

# Detecting Shopper Groups in Video Sequences

Alex Leykin

Department of Computer Science  
Indiana University  
Bloomington, IN 47405-7104

Mihran Tuceryan

Department of Computer Science  
Indiana University - Purdue University Indianapolis  
Indianapolis, IN 46202-5132

## Abstract

*We present a generalized extensible framework for automated recognition of swarming activities in video sequences. The trajectory of each individual is produced by the visual tracking sub-system and is further analyzed to detect certain types of high-level grouping behavior. We utilize recent findings in swarming behavior analysis to formulate a problem in terms of the specific distance function that we subsequently apply as part of the two-stage agglomerative clustering method to create a set of **swarming events** followed by a set of **swarming activities**. In this paper we present results for one particular type of swarming: *shopper grouping*. As part of this work the events detected in a relatively short time interval are further integrated into activities, the manifestation of prolonged high-level swarming behavior. The results demonstrate the ability of our method to detect such activities in congested surveillance videos. In particular in three hours of indoor retail store video, our method has correctly identified over 85% of valid "shopper-groups" with a very low level of false positives, validated against human coded ground truth.*

## 1 Introduction

Visual surveillance is entering a new more intelligent phase. Vision systems are no longer simply recording the observed visual information, but attempt to extract low-level motion information and, lately, analyze complex behaviors in the scene. Of particular interest for marketing intelligence are moving customers, the products or fixtures they interact with as well as how they interact with each other. Detecting shopper groups can provide several useful statistics, to be subsequently utilized by marketing research community and implemented in practice by retailers. This is particularly so as marketing intelligence is entering a new stage of managing customer experience, where such indicators as store traffic, shopping path, aisle penetration, dwell time, product interaction and conversion rate become of essence.

This work was inspired by the studies of swarming

behaviors in living organisms and their consequent implementations for modeling systems with complex intra-connectivities. One of the early implementations of swarming intelligence was presented by Reynolds [13] for animating the flock of CG birds. Each bird/actor there following a set of simple rules, such as steering toward the center of the flock and maintaining distance with other flock members. Intelligent swarms and in particular a technique called *particle swarm optimization* or *PSO* are currently applied for simulation of complex processes involving multiple locally-interacting agents [4]. In this approach the problem is modeled by particles in multidimensional space. These particles are flying through hyperspace (i.e.,  $\mathbb{R}^n$  and have two essential reasoning capabilities: their memory of their own best position and knowledge of the swarm's best, i.e. the particle with the smallest objective value. Members of a swarm communicate good positions to each other and adjust their own position and velocity based on these good positions. Our method is different from PSO in that the target value is already given to us by the tracker. From the generative approach we transgress to a recognition problem with out goal being: find the best set of particles that, given their tracking data, best behave as a group.

We present a framework to detect so-called "shopper groups" in tracked video sequences of retail stores. Our system uses tracked coordinates to detect a series of swarming events, i.e. the scenarios where several people behave with intrinsic group characteristics. There can be multiple events for a single group as people who enter the store as a group may repeatedly split apart and reconvene. Therefore swarming events serve as short-term manifestations of a more long-term group behavior, what we call a *swarming activity*. In section 2 we describe how two stages of agglomerative clustering can be used to detect shopper groups. At the first stage, to detect swarming events we employ a deterministic clustering of inter-actor discrepancies in location, orientation and dwelling status. The number of clusters and termination criteria is determined automatically by optimizing the clustering validity indexes. At the second stage our system integrates large quantities of swarming events

to obtain a shorter list of more meaningful clusters, corresponding to shopper groups. Considering several clustering methodologies we found that the fuzzy agglomerative techniques, such as proposed in [5] achieve the best segmentation and are robust to noise in form of outliers.

In computer vision community detection of shopper groups in checkout lines has been attempted by Haritaoglu [7]. For grouping authors use inter-body distances as well as such specific environmental clues as the cashier’s activities to determine the start and end of shopping transactions. Buzan et al.[3] perform trajectory-based clustering and retrieval, using a modified version of edit distance, called longest common subsequence. Similarities are computed between projections of trajectories on coordinate axes. Trajectories are grouped based, using an agglomerative clustering algorithm.

Rosario et al. [12, 14] have developed a framework based on coupled HMMs to recognize customer interactions in visual surveillance videos. In this work simple behaviors such as walking or changing direction are grouped into higher level interaction scenarios, for instance ‘approach, meet and walk together’. This is in our knowledge the closest to our work publication, with the key difference being in the time span that we consider to find shopper groups. Additionally, we formulate our model as a recognition of the fittest swarming behavior, which gives us a freedom to not establish explicit ties between event present in Markov modeling.

