Evaluation of Clustering Methods for Finding Dominant Optical Flow Fields in Crowded Scenes

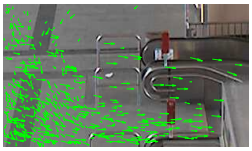
Günther Eibl, Norbert Brändle
Human Centered Mobility Technologies
arsenal research
A-1210 Vienna, Austria
{Guenther.Eibl | Norbert.Braendle }@arsenal.ac.at

Abstract

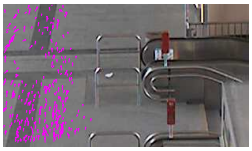
Video footage of real crowded scenes still poses severe challenges for automated surveillance. This paper evaluates clustering methods for finding independent dominant motion fields for an observation period based on a recently published real-time optical flow algorithm. We focus on self-tuning spectral clustering and Isomap combined with k-means. Several combinations of feature vector normalizations and distance measures (Euclidean, Mahanalobis and a general additive distance) are evaluated for four image sequences including three publicly available crowd datasets. Evaluation is based on mean accuracy obtained by comparison with a manually defined ground truth clustering. For every dataset at least one approach correctly classified more than 95% of the flow vectors without extra tuning of parameters, providing a basis for an automatic analysis after a view-dependent setup.

(a) Examples from a 1000 Frame Train Station Sequence

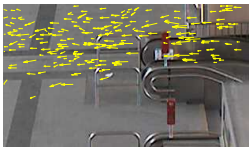
(b) Flow Vector Sample



(c) Flow Cluster 1



(d) Flow Cluster 2



(e) Flow Cluster 3

Figure 1. Clustering Dominant Optical Flow Fields

1. Introduction

Automated visual surveillance of scenes involving humans has a wide range of safety and security applications. A model of typical (dominant) people motions within an infrastructure is an important base for planners, for the detection of abnormal situations and can provide valuable input for realistic pedestrian simulation models. Automated analysis of crowded scenes involving many individuals still poses significant challenges for approaches based on detection and tracking of individual objects [1].

Global approaches for the analysis of dense groups of moving people are often based on optical flow analysis. For example, Figure 1a shows eight examples of a 1000 frame scene in front of an escalator of a train station. During the observation period one can identify several dominant flows of people, including the flow leaving from the escalator, leading to the escalator and a flow of people crossing the scene without using the escalator. Figure 1b shows a sample of optical flow vectors randomly drawn from dense frame-to-frame optical flow fields (see Section 2.1 for details). The objective is to cluster this set of flow vectors such that the vector clusters represent valid flow fields. Figures 1c-d show an automatic clustering of the vectors of Figure 1b into three groups. Note that there are spatial regions belonging to multiple clusters, like the upper left part of the field of view in Figure 1b. In this paper we compare results for several combinations of normalized features spaces, distance measures and clustering.

Wright and Pless [6] determine persistent motion patterns by a global joint distribution of independent local brightness gradient distributions and model this huge random variable – the number of components is equal to three dimensions times the number of pixels in the scene – with a

Gaussian mixture model. This approach assumes all motion in a frame to be coherent, independent motions like pedestrians moving independently violate these assumptions. Ali and Shah [1] present an approach inspired by particle dynamics, where they first determine spatial flow boundaries by advecting particles through the optical flow field and subsequently performing graph-cut based image segmentation. Their image sequences do not contain overlapping motions for a region like in Figure 1. Andrade et al. [2] use features based on linear PCA of optical flow vectors as input for a temporal model.

In this paper we evaluate two cluster approaches we have found promising for dealing with real flow data: self-tuning clustering [8] and the non-linear dimension reduction Isomap [4] followd by k-means. Apart from the sequence in Figure 1a we present evaluation results for three example of the publicly available UCF crowd data set [1].

2. Optical Flow Clustering

In the following we address (i) representation and normalization of the flow data, (ii) appropriate distance measures and (iii) clustering algorithms used for our evaluation.

2.1 Feature Vectors

The feature space $\mathcal{F}$ is a four-dimensional vector space with N feature vectors $\mathbf{f} = (x\ y\ u\ v)^T$, where $\mathbf{p} = (x\ y)^T$ are image location coordinates and $\mathbf{v} = (u\ v)^T$ are velocity vectors. We denote $R(\mathbf{v})$ as the magnitude of a vector and $\Theta = \angle(\mathbf{v}_i, \mathbf{v}_j)$ as the angle between any two vectors $\mathbf{v}_i$ and $\mathbf{v}_j$, with $1 \leq i, j \leq N$. The N feature vectors are randomly sampled from dense optical flow fields obtained for the observation period by a real-time variational approach recently published in [7]. Flow vectors with a speed below a minimum magnitude are not taken into account, with a scene-dependent threshold. Random sampling is performed for computational reasons such that the clustering time should only take a few seconds. Since the investigated video sequences are relatively short, we omit time information in the feature vectors.

