Unsupervised Bayesian Detection of Independent Motion in Crowds

Gabriel J. Brostow and Roberto Cipolla
University of Cambridge
{gjb47 | cipolla}@eng.cam.ac.uk

Abstract

While crowds of various subjects may offer application-specific cues to detect individuals, we demonstrate that for the general case, motion itself contains more information than previously exploited. This paper describes an unsupervised data driven Bayesian clustering algorithm which has ***detection*** *of individual entities as its primary goal.*

We track simple image features and probabilistically group them into clusters representing independently moving entities. The numbers of clusters and the grouping of constituent features are determined without supervised learning or any subject-specific model. The new approach is instead, that space-time proximity and trajectory coherence through image space are used as the only probabilistic criteria for clustering. An important contribution of this work is how these criteria are used to perform a one-shot data association without iterating through combinatorial hypotheses of cluster assignments. Our proposed general detection algorithm can be augmented with subject-specific filtering, but is shown to already be effective at detecting individual entities in crowds of people, insects, and animals. This paper and the associated video examine the implementation and experiments of our motion clustering framework.

1. Introduction

Detection of individuals in *dense* crowds has received comparatively little attention in vision because it is at once a problem of segmentation, recognition, and tracking. Automatic analysis of crowd and traffic flow has mostly been approached using the same models of appearance and shape that apply to images containing one or few individual entities. Those techniques scale variously to handle larger numbers of entities, but generally struggle to initialize when the crowd is dense. Dense scenes have so many individuals that background subtraction and other preprocessing techniques fail to find meaningful boundaries between entities. We hypothesize that when dealing with video, just the independent motions of those entities can already offer good initialization for tracking in dense crowds.

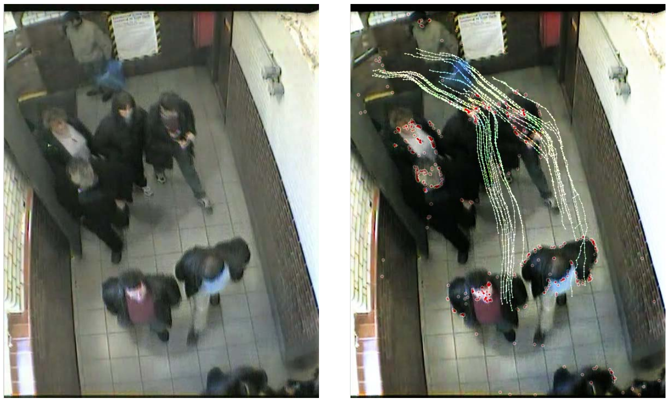
The central premise of our algorithm is that a pair of points that appears to move together is likely to be part of the same individual (see Figure 1). This assumption holds in many scenarios and has previously been exploited for rigid and nonrigid motion factorization [29, 23], but is particularly applicable to even somewhat-overhead views of crowds of people and other subjects. Such crowds are a particularly challenging class of video. The extensive occlusions, lighting variations, and significant pose-dependent appearance changes conspire to preclude repeated detection of the same features throughout a whole sequence.

We are inspired by Johansson's experiments where only "moving light" at the joints of one actor allowed perception subjects to recognize that it was human motion, and to identify the activities and eventually the gender of the performer [8]. Instead of one, we have many actors, who in their natural appearance, can be thought of as wearing "lights" at random and changing locations. When viewing video with such reduced content, human observers are able to detect the number and locations of separate individuals. We propose a new algorithm which performs pairwise Bayesian clustering to accomplish the same task automatically.

2. Related Work

There is already commercial interest to develop crowd detection systems for specific applications. Some stores use infrared video to roughly gauge the timing and density of customer flow [14], while visible light video is used to identify when groups of people are crossing the path of a moving vehicle [17].

However, the limitations imposed by significant occlusions in crowd videos render much of the previous human-detection and tracking related literature inappropriate. Most relevant, Zhao and Nevatia [31, 32] attack the problem of people tracking specifically, and have tracked the largest crowd of people thus far. They use articulated ellipsoids to model human shape, color histograms to model different people's appearance, and an augmented Gaussian distribution to model the background for segmentation. When moving head pixels are detected in the scene, a princi-



(A) (B)

Figure 1. (A) One characteristically noisy frame from input sequence *tunnel-A125*. (B) Features are marked here as red dots on white, and all current trajectories passing through a user-selected (for illustration only) region show differing paths, even when people are walking arm-in-arm. Despite perspective scale, the trace lines are closest to other lines generated by the same person.

pled MCMC approach is used to maximize the posterior probability of a multi-person configuration. Markov chain jump/diffusion dynamics are used to perform hypothesis testing of the *numerous* possible configurations. The algorithm is tested to give quantitative results on crowds of up to 33 people. See [9] for interesting examples of using related techniques to track more sparse "crowds" of ants, or hockey players in the case of [15]'s impressive boosted particle filter. These algorithms skip the modeling of articulations in favor of appearance models trained for the specific unoccluded appearance of their respective subjects. Such training was sufficient for Leibe *et al*. [10] to detect several pedestrians at a time in street-level photos of even cluttered city scenes.