Another approach is to consider single- or multi-threaded event [8] with consequent events satisfying a predefined decision tree. Here activity is considered to be composed of action threads, each thread performed by a single actor. A thread is modeled by a stochastic finite automaton of event states, which are derived from the trajectory and shape of moving blob via Bayesian inference. However this method is more suitable to address single actor behaviors with a well defined time-sequential structure.

Some attention has been given to person-to-person interactions in context of security. In [1] the authors detect object hand on events by using context-free grammar parsing mechanism. The grammar and parser provide longer range temporal constraints, disambiguate uncertain low level detections, and allow the inclusion of a priori knowledge about the structure of temporal events in a given domain. Again the attention here is give to events happening sequentially in time, we will show that such approach fails when applied to long-term group detection.

## 2 Method

Here we outline a mechanism of applying intelligent swarm principles in a recognition task. Using a combination of

swarm-driven distance computation and advanced clustering methods we derive a two-stage generalized algorithm for higher-level human activity recognition.

### 2.1 Tracking Sub-System

To achieve a reliable result in recognizing group activities we seek for low-noise highly reliable input data. For this purpose we built a multi-pedestrian blob tracker [11, 10] with two key characteristics. Firstly, low level of false positive tracks is desired to hypothesize about swarming events. Secondly, the tracks have to remain continuous/uninterrupted for prolonged periods of time. This allows to single out the coincidental shopper groups in favor of actors re-appearing in the same swarm throughout larger spans of time. Figure 1 illustrate a single frame snapshot from a typical tracking sequence, with green ellipses representing projected spheroidal models of each human body and white labels above contain a unique tracked customer id.



Figure 1: Snapshot of the tracking sequence

Knowing the parameters of the camera, this tracker converts image coordinates of each person to 3D floor coordinates  $x, y$  ( $z = 0$ ). By modeling human body as a single spheroid it provides us with three additional parameters: width, height and orientation.

### 2.2 Defining Group Behaviors as Swarming Activities

By observing hundreds of hours of retail store surveillance videos we found some striking similarities in spatial interaction within shopper groups and spatial coordination of groups of animals. Because flocking model was able to recreate bird behavior so realistically [13] we decided to convert this method and use simple principles of group coordination to model groups of customers who shop together.

It is important to note that, for our purposes, we are neglecting the goal-oriented fashion in which people make their purchases. However, given specifics about product locations as well as some prior knowledge of customer habits the goals can be incorporated into our framework.

## 2.3 Detecting Swarming Events

Swarming events are defined as a short time activity sequences of multiple agents interacting with each other. An agent  $b$  is an instance of customer's path generated by the tracker, taken at current frame.

Depending on the type of swarming events to be detected, various proximity metrics or other heuristics can be used. In the case of grouping events, for each actor we used the relative position on the floor  $f_{xy}$ , body orientation azimuth  $\phi$  and binary dwelling state  $\delta = [T, F]$  to compute the metric as follows:

$$d(b_i, b_j) = w_1 |f_{xy_i}, f_{xy_j}| + w_2 |\phi_i, \phi_j| + w_3 |\delta_i, \delta_j| \quad (1)$$

$$D(e_i, e_j) = \sum d(b_i, b_j) \quad (2)$$

While considering clustering methodology it is important to keep two factors in mind: firstly the existing work outlining the principles of building a computationally efficient clustering algorithm and secondly, the specifics of the applied area which might dictate the choice of such method.

In our work we applied hierarchical clustering in two instances both of which are characterized by the presence of non-euclidean distances. More specifically, the rules that hold for standard "cloud of Euclidean points" clustering, (e.g. the fact that the cluster can be simply regarded as a centroid of its elements) will break under the absence of Euclidean geometry assumptions. Since no cluster means can be determined, some classical clustering validity measures (such as Davies-Bouldin index [9]) cannot be applied.

Additionally, the intuition behind the agglomerative approach is that the algorithm starts by introducing the least possible bias into the entire process. In other words, most likely agglomerations are made in the beginning when the price of error is the highest.