We normalize image location coordinates x and y by subtracting the means and dividing by the standard deviations. The velocity components u and v are either normalized by subtracting the means and dividing by the standard deviations or not normalized. Both options are considered in the evaluation.

2.2 Distances and Similarities

We define three distance measures $D(i,j) := D(\mathbf{f}_i, \mathbf{f}_j)$ between any two feature vectors, with $1 \leq i, j \leq N$. The corresponding similarity $S(i,j)$ between two feature vectors is defined as $S(i,j) = \exp(-D^2(i,j)/s)$, where s is a local scale estimated from neighbors described in [8]. We consider the following distances:

1. *Squared Euclidean distance*

$$D^2_{\mathrm{E}}(i,j) = ||\mathbf{f}_i - \mathbf{f}_j||^2_2. \tag{1}$$

2. *Mahanalobis distance*

$$D^2_{\mathrm{M}}(i,j) = (\mathbf{f}_i - \mathbf{f}_j)\Sigma^{-1}(\mathbf{f}_i - \mathbf{f}_j)^T, \tag{2}$$

   where Σ is the covariance matrix between the components of the feature vectors. (2) takes into account the correlation between the features and has the appealing property that the coordinates are already intrinsically normalized.

3. *Weighted additive distances*

$$D^2_{\mathrm{A}}(i,j) = w_{\mathbf{p}} D^2_{\mathbf{p}} + w_\Theta D^2_\Theta + w_R D^2_R + w_\beta D^2_\beta, \tag{3}$$

   with weights $w_* \geq 0$ and $D_{\mathbf{p}}$ as the Euclidean distance between two flow locations, D_Θ as the angle between flow directions, D_R as the speed difference and D_β the angle β between the mean of the two velocity vectors and the straight line through the flow vector locations. Note that the last term directly couples location and velocity. It is motivated by the fact that if two flow vectors arise from a smooth movement of the same object, one of the two velocity vectors should point to the location of the other one.

The distances in (3) are additive such that the corresponding similarity can be written as a product of the individual similarity terms. This is in contrast to [3], where the *similarities* are considered as additive. Preliminary trials with additive similarities have lead to poor results for additive similarities, they are therefore not considered here. As for setting weights w_* in (3), we are interested in the combined behaviour of normalization, distance metric and clustering algorithm and not in (over)tuning parameter values to particular datasets. The weights are determined such that all products $w_* D^2_*$ have the same expected value, i.e. all terms in (3) should be approximately equally important. The weights $w_{\mathbf{p}}$ and w_R can be optionally set to 0. Assuming independent normalized location coordinates x and y, the expected value for the squared distance is

$$E\left(D^2_{\mathbf{p}}\right) = E\left((x_1 - x_2)^2 + (y_1 - y_2)^2\right) = 4E\left(x_1^2\right) = 4.$$

The remaining weights are set in order to match this expected squared distance: Assuming uniformly distributed angles between velocity vectors $\Theta \sim U(0, \pi)$, we get $E\left(D^2_\Theta\right) = \pi^2/3$, so $E\left(w_\Theta D^2_\Theta\right) = E\left(D^2_{\mathbf{p}}\right)$ for $w_\Theta = 12/\pi^2$. Assuming $\beta \sim U(0, \frac{\pi}{2})$, an analogue calculation leads to $w_\beta = 12/(\pi/2)^2 = 48/\pi^2$.

2.3 Clustering

The (normalized) feature vectors combined with different distance/similarity measures as described above serve as input for two clustering methods:

1. Self-tuning spectral clustering (SC) as described in [8] using similarities and a local scale.
2. Nonlinear dimension reduction Isomap [4] (ISO) with subsequent k-means clustering in the resulting subspace.

3. Clustering Evaluation

We first describe the data sets and the evaluation criterium and then describe the evaluation of different combinations of normalization, distance measures and clusterings.

