

# Relevance Learning for Spectral Clustering with Applications on Image Segmentation and Video Behaviour Profiling

Tao Xiang and Shaogang Gong  
Department of Computer Science  
Queen Mary, University of London, London E1 4NS, UK  
{sgg, txiang}@dcs.qmul.ac.uk

## Abstract

*We aim to tackle the problem of unsupervised visual learning. A novel relevance learning algorithm is proposed for data clustering using eigenvectors of a data affinity matrix. We show that it is critical to select the relevant eigenvectors for both estimating the optimal number of clusters and performing data clustering especially given noisy and sparse data. The effectiveness of our algorithm are demonstrated on solving two challenging visual data clustering problems: image segmentation and video behaviour profiling.*

## 1. Introduction

Spectral clustering has been widely used for unsupervised visual learning [10, 12, 9, 7, 8]. Although there have been extensive studies on spectral clustering, two critical issues remain largely unsolved: (1) How to automatically determine the number of clusters? and (2) How to perform effective clustering given noisy and sparse data. Regarding the first issue, most previous work assumed that the number of clusters is known or has been manually set. To our knowledge, only two approaches have been proposed in the literature to automatically determine the cluster number. Smyth [9] proposed to use a Monte-Carlo cross validation approach to determine the number of clusters for sequences modelled using Hidden Markov Models (HMMs). This approach is computationally expensive and thus not suitable for large data sets common to applications such as image segmentation. Porikli and Haga [8] employed a validity score computed using the largest eigenvectors (i.e. eigenvectors corresponding to the largest eigenvalues) of a data affinity matrix to determine the number of clusters for video-based activity classification. However, their approach does not take into account the inevitable presence of noise in a realistic data set and thus is error prone especially when the sample size is small. There is no published work that has tackled the second issue explicitly.

We propose a novel spectral clustering algorithm based on learning the relevance of eigenvectors of a data affinity matrix for data clustering. Instead of using the eigenvectors indiscriminately, the relevance of each eigenvector is measured according to how well it separates one group of data

samples from others. To minimise the influence of noise on the distributions of eigenvectors and reduce the dimensionality of the affinity matrix eigenspace, the irrelevant eigenvectors are removed and the relevant ones are weighted using the learned relevance measure for data clustering. Unlike previous unsupervised feature relevance learning algorithms such as [4, 2], our algorithm is specially tailored for fast and robust selection of the relevant eigenvectors of an affinity matrix based on the *a priori* knowledge on the data distribution in the eigenspace. We demonstrate the effectiveness of our algorithm through experiments on image segmentation and video behaviour profiling.

## 2. Learning Eigenvector Relevance

Given a set of feature vectors  $\mathbf{D} = \{\mathbf{f}_1, \dots, \mathbf{f}_n, \dots, \mathbf{f}_N\}$ , we aim to estimate the optimal number of clusters  $K_o$  for best describing the underlying distribution of the data set. Note that different feature vectors can be of different dimensionality. An  $N \times N$  affinity matrix  $\mathbf{A} = \{A_{ij}\}$  can be formed whose element  $A_{ij}$  measures the distance/dissimilarity between the  $i$ th and  $j$ th feature vectors. The eigenvectors of  $\mathbf{A}$  can be employed directly for clustering. However, it has been shown in [10] that it is more desirable to perform clustering based on the eigenvectors of the normalised affinity matrix  $\bar{\mathbf{A}}$ , defined as  $\bar{\mathbf{A}} = \mathbf{L}^{-\frac{1}{2}} \mathbf{A} \mathbf{L}^{-\frac{1}{2}}$  where  $\mathbf{L}$  is an  $N \times N$  diagonal matrix with  $L_{ii} = \sum_j A_{ij}$ . It has also been shown in [12, 10] that the largest  $K$  eigenvectors of  $\bar{\mathbf{A}}$  are sufficient to partition the data set into  $K$  clusters.

We assume that the number of clusters  $K$  is between 1 and  $K_m$ , a number considered to be sufficiently larger than  $K_o$ . The training data set is then represented using the  $K_m$  largest eigenvectors of  $\bar{\mathbf{A}}$ , denoted as  $\mathbf{D}_e = \{\mathbf{x}_1, \dots, \mathbf{x}_n, \dots, \mathbf{x}_N\}$ , with the  $n$ th feature vector  $\mathbf{f}_n$  being represented as a  $K_m$  dimensional vector  $\mathbf{x}_n = [e_{1n}, \dots, e_{kn}, \dots, e_{K_m n}]$ , where  $e_{kn}$  is the  $n$ th element of the  $k$ th largest eigenvector  $\mathbf{e}_k$ .

Because only the first  $K_o$  largest eigenvectors are needed for grouping  $K_o$  clusters, there are certainly redundant/irrelevant eigenvectors among the  $K_m$  largest eigenvectors. It is important to identify and remove those irrelevant but large eigenvectors because that (1) irrelevant

features degrade the accuracy of learning, and (2) the dimensionality of the features space ( $K_m$ ) can be high compared to the sample size ( $N$ ) resulting in learning subject to the curse of dimensionality. To overcome these problems, we derive here a novel eigenvector relevance learning algorithm based on the following observations of the affinity matrix eigenspace: (1) A small eigenvector is less likely to be relevant in data clustering than a large one, and (2) Each of the relevant eigenvectors of the normalised affinity matrix is able to separate one cluster from others while those irrelevant ones are not [10]. Specifically, we proposed to measure the relevance of an eigenvector according to how well it can separate a data set into different clusters.

We denote the likelihood of the  $k$ th largest eigenvector  $\mathbf{e}_k$  being relevant as  $R_{\mathbf{e}_k}$ . Apparently, we have  $0 \leq R_{\mathbf{e}_k} \leq 1$ . We assume that the elements of  $\mathbf{e}_k$ ,  $e_{kn}$  follow two different distributions depending on whether  $\mathbf{e}_k$  is relevant. The probability density function (pdf) of  $e_{kn}$  is thus formulated as a mixture model of two components:

$$p(e_{kn}|\theta_{e_{kn}}) = (1 - R_{\mathbf{e}_k})p(e_{kn}|\theta_{e_{kn}}^1) + R_{\mathbf{e}_k}p(e_{kn}|\theta_{e_{kn}}^2)$$

where  $\theta_{e_{kn}}$  are the parameters describing the distribution,  $p(e_{kn}|\theta_{e_{kn}}^1)$  is the pdf of  $e_{kn}$  when  $\mathbf{e}_k$  is irrelevant/redundant and  $P(e_{kn}|\theta_{e_{kn}}^2)$  otherwise.  $R_{\mathbf{e}_k}$  acts as the weight or mixing probability of the second mixture components. The distribution of  $e_{kn}$  is assumed to be a single Gaussian to reflect the fact that  $\mathbf{e}_k$  cannot be used for data clustering when it is irrelevant:

$$p(e_{kn}|\theta_{e_{kn}}^1) = \mathcal{