Once the metric is defined we start out with each body/actor representing a singleton cluster and iteratively agglomerate, by merging two closest sub-clusters. For each new step in clustering process, given the current clustering  $C_n, n \in [1, N]$  we compute clustering validity index (see [2]) as follows:

$$\begin{aligned} I &= \frac{\sum_{n=1}^N I_{n_i} + I_{n_c}}{N} \\ I_{n_i} &= \sum_{m=1}^M a_m \in C_n \\ I_{n_c} &= 1 - \mu \left( \frac{D(C_n)}{D(\{C\})} \right) \end{aligned} \quad (3)$$

Our method emphasizes computational efficiency, which is important with the  $O(N^2)$  computational complexity,

when clustering perhaps tens thousands of bodies or events (see section 2.4). Some natural limitations of the problem domain can be utilized to lower computation. For instance we use a minimum distance threshold for  $|f_{xy_i}, f_{xy_j}|$  when grouping two actors into an event (e.g. 300 cm), therefore much of the elements of the  $N^2$  matrix are not considered.

The validity index consists of the isolation index  $I_i$  and compactness index  $I_c$  computed over all existing clusters and normalized by their number. Isolation index for each node shows the percentage of nearest neighbors that belong to the same cluster. Compactness index indicates how compact the clusters are in comparison to the diameter of the entire node cloud (see Figure 2).

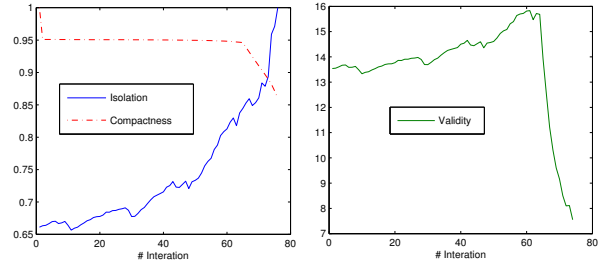


Figure 2: Clustering validity: isolation vs compactness

The clustering process continues until  $I$  stops increasing. At each step two closest clusters are computed based on their proximity according to equations 2, 4, and subsequently merged.

This way a distance is computed as a linear combination of three components: cohesion, co-alignment and co-dwelling.

## 2.4 Detecting Swarming Activities

We define swarming activities as prolonged higher level behavioral activities involving multiple human agents/actors and comprised of one or more swarming events, possible distant in time. In this paper we introduce a method of grouping swarming events into such activities based on their time co-ordination and agent composition.

For clustering purposes the centroid of an activity cannot be defined explicitly, instead the distance from event  $e_j$  to activity  $a_i$  is defined as a normalized weighted distance to all its constituent events.

$$D(a_i, e_j) = \frac{\sum_{\forall e_k \in a_i} u_{ik}^2 \psi_{ik} D(e_k, e_j)}{\sum_{\forall e_k \in a_i} u_{ik}^2 \psi_{ik}} \quad (4)$$

Where  $u$  are membership weights, constrained by eq. 6 and  $\psi$  are robust W-estimators defined in eq. 7.

As it can be observed from figure 3 there is a clear ridge in our distance function  $d$ . The distance function is design

to favor groups with all matching participating actors and the matching of average time distance between two events is of secondary importance.

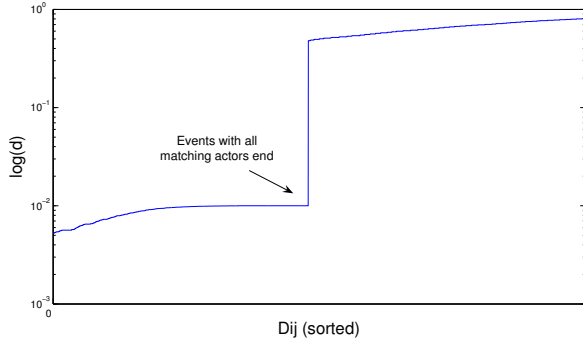


Figure 3: Distance function profile: sorted log distances between 262 swarming events

The profile of the graph 3 will vary, base on the problem dependent properties of the distance function. For detection of shopping groups, the critical role is given the the actor composition of the group (i.e. at least some people must be the same from one event to another) whereas the time distance, while important, has a lower weight.

The essential characteristic of our swarming event data is the presence of outliers as well as the fuzzy character of activity memberships. Some of the swarming events happen coincidental due to crowding effects in the store. To illustrate this point lets consider a two simple scenario in figure 4.

Considering above factors we have found the most successful to apply a robust fuzzy agglomerative clustering method (see [5]). It deals with the problem of outliers by weighing each distance function with robust statistics methods and it also maintains the fuzzy membership character that we seek for.