3.1 Evaluation Criterion

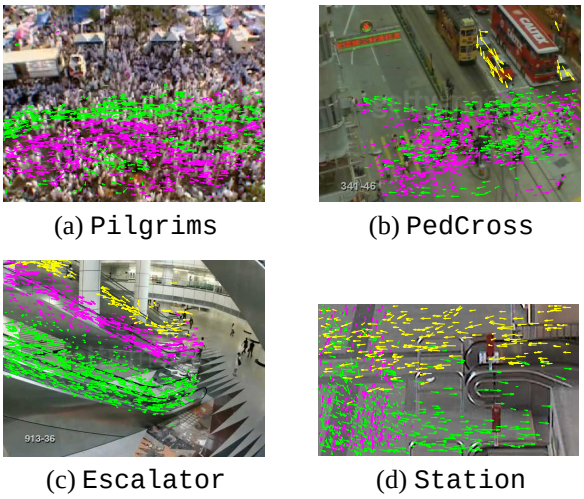
Figure 2 shows the four data sets with superimposed 'ground truth' flow clusters which are defined by simple rules with regard to flow direction and location. Figures 2a-c are taken from the UCF crowd data set [1]. The `Pilgrims` and `PedCross` sequences contain two spatially mixed flows in opposite directions, where `PedCross` has an additional flow cluster stemming from a car arriving at the crossing. The `Escalator` scene has three flow clusters corresponding to the three escalators. For regions on the floor no ground truth is defined.

For all four datasets the sample size is set to $N = 1000$ flow vectors. The N flow vectors are then clustered into the predefined number of flow fields, the permutation of the cluster labels maximizing the coincidences with the true labels is determined and the percentage of correctly assigned labels is reported. The accuracy measure should also reflect the stability with regard to different random samples, hence the accuracy over ten different random samples is determined as the mean average.

One important isssue is how to automatically chose a suitable number of clusters. Figure 3 shows the criterion function of [8] for cluster numbers between 2 and 15, where each plot corresponds to a different flow vector sample. Since no clear maximum can be identified in Figure 3 (the cluster numbers marked by red circles should be best suited for the data set), we have fixed the number of clusters. ISOMAP turned out to be insensitive with respect to the embedding dimension of Isomap, we therefore set the dimension of the embedding space to 4.

3.2 Results

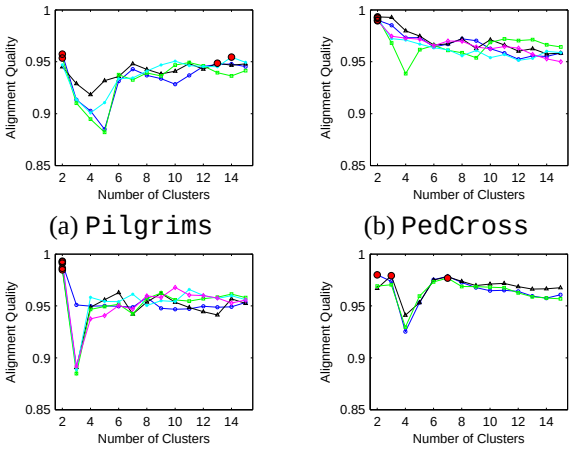
Figure 4 shows a qualitative comparison of spectral clustering with Mahanalobis distance with the final flow segmentation results for the `Pilgrims` sequence published in [1].



(a) `Pilgrims` (b) `PedCross`

(c) `Escalator` (d) `Station`

Figure 2. Test Sequences with 'Ground Truth'



(a) `Pilgrims` (b) `PedCross`

(c) `Escalator` (d) `Station`

Figure 3. Model Selection Criterion Function [8] for Different Vector Samples

In general, our results better retain the counter-flows within larger flows detected by the optical flow algorithm.

Table 1 lists the mean accuracies of clustering with the Euclidean distance (1) and the Mahalanobis distance (2). Table 2 shows the mean accuracies found using variants of the additive distance (3).

For all four datasets a combination with mean accuracy greater than 95% over all random samples could be found without extra tuning of parameters for a particular dataset.

A detailed comparison of the individual combinations shows no surprising results. The choice of the distance measure is the most important factor. As for the additive distance (3), including the flow angle is crucial for achieving good results. The accuracies were also compared with ordinary k-means clustering (KM), which underperformes compared to SC and ISO. ISO and SC show similar performance, but for a given dataset and distance measure the

Table 2. Mean Accuracies for General Additive Distance (3)