The second most related work is that of Tu and Rittscher and Rittscher *et al*. [24, 18]. Both systems group an image's spatial features, performing a global annealing optimization that propagates the certainty at distinct person-boundaries to uncertain areas where those people's outlines are ambiguous. The earlier work assumes an overhead view, modeling people as roughly circular collections of periphery vertices. By maximizing the connectedness of each graph, detection cliques emerge reliably when at least half of a person's contour is distinct. In the newer work, the iterative feature grouping is interleaved with a generative model that proposes the number, location, and $2D$ bounding box parameters of people's shape until an image's observations have a consistent explanation. Seen from lower angles, observations consist of sections of people's background-subtracted bounding contours that are automatically tagged as a person's top, bottom, or either side. The detection results are robust to various scale, shadow, and viewing angle conditions, and serve as good initialization for a template based tracker that was demonstrated on video of nine people. Our proposed algorithm finds similar initial entity locations, but in dense and crowded situations where boundary contours are unavailable, while motion cues are.

Isard and MacCormick's BraMBLe system [6] uses a very simple motion model to help with detection in ambiguous situations. The joint inference on both the number of objects and their configurations is shown to perform efficiently on scenes containing up to three people at a time, with complex interactions and only a single foreground model. We established that an extension is needed that would efficiently create initialization samples in crowded scenes. This is an element crucial to the success of other particle-system techniques as well [25, 21], despite their more specific models of the scene, human appearance, and human motion.

The mutually supportive detection and tracking of Ramanan and Forsyth [16] finds multiple people by clustering relevant image patches. They, like [13, 20, 4], and [27] for the static case, have a probabilistic scheme for finding body part primitives that are likely to fit together. Such probabilistic assembly, whether to satisfy kinematic constraints or other training data, has the attractive quality of generating person-location hypotheses from low level features, even when some components are occluded [26]. Our approach of tracking-before-detection operates at an even lower level, where detection is based on clustering of *image* features, that we need not identify explicitly.

Our work can be seen as a type of middle ground between the domains of motion segmentation and multi-body factorization. Layered motion segmentation depends on motion discontinuities and texture to reveal the relationship between patches of pixels. The recent work of [5] elegantly demonstrates the flexible utility of clustering flow vectors. With example data used to train their Bayesian clustering, they are able to distinguish eight classes of facial expressions. With sufficient training data, it is conceivable that their system could learn to distinguish the same flow patterns that we look for explicitly (space-time proximity and movement in unison).

Data used in multi-body factorization [3, 29] is typically like our own in that sparse features are tracked, but texture support is limited. Also similar is the shared NP-complete challenge: minimizing the energy of off-diagonal blocks in factorization compares to associating tracked features with the different possible configurations of number and location of people. However, with the exception of Gruber and Weiss [3], motion factorization is ill-equipped to handle observations drawn from even moderately nonrigid bodies. Typically, motion factorization is known for breaking down in the face of even mild noise, and is fairly dependent on complete tracking (no dropouts) of the features being clustered. Our simple approach differs in spirit from that of Gruber and Weiss in that our move-in-unison motion model is intentionally naive, so it can generalize to highly variable motion of subjects. Further, we implicitly have temporal coherence without iterating, and measure travel distance in both time and image space. The context for distinguishing between noise and meaningful motion is more forgiving in our problem domain of crowds because space-time proximity serves as a powerful prior.

3. Bayesian Framework

A $2D$ image feature, x, traces out a trajectory, X, when tracked over time. Assuming several feature points move together on each person in the scene, we want the most probable clustering of points given distance matrix $\mathbf{Z}(X_{1:N})$, the symmetric distance-measure $Z(X_i, X_j)$ applied to all the point trajectories (defined with the likelihood in 3.1). Ideally, this means choosing the most probable clustering arrangement hypothesis H_m among M combinations: $P(H_m|\mathbf{Z})$. M is obviously very large since it enumerates the combinatorial ways the X_n's could be "joined," *i.e.* regarded as belonging to the same bodies. The number of bodies is itself unknown.

We suggest that the search for H_m can be made tractable, even for large crowds, if we exploit general motion information to constrain the size of M. We propose that pairwise decisions about whether or not to merge groups of features can reveal clusterings. Referring to one or more features X that are previously clustered as C_i, we use a probabilistic framework to decide for or against merging two clusters C_i and C_j: $P(C_i \cup C_j | Z(X_{Ci}, X_{Cj}))$, where $X_{Ci} = \{X_n : n \in C_i\}$:

$$P(C_i \cup C_j | Z(X_{Ci}, X_{Cj})) = \frac{P(Z(X_{Ci}, X_{Cj}) | C_i \cup C_j) \times P(C_i \cup C_j)}{P(Z(X_{Ci}, X_{Cj}))} \qquad (1)$$

Besides finding an implementation of (1), we also have a choice of ways for combining the pairwise results. Any greedy or coarse-to-fine approach to merging of merged clusters must (1) have a repeatable means of choosing with which pairs to start, (2) have an appropriate criterion for stopping the merging process, and (3) be robust to shared features, *i.e.* features on person-person boundaries. Such features are troublesome when they are just similar enough to two separately moving clusters to act as a bridge. We avoid the first two requirements by performing a one time flat evaluation of all $(\frac{n(n-1)}{2}) = O(n^2)$ pairings within C_n, and then building minimum spanning trees in the resulting merged and disjoint space of pairs. Note that while an X_{Ci} naturally contains only one tracked feature X, practical situations may start off with known groupings of X's.