Let  $E = \{e_j | j = 1, \dots, N\}$  be a set of  $N$  swarming events and  $D_{ij}$  is a distance between events  $e_i$  and  $e_j$  as defined by eq. 2. Also, let  $A = \{a_i | i = 1, \dots, M\}$  be a target set of swarming activities. The fuzzy memberships are represented as matrix  $\bar{U} = u_{ij}$  for event  $e_j$  in activity  $a_i$ . The target function is:

$$G(A, \bar{U}; E) = \sum_{i=1}^M \sum_{j=1}^N u_{ij}^2 \rho_{ij} - \alpha \sum_{i=1}^M \left[ \sum_{j=1}^N \psi_{ij} u_{ij} \right]^2 \quad (5)$$

with constraint

$$\sum_{i=1}^M u_{ij} = 1, \text{ for } 1 \leq j \leq N \quad (6)$$

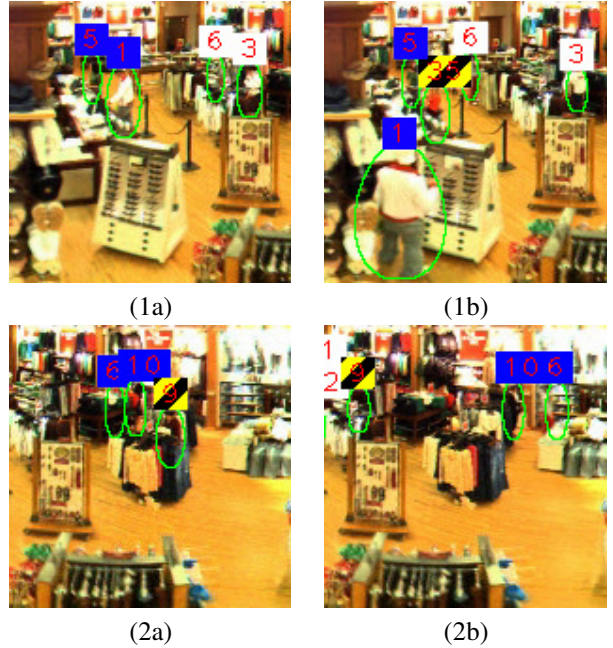


Figure 4: (1a) Two actors form a real shopper group (marked with solid blue) (1b) Of of the actors from 1a in coincidental event (marked with zebra yellow) (2a) Passer-by temporarily increases group cardinality to three (2b) Passer-by walking away

In equation 5 the first term minimizes the sum of square distances, thus reaching an optimal clustering. The second term minimizes the negative of the sum of the squares of cluster cardinalities, making stronger cluster attract more members. In iterative process, clusters with low cardinality values die out and parameter  $\alpha$  balances two terms to obtain the best solution with respect to number of clusters  $M$ . Note that clustering is completely unsupervised and only the value for alpha selected empirically.

To handle outliers a robust loss function  $\rho(d_{ij})$  is introduced, which measures the degree of typicality of point  $e_j$  with respect to cluster  $a_i$ . The function  $\rho$  is an  $M$ -estimator from robust statistics (M-estimators are a generalized form of maximum likelihood estimators (MLEs)) [6]. The function  $\psi$  is a corresponding  $W$ -estimator equivalent, in our algorithm, to weight function

$$\psi_{ij} = \psi(d_{ij}^2) = \delta\rho_i(d_{ij}^2)/\delta d_{ij}^2 \quad (7)$$

### 3 Results

To test our approach we used three tracking sequences recorded with the panoramic camera in apparel retail store over a period of one hour each. We have used human operator assisted shopping groups markup as ground truth. The total number of customers appearing in the scene in these three hours is 245 and the actual number of shopper groups is 50 with the total 112 customers in groups. Of these groups 7 were composed of three people and 2 of four people the rest being two-customer groups. We decided to exclude from consideration the groups formed earlier than 5 minutes prior to the sequence end as well as customers who were at the store at the beginning and left in the first 5 minutes of tracking, since the tracking information was less than 20% complete for such tracks. Several store employees were also excluded from the results.

The table 1 shows the total number of groups present in each scene (from the ground truth dataset), the percentages of correctly identified shopper groups as well as false positives (groups detected where none were present) and false negatives (missed groups).

Three sequences present one hour of typical store traffic on three different day taken from 4PM to 5PM. It is interesting to observe that the correct rates are higher for the first sequence, which also contained some of the heaviest traffic. We conclude that the performance accuracy of our group detector is proportional to the length of the tracks involved. Average store visit for a group can range anywhere from 5 to 15 minutes, which provided significant length of tracking data for our two stage clustering unit in most cases.

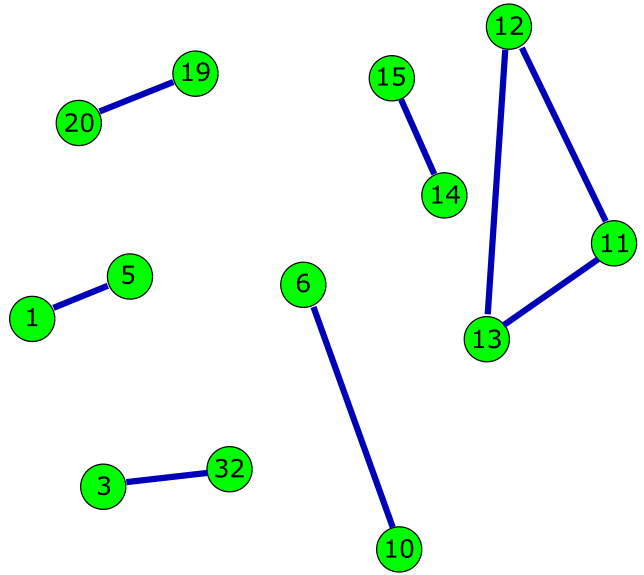


Figure 5: Shopping groups from sequence 2 (first 15 minutes) automatically arranged by proximity metric. The lengths of the edges connecting numbered nodes, correspond to the distance in non-Euclidean space calculated per equation 4

### 4 Discussion

In this paper we have demonstrated a two stage unsupervised shopper group detection algorithm. The algorithm feeds on tracking data, therefore reliability of the tracking system remains crucial. We have established that simple spatial grouping of actors is not enough for higher level group activity recognition as the groups might form coincidentally. Due to fuzzy temporal clustering of simple grouping events into higher-level behaviors our method provides an increased accuracy for group activity recognition (see Figure 4).

This work also present a generalized framework for detecting other types of group activities (multiple actors interacting with each other). Some examples include customers interacting with sales representatives, checkout line dynamics and group product browsing. By modifying distance metrics in equations 1, 2 and 4 and by using additional information about store layout and customer characteristics we can address these group behavior detection tasks with minimal changes to our setup.

### References

- [1] Aaron Bobick and Yuri Ivanov. Action recognition using probabilistic parsing. In *International Conference on Computer Vision and Pattern Recognition*, 1998.

Sequence	Groups	Correctly Detected (%)	False Positive (%)	False Negative (%)
1	20	90.0	5.0	10.0
2	16	87.5	6.2	12.5
3	14	78.5	14.2	21.5
Total	50	86.0	8.0	14.0

Table 1: Ground truth validation results

- [2] Francois Boutin and Mountaz Hascoet. Cluster validity indices for graph partitioning. In *International Conference on Information Visualisation*, 2004.
- [3] Dan Buzan, Stan Sclaroff, and George Kollios. Extraction and clustering of motion trajectories in video. In *International Conference on Pattern Recognition*, 2004.
- [4] Maurice Clerc. *Particle Swarm Optimization*. ISTE Publishing Company, 2006.
- [5] Hichem Frigui and Raghu Krishnapuram. A robust competitive clustering algorithm with applications in computer vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(5):450–465, May 1999.
- [6] Frank Hampel, Elvezio Ronchetti, Peter Rousseeuw, and Stahel Werner. *Robust Statistics: The Approach Based on Influence Functions (Wiley Series in Probability and Statistics)*. Wiley-Interscience, 2005.
- [7] Ismail Haritaoglu and Myron Flickner. Detection and tracking of shopping groups in stores. In *International Conference on Computer Vision and Pattern Recognition*, 2001.
- [8] Somboon Hongeng, Ram Nevatia, and Francois Bremond. Video-based event recognition: activity representation and probabilistic recognition methods. *Comput. Vis. Image Underst.*, 96(2):129–162, 2004.
- [9] Anil Jain and Richard Dubes. *Algorithms for Clustering Data*. Prentice-Hall, 1988.
- [10] Alex Leykin and Riad Hammoud. Robust multi-pedestrian tracking in thermal — visible surveillance videos. In *CVPRW*, 2006.
- [11] Alex Leykin and Mihran Tuceryan. A vision system for automated customer tracking for marketing analysis: Low level feature extraction. In *Human Activity Recognition and Modelling Workshop*, 2005.
- [12] Nuria Oliver, Barbara Rosario, and Alex Pentland. A bayesian computer vision system for modeling human interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):831–843, 2000.
- [13] Craig Reynolds. Steering behaviors for autonomous characters. In *Game Developers Conference*, 1999.
- [14] Barbara Rosario, Nuria Oliver, and Alex Pentland. A synthetic agent system for bayesian modeling of human interactions. In *AGENTS '99: Proceedings of the third annual conference on Autonomous Agents*, pages 342–343, New York, NY, USA, 1999. ACM Press.