Normalization of $\mathbf{v}$	w_R	$w_\mathbf{p}$	w_β	w_Θ	Pilgrims		PedCross		Station		Escalator	
					SC	ISO	SC	ISO	SC	ISO	SC	ISO
y	1	0	0	$12/\pi^2$	96.4	94.9	79.8	97.3	92.9	98.0	55.1	62.4
y	1	0	$48/\pi^2$	$12/\pi^2$	96.1	96.7	86.3	97.2	97.8	90.3	59.0	60.5
y	1	1	0	$12/\pi^2$	94.2	94.2	93.7	98.9	97.8	95.8	66.1	73.0
y	1	1	$48/\pi^2$	$12/\pi^2$	88.6	91.0	90.4	98.0	96.6	95.6	66.4	73.9
n	0	0	0	$12/\pi^2$	**99.5**	90.8	99.1	90.5	96.0	87.6	66.6	65.2
y	0	0	0	$12/\pi^2$	99.2	92.2	94.2	92.5	96.7	80.7	46.2	47.2
n	0	0	$48/\pi^2$	$12/\pi^2$	95.3	**96.8**	72.9	**99.6**	**98.4**	**98.5**	85.3	77.4
y	0	0	$48/\pi^2$	$12/\pi^2$	94.0	**96.8**	74.3	89.4	97.7	97.6	54.4	48.6
n	0	1	0	$12/\pi^2$	92.8	90.0	90.4	96.5	98.2	97.0	60.2	72.2
y	0	1	0	$12/\pi^2$	89.0	93.4	**99.4**	98.4	97.6	96.8	58.6	66.8
n	0	1	$48/\pi^2$	$12/\pi^2$	89.7	91.7	72.8	75.8	97.5	96.8	**91.5**	**95.8**
y	0	1	$48/\pi^2$	$12/\pi^2$	88.1	91.5	90.2	82.8	96.2	95.1	56.3	63.7

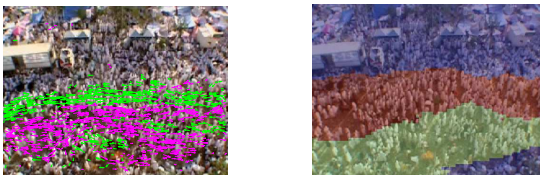
Table 1. Mean Accuracies for Euclidean and Mahalanobis Distance

Dataset	Distance	KM	SC	ISO
Pilgrims	D_E^2	92.6	88.4	95.1
	D_M^2	**94,4**	**98,3**	**95,1**
PedCross	D_E^2	55.1	**95.8**	**91.8**
	D_M^2	**80.1**	80.3	90.1
Station	D_E^2	91.1	**97.7**	**95.2**
	D_M^2	**92.1**	87.5	96.3
Escalator	D_E^2	**54.5**	**64.8**	84.2
	D_M^2	47.5	57.2	**97.5**

difference between the two methods can be big as can be seen for the PedCross dataset and the additive distance. Velocity normalization plays a minor role for these datasets. Given a particular dataset several approaches should be investigated because the best combination strongly depends on the dataset.

4. Discussion

It is obvious that the 'ground truth' clusters based on simple rules as shown in Figure 2 are not perfect. Furthermore, choosing a suitable similarity measure will certainly have a strong impact on the subsequent clustering approach. For the examined data sets, at least one combination can be found having a mean accuracy higher than 95% over all random samples of optical flow vectors, without tuning parameters for a particular dataset. Execution time is less than 5 seconds on a MATLAB implementation. For a real-time surveillance system with static camera views, different clusterings might be proposed to operators during a short setup time, and the most plausible model suitable for the particular camera perspective and infrastructuere could then be subsequently used. In this sense, we plan to add background knowledge in a constrained clustering setup in the spirit of [5].



(a) SC with Mahanalobis Distance (b) Crowd Segmentation of [1]

Figure 4. Qualitative Comparison

5. Acknowledgements

This work has been partially funded by the Austrian FIT-IT Visual Computing Program under Grant No. 813395/12441. We thank Horst Bischof of ICG TU Graz for providing the GPU-Based Optical Flow Software.

References

[1] S. Ali and M. Shah. A Lagrangian Particle Dynamics Approach for Crowd Flow Simulation and Stability Analysis. In *Proceedings CVPR*, 2007.

[2] E. Andrade, S. Blunsden, and R. Fisher. Modelling Crowd Scenes for Event Detection. In *Proceedings ICPR*, 2006.

[3] H. Li and I.-F. Shen. Similarity Measure for Vector Field Learning. In *Advances in Neural Networks - ISNN (1)*, pages 436–441. Springer Berlin / Heidelberg, 2006.

[4] J. Tenenbaum, V. de Silva, and J. Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290:2319–2323, 2000.

[5] K. Wagstaff, C. Cardie, S. Rogers, and S. Schrödl. Constrained K-means Clustering with Background Knowledge. In *Proceeding 8th Intl. Conf. on Machine Learning*, 2001.

[6] J. Wright and R. Pless. Analysis of Persistent Motion Patterns Using the 3D Structure Tensor. In *WACV/MOTION*, 2005.

[7] C. Zach, T. Pock, and H. Bischof. A Duality Based Approach for Realtime TV-L^1 Optical Flow. In *Proc. 29th DAGM Symposium on Pattern Recognition*, 2007.

[8] L. Zelnik-Manor and P. Perona. Self-Tuning Spectral Clustering. In *Adv. Neural Inf. Process. Syst.*, pages 1601–1608, 2004.