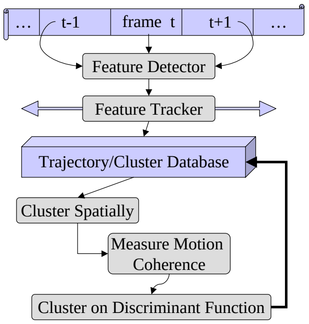


Figure 2. The system implementing our algorithm can run on short sequences or continuously (and in parallel) through use of a database server.

3.1. Algorithm

Our algorithm, illustrated in Figure 2, starts with image-features tracking. Appearance is not used because our aim is to evaluate the performance of the single cue of motion, for purposes of detection. Incorporating the two cues would be an extension of this algorithm. For a detected feature to be useful, it must be tracked with a high degree of confidence both forward and backward in time. For detection, we locate both Rosten-Drummond features [19] and Tomasi-Kanade features [22]. A hierarchical optical flow implementation [1] of [11] tracks all the features for two frames. We experimented with several other features such as Harris corners. While not substantially different, our approach found a superset of those corners.

To quickly isolate promising features among each of frame f's pool of detected corners D_f, we corroborate that independent feature finding in two subsequent frames agrees in accuracy to within one pixel. Function $W(D_f, n)$ returns the image coordinates of projecting corners in D_f along their optical flow paths forward for n frames:

$$W(D_f, 2) = W(D_{f+1}, 1) = D_{f+2}. \qquad (2)$$

A feature in D_f which satisfies (2) is then treated as an image feature x. In processing a finite video, x is tracked in subsequent frames until it is lost, then again in previous frames in a second pass through the video in reverse. While optical flow helps initialize reliable features, accurate and faster tracking is accomplished by searching a $2D$ window around each x using normalized cross correlation.

Ideally, the size of the $2D$ search window would be adjusted for different parts of the image using a known ground-plane calibration. This can be determined using [12], or as in our implementation, by hand-clicking an approximate horizon line. Many alternative and adaptive tracking techniques exist (*e.g.* [7]), but we found sufficiently reliable trajectories X_n in all our videos by simply following a small (image-space dependent) search window of the initial template, until its correlation fell below 0.96. This was determined empirically as a conservative threshold. To compare two trajectories X_i and X_j, which respectively extend in time over Δt_i and Δt_j, we consider only the overlapping range of frames $\{f_n : n \in \Delta t_i \cap \Delta t_j\}$.

Spatial Prior To perform clustering in a frame f, we employ the X_n trajectories which have data at f. Each currently active X is sampled in time extending ± 30 frames. Where an X's data runs out, the missing samples are generated as linear extrapolations of the last known velocity. To calculate the prior term from (1), $P(C_i \cup C_j)$, we compute a clustering of these sampled loci, shown in Figure 4(A). The Euclidean distance between each of the locus pairs (X_i, X_j) is used to build a distance tree. The tree is assembled following the criterion of furthest distance: $max(dist(X_{r_i}, X_{s_j})), i \in (1, \ldots n_r), j \in (1, \ldots n_s)$. This tree is split into c clusters, where c is chosen manually (once) as $3 - 5$ times the number of bodies that could fit in the field of view. Choosing a c that matches the actual number of bodies can speed up computations, but oversegmenting is safe and preferable, with the upper limit that this prior ceases to be useful as c exceeds half the number of features $\frac{|X|}{2}$. The prior probability of a hypothesized merging of C_i and C_j is calculated as

$$P(C_i \cup C_j) = \begin{cases} 1 & C_i, C_j \in c, \text{ and are neighbors} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

This prior acts as a noisy but effective initialization; X's that are too far apart spatially are ignored as possible pairwise candidates for clustering when calculating the posterior.

Even if specifying a very high c for oversegmentation, features can initially be forced into the same prior C_i despite disparate trajectories when many dissimilar noise features elsewhere in the scene use up the available clusters. For this reason, we precede the final clustering stage (which only merges) with a cleaving of the prior's X_{Ci} whenever islands of features exceed a distance to their neighbors of 5 in (4)'s distance metric. 5 was chosen empirically because it excluded grossly dissimilar motions for all our sequences.

Coherent Motion Likelihood To compute the likelihood term of (1), $P(Z(X_{Ci}, X_{Cj})|C_i \cup C_j)$, we start with the constituent trajectories of C_i and C_j, X_{Ci} and X_{Cj} (all original samples, unlike the prior). We seek a $P(Z(X_{Ci}, X_{Cj}))$ that relates the two sets of trajectories in proportion to the probability that all points $\{X_{Ci}, X_{Cj}\}$ moved together on one body. We use the assumption that two individual features, X_u and X_v are more likely to come from the same body if the variance in distance between them is small:

$$Q(X_u, X_v) = \frac{1}{1 + Var(X_u, X_v)}, \tag{4}$$

where $Var(X_u, X_v) = Var(Distance_{Eucl}(X_u, X_v))$ for the Δt frames of overlap. Ideally, two features moving on a rigid body would be a constant distance apart, yielding $Q(.) = 1$. In reality, this is only true of motion parallel to the image plane, but has proven to be an acceptable approximation, in part because perspective scale varies less when trajectory overlap is more brief. Figure 4(B) shows lines between feature pairs that remain after applying a minimal threshold on $Q(.)$. With $Q(X_u, X_v)$ as a measure of the probability that X_u and X_v moved together rigidly, we compute the class conditional probability that all points X_{Ci} and X_{Cj} did as well:

$$P(Z(X_{Ci}, X_{Cj})|C_i \cup C_j) = \prod_{u,v: X_u, X_v \in (X_{Ci} \cup X_{Cj})} Q(X_u, X_v) \tag{5}$$

$P(Z(X_{Ci}, X_{Cj}))$ is a measure of the inter-cluster variance for $(i \neq j)$, and the intra-cluster variance when $(i = j)$.

Evidence Ideally, the normalization term $P(Z(X_{Ci}, X_{Cj}))$ from (1) represents the unconditional probability of observing all features X_{Ci} and X_{Cj} moving together rigidly, among all other possible hypotheses H_m of which clusters moved together and were observable as such:

$$\sum_{m=1}^{M} P(Z(X_{Ci}, X_{Cj}|H_m))P(H_m) = 1. \tag{6}$$

As an approximation, we instead compute $\sum Q(X_i, X_j)$ over $\{X_i \in C_i, X_j \in C_j\}$ as a fraction of the same sum but with $\{X_i, X_j \in C_{1 \ldots N}\}$. The $P(Z(X_{Ci}, X_{Cj}))$ represents the fraction of "good" feature-to-feature pairings just between cluster i and j to the number found throughout the whole network of X's.

Discriminant Function With the established means of computing the posterior $P'_{ij} = P(C_i \cup C_j|Z(X_{Ci}, X_{Cj}))$ for each pair of clusters (C_i, C_j), we make a single decision about whether or not to merge them. A log ratio discriminant function

$$S_{ij} = -ln(P'_{ij}) - (-ln(P'_{ii}) - ln(P'_{jj})), \tag{7}$$

compares the probability of a joined cluster $C_{ij} = (C_i \cup C_j)$ to that of two separate clusters, C_i and C_j (see Figure 3). We could just as well view this as a minimum description length (MDL) problem; positive S_{ij} indicates that the first term has a greater message length than the last two, so C_i

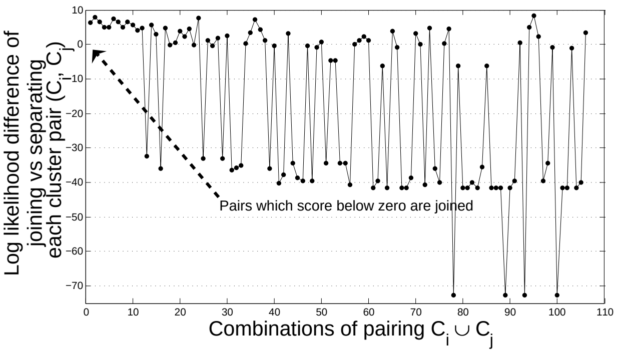


Figure 3. $\frac{n(n-1)}{2}$ pairwise comparisons whether candidates for merging C_i and C_j are more probably separate clusters (> 0) or part of the same (< 0). Pictured are the cluster pairs for frame 20 of the *tunnel-A125* sequence. Examining the pairs individually, we find that falsely separated and falsely joined clusters scored $+1$ and -1 respectively, and in case of false separation, other unions can still bring the two together indirectly.

and C_j should be "sent" separately. Leaving cluster C_i and C_j separate when $S_{ij} > 0$ and joining them when $S_{ij} \leq 0$ for all unique pairings has the effect of indirectly connecting clusters into minimum (or maximum probability) spanning trees, shown in close-up in Figure 4(C).

Implementation Trajectories are stored as a simple data structure with fields for image coordinates, frame number, and per-frame cluster assignments. We eventually built an SQL database interface to address the bookkeeping challenges of long sequences. The custom data types of MySQL allow storage and analysis of data streams extending months and years. An added future benefit is that speed can be increased using multiple computers, which can access the database to perform parts of this highly parallelizable algorithm.

4. Results

The footage used in our experiments features crowds of specific subjects, including bees, ants, penguins, and mostly humans. The 10 sequences we tested vary in length from 3sec. to one hour, and the videos of human crowds also vary significantly in density and viewing angle. Some segments are compressed and small, have interlacing artifacts, significant video noise, and regions that are out of focus. These flaws are typical of the expected quality of footage featuring public spaces, and motivated our research into the extent to which non-appearance based detection would segment crowds.

Different spatial clusters, representing the computed prior, are color coded for a frame from sequence *escalator-A128* (see Figure 4(A)). Knowing the perspective scale from the ground plane calibration is again useful here, since the prior can eliminate (C_i, C_j) pairs from the pool of merge-candidates if they are too far apart in "depth," saving time and possibly preventing false clustering unions. Spatial clustering of each frame takes an average of 5 seconds. For illustration purposes, the likelihoods are rendered in Figure 4(B) by drawing an edge between any pair of features that do not meet the cleaving criterion. Calculating the likelihoods is the slowest stage because frames with many image features must evaluate many candidate pairings (additional 5 sec. to 3 min.). Parallelization and a hierarchical implementation will improve performance significantly.

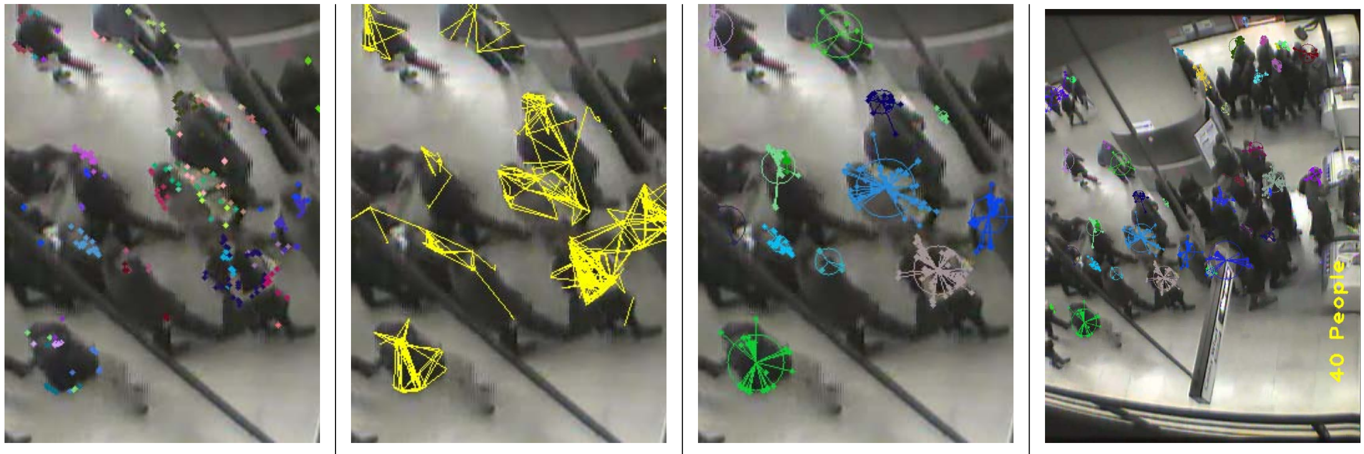
The implemented algorithm detects entities that exhibit independent motion, even under crowded and noisy conditions. The final cluster assignments, or detected entities, that result from building spanning trees in the space of successful pairings are shown in Figures 4(C,D), 5, 7, 8, and 9(B) for some of our test sequences. These and other sequences appear in the associated video. Note that tree nodes are rendered only if a cluster has 3 or more non colinear features.

While features are tracked over time, the detections are computed separately for each frame (*i.e.* **entities are not tracked**). Entities inevitably lose some image features and acquire new ones as they move about. Consequently, what is detected as an individual entity in one frame may later reveal itself to be multiple individuals that had been moving in unison. Conversely, a detected individual may eventually merge with other entities if its distinguishing feature trajectories are replaced by features that move in synch with a neighbor.

One can not expect a general-purpose motion segmentation algorithm to perform favorably against subject-specific trackers. However, one measure of how fully we are exploiting motion is the degree to which those systems' results are approximated. For qualitative comparison, the video and Figures 7 and 8 show side-by-side performance of our independent motion detector on the sequences from Zhao and Nevatia's [32] and Rittscher *et al.*'s [18] people tracking, respectively.

Emphasizing again that the aim is only to show that detection of independent motion has potential to aid in people-tracking, we also performed a limited quantitative comparison. Because of the large number of people traveling through the fairly long sequences, Table 1 summarizes the performance of our implementation against hand labeled ground truth for only several independent frames, chosen from the *tunnel-A125* sequence by a random number generator. Zhao and Nevatia's scores are listed for convenience, as the performance of a subject-specific tracker is naturally superior.

Finally, as a sample application that employs our algorithm to detect entities, in this case people, we implemented a gaze-direction visualization system, pictured in Figure 6.



(A) (B) (C) (D)

Figure 4. Result of clustering coherent motions in frame 94 of sequence *escalator-A128*: (A) Spatial clustering prior, (B) motion coherency likelihood (thresholded for illustration only), (C) resulting disjoint clusters (D) people-counter reports 40 individual bodies. Please see the submitted video to examine this sequence.

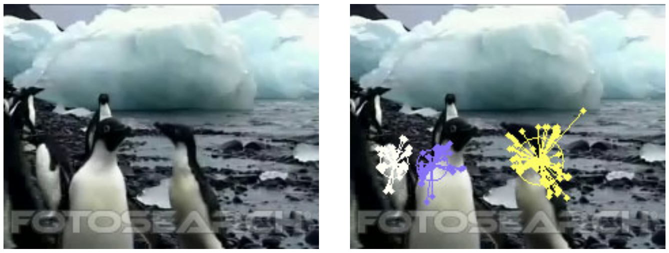
	Ours	Zhao & Nevatia'04
distinct detections	144	8466
correctly detected	136	7881
missed detections	8	585
false detections	33	291
detection rate	94%	93.09%
false detection rate	22.9%	3.43%

Table 1. While the rate of correct detections is comparable to Zhao & Nevatia's [32], our false detection rate is substantially worse. We expect that temporal averaging or even a simple threshold on the minimum number of consecutive detections will improve our algorithm's score. However, situations where the individuals' arms, legs, and luggage are visibly moving will continue to appear as distinct to our algorithm which has no human body model. The "distinct detections" criterion is counting distinct detections in our systems, but counts repeated detections of each tracked individual in Zhao & Nevatia's system, which saw a maximum of 33 people at any one time.

The CG floor and walls were calibrated and modeled by hand, but the rendered sequence was generated by the algorithmic equivalent of attaching a headlamp to each detected individual. Currently, advanced architectural floorplan visualization systems still operate using *isovists* [2], which measure the gaze of only one person.

5. Conclusions

A simple unsupervised Bayesian clustering framework for detecting individuals in moving crowds is the main contribution of this paper. This probabilistic clustering of low level image features is surprisingly good at finding a first approximation of the number and location of individual entities in crowded video sequences. Footage of animals, insects, and complex pedestrian traffic containing significant occlusions, noise, and perspective foreshortening is processed in a one-shot fashion, without the benefit



(A) (B)

Figure 5. (A) Frame 115 from *penghurry-01*. (B) Features on all three moving penguins are correctly detected to be independent.

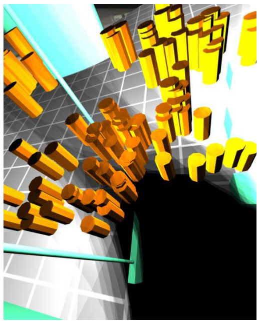


Figure 6. Sample image of gaze-rendering application, fed by the *escalator-128* results (pictured in Figure 4(D)). Note that bright areas in the gaze-rendering image are "seen" by many people, while dark regions, such as the hanging sign in the lower right, are likely being ignored.

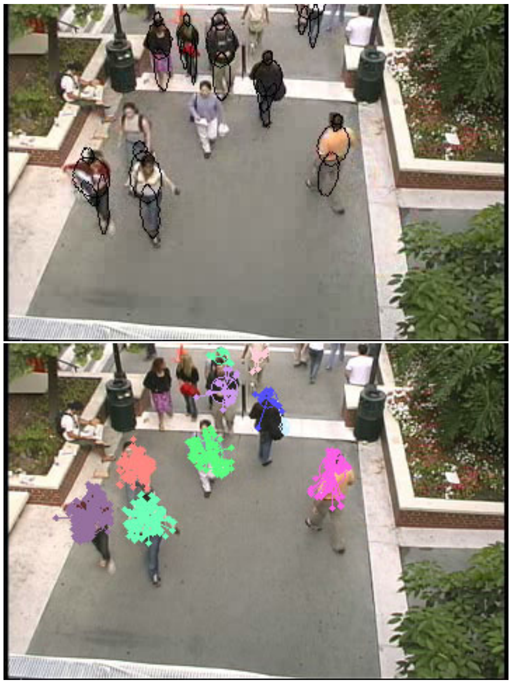


Figure 7. Frame 540. Example results from Zhao and Nevatia's multiple-human tracker [32] (above) on their *Commons01* sequence, and the results of our independent-motion detection. Note different false negatives in both. See video for entire sequence.

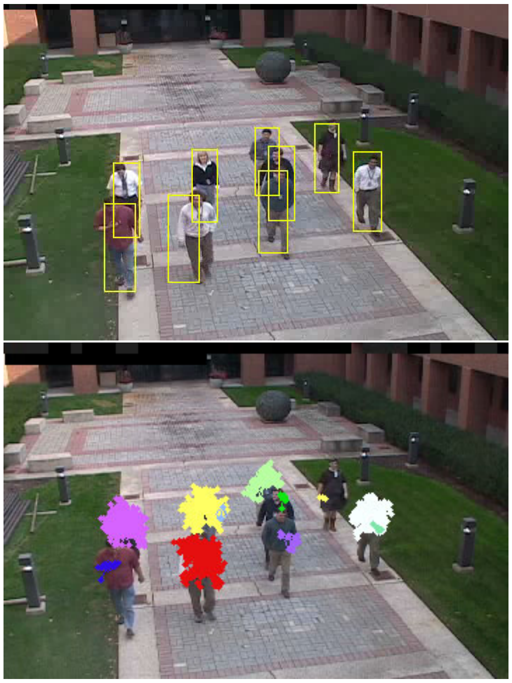


Figure 8. Example results (above) from Rittscher *et al*.'s multiple-human tracker mentioned in [18], and the results of our detection of independent motion.

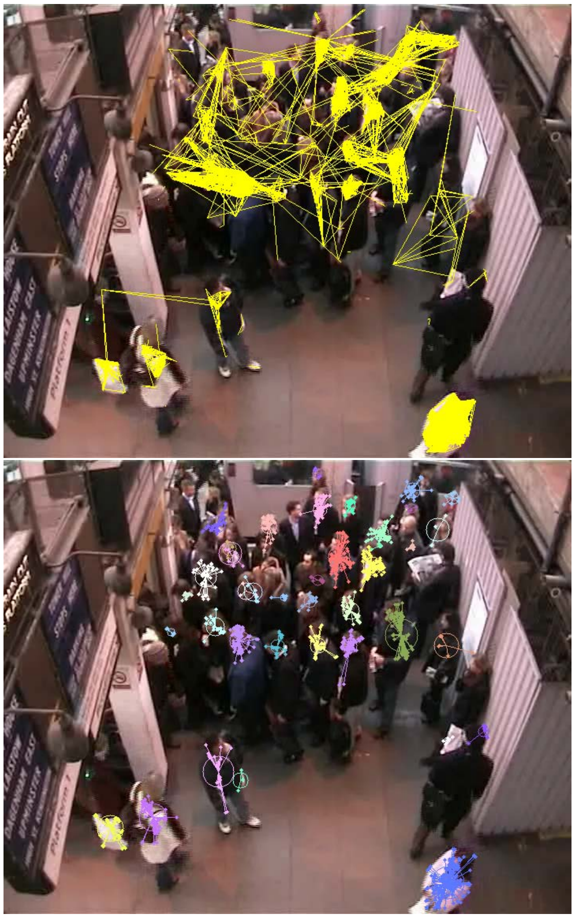


Figure 9. Sequence *subway-B152* is a minute long and features our most dense pedestrian traffic. (A) Pairs of features with high likelihoods of being joined. (B) Isolated groups of features result from applying the discriminant function. Please view the video results for an excerpt of the sequence.

of training data or any notion of an appearance model at all. In our experiments, we found that the trajectories of tracked features are usually unique enough that joint evaluation of different hypotheses is unnecessary. This finding has the secondary effect that the complexity of non-temporally-smoothed entity detection is primarily limited by the scene complexity, and less by the number of individuals.

The limitations of our motion-only approach are not unexpected. First, if an individual is camouflaged so that image features, in particular corners, are absent, then all subsequent clustering will ignore that individual. Features on one body are assumed to be moving rigidly, so features from the same body can erroneously be left separated if the body deforms or exhibits sustained articulations. Interestingly, leg motion is less of a problem in our experiments than arms precisely because of the extensive person-person occlusion. The most common false positives occur in pedestrian scenes when persons carry items such as newspapers or backpacks, which presents an interesting sub-problem of moving-object identification. False negatives occur when two (though possibly more) bodies move near each other and in step. With this method, we can expect no meaningful results from *e.g.* footage of a marching army, or people standing in place. Finally, while the algorithm uses features tracked over sequences of frames, the clustering and detection of individuals is happening independently for each frame, meaning that the current system does not track entities per se.

There are many possible extensions of this work since robust *tracking* of individuals in dense crowds will benefit from merging our motion based detection into various appearance based methods of filtering and tracking. To start, the Bayesian clustering presented here can be applied to extended sequences where periods of particularly reliable clustering and sparse flow would allow for autocalibration. The existing system could benefit most from automatically tuning relative scale (perspective). These parameters would both limit the search area for feature tracking *and* could be incorporated as a prior for the image "footprint" size and lighting of subjects in different parts of the image.

There is room for obvious performance improvements, since this work has been assessing the value of motion, and the information content of appearance has been left out as the control. Wu & Nevatia [28] have a new appearance based approach to people tracking based on body part detection, which is very complimentary and will benefit from motion cues. Temporal smoothing is the next most obvious extension, and it would be trivial to automatically learn the layout of entrances and exits. These would in turn prevent the "spawning" of new entity hypotheses in the middle of the scene. It is also exciting to imagine that pedestrian traffic flow patterns could be learned, with the aim of predicting or filling in tracks missing due to occlusion or local camouflage. Finally, further experiments and ground truth testing are needed to objectively compare algorithms such as [30] against our own in the context of crowd scenes.

6. Acknowledgements

We thank Tom Drummond and David MacKay for useful discussions. We are very grateful to Niccolò Caderni of Legion Intl., Tao Zhao, Peter Tu, Alan Lerner, Frank Dellaert and Zia Khan for sharing their crowd videos. The *penghurry-01* sequence comes from fotosearch.com. The first author was supported by funding from the Cambridge-MIT Institute.

References

[1] G. Bradski. Opencv: Examples of use and new applications in stereo, recognition and tracking. In *VI02*, page 347, 2002.

[2] R. Conroy and N. Dalton. Omnivista: An application for isovist field and path analysis. Space Syntax - III International Symposium, Atlanta, GA, 2001.

[3] A. Gruber and Y. Weiss. Multibody factorization with uncertainty and missing data using the em algorithm. In *CVPR (1)*, pages 707–714, 2004.

[4] I. Haritaoglu, D. Harwood, and L. S. Davis. W4: Real-time surveillance of people and their activities. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):809–830, 2000.

[5] J. Hoey and J. J. Little. Bayesian clustering of optical flow fields. In *ICCV*, pages 1086–1093, 2003.

[6] M. Isard and J. MacCormick. Bramble: A bayesian multiple-blob tracker. In *ICCV*, pages 34–41, 2001.

[7] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi. Robust on-line appearance models for visual tracking. In *CVPR*, volume 1, pages 415–422, Dec. 2001.

[8] G. Johansson. Visual motion perception. *Scientific American*, 14:76–88, 1975.

[9] Z. Khan, T. R. Balch, and F. Dellaert. An mcmc-based particle filter for tracking multiple interacting targets. In *ECCV (4)*, pages 279–290, 2004.

[10] B. Leibe, E. Seemann, and B. Schiele. Pedestrian detection in crowded scenes. In *CVPR (1)*, pages 878–885, 2005.

[11] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision (ijcai). In *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI '81)*, pages 674–679, April 1981.

[12] F. Lv, T. Zhao, and R. Nevatia. Self-calibration of a camera from video of a walking human. In *ICPR (1)*, page 562, 2002.

[13] K. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *ECCV (1)*, pages 69–82, 2004.

[14] H. Nanda and L. Davis. Probabilistic Template Based Pedestrian Detection in Infrared Videos. In *Procs. IEEE Intelligent Vehicles Symposium 2002*, Versailles, France, June 2002.

[15] K. Okuma, A. Taleghani, N. de Freitas, J. J. Little, and D. G. Lowe. A boosted particle filter: Multitarget detection and tracking. In *ECCV (1)*, pages 28–39, 2004.

[16] D. Ramanan and D. A. Forsyth. Finding and tracking people from the bottom up. In *CVPR (2)*, pages 467–474, 2003.

[17] P. Reisman, O. Mano, S. Avidan, and A. Shashua. Crowd detection in video sequences. In *IVS04*, pages 66–71, 2004.

[18] J. Rittscher, P. H. Tu, and N. Krahnstöver. Simultaneous estimation of segmentation and shape. In *CVPR (2)*, pages 486–493, 2005.

[19] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. In *European Conference on Computer Vision (to appear)*, May 2006.

[20] A. Shashua, Y. Gdalyahu, and G. Hayun. Pedestrian detection for driving assistance systems: Single frame classification and system level performance. In *IVS04*, 2004.

[21] H. Tao, H. S. Sawhney, and R. Kumar. A sampling algorithm for tracking multiple objects. In *ICCV '99: Proceedings of the International Workshop on Vision Algorithms*, pages 53–68. Springer-Verlag, 2000.

[22] C. Tomasi and T. Kanade. Detection and tracking of point features. Technical Report CMU-CS-91-132, Carnegie Mellon University, April 1991.

[23] L. Torresani and A. Hertzmann. Automatic non-rigid 3d modeling from video. In *ECCV (2)*, pages 299–312, 2004.

[24] P. Tu and J. Rittscher. Crowd segmentation through emergent labeling. In *ECCV Workshop SMVP*, pages 187–198, 2004.

[25] K. J. Venegas S and T. J. Multi-object tracking using the particle filter algorithm on the top-view plan. Technical report ITS-02-04, 1015 Ecublens, 2004.

[26] P. Viola, M. J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*, page 734, Washington, DC, USA, 2003. IEEE Computer Society.

[27] B. Wu and R. Nevatia. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In *ICCV*, October 2005.

[28] B. Wu and R. Nevatia. Tracking of multiple, partially occluded humans based on static body part detection. In *CVPR*, June 2006.

[29] L. Zelnik-Manor and M. Irani. Degeneracies, dependencies and their implications in multi-body and multi-sequence factorizations. In *CVPR (2)*, pages 287–293, 2003.

[30] L. Zelnik-Manor and P. Perona. Self-tuning spectral clustering. In L. K. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*. MIT Press, Cambridge, MA, 2005.

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

[32] T. Zhao and R. Nevatia. Tracking multiple humans in crowded environment. In *CVPR (2)*, pages 406–413, 2004.