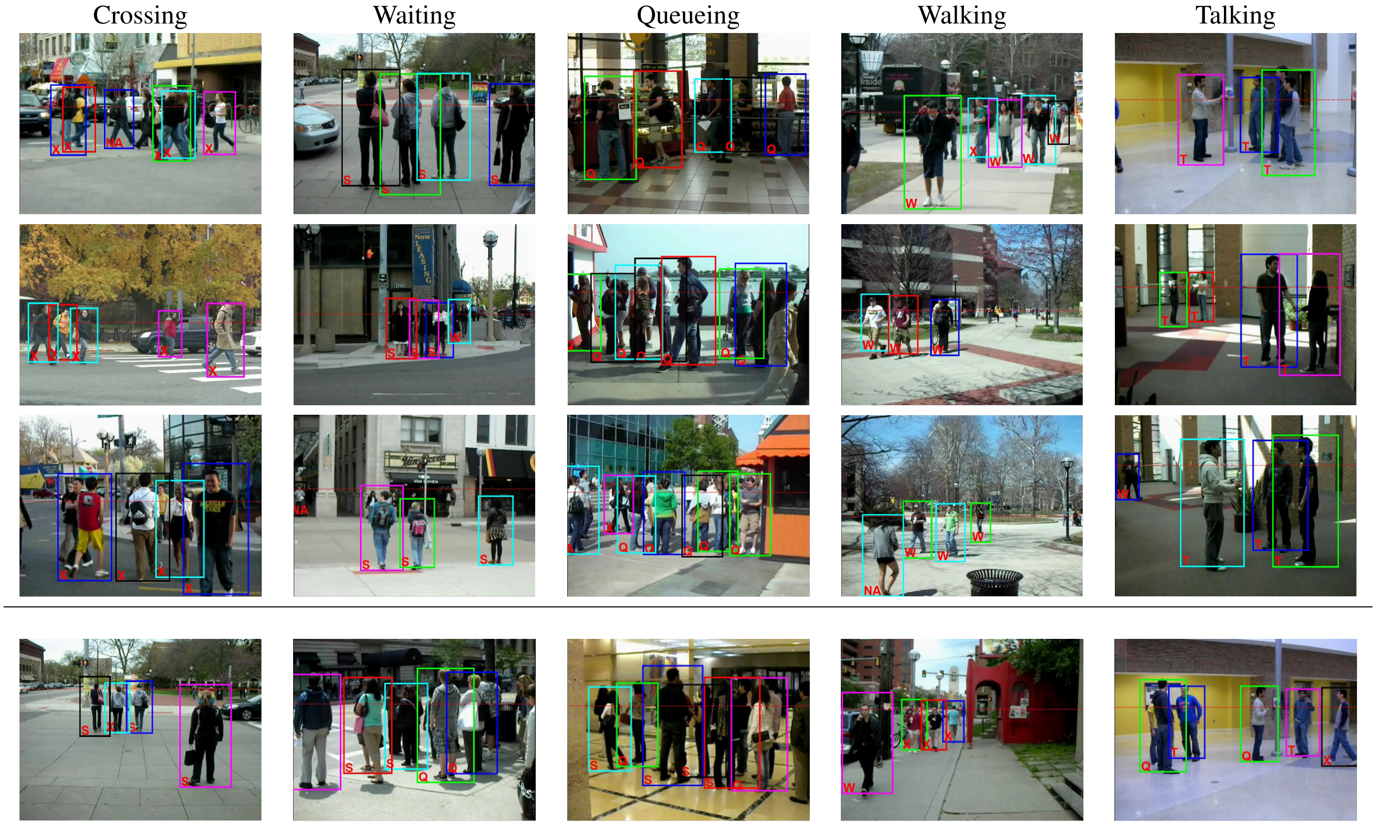


Figure 9. Example results. Top 3 rows show examples of good classification and bottom row shows examples of false classification. Estimated horizon is overlayed onto the images as a red dotted line. The algorithm fails in classificaiton when estimated poses are too much different from actual poses or collective activity is not well-defined by people's movement (4^{th} crossing example). The labels X, S, Q, W, T and NA indicate crossing, waiting, queueing, walking, talking and not assigned, respectively. When there is not enough information for a STL descriptor, it will be left NA and ignored as described in Fig.7. This happens when a person appears suddenly or when there are not enough people in the scene.

[22] J. C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. *IJCV*, 79(3):299–318, Sep. 2008.

[23] D. Ramanan, D. Forsyth, and A. Zisserman. Tracking people by learning their appearance. *PAMI*, 29(1):65–81, Jan. 2007.

[24] X. Ren, A. Berg, and J. Malik. Recovering human body configurations using pairwise constraints between parts. In *ICCV*, pages 824–831, 2005.

[25] P. Turaga, R. Chellappa, V. Subrahmanian, and O. Udrea. Machine recognition of human activities: A survey. *Circuits and Systems for Video Technology, IEEE Transactions on*, 18(11):1473–1488, Nov. 2008.

[26] O. Tuzel, F. Porikli, and P. Meer. Pedestrian detection via classification on riemannian manifolds. *PAMI*, 30(10):1713–1727, Oct. 08.

[27] P. Viola, M. J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *ICCV*, page 734, Washington, DC, USA, 2003. IEEE Computer Society.

[28] G. Welch and G. Bishop. An introduction to the kalman filter. Technical report, Computer Science, UNC Chapel Hill, 1995.

[29] B. Wu and R. Nevatia. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In *ICCV*, volume 1, pages 90–97 Vol. 1, Oct. 2005.

[30] K. Yamaguchi, T. Kato, and Y. Ninomiya. Vehicle ego-motion estimation and moving object detection using a monocular camera. In *ICPR*, volume 4, pages 610–613, 0-0 2006.

[31] A. Yilmaz and M. Shah. Actions sketch: a novel action representation. In *CVPR*, volume 1, pages 984–989 vol. 1, June 2005.

[32] L. Zelnik-Manor and M. Irani. Event-based analysis of video. In *CVPR*, volume 2, pages II–123–II–130 vol.2, 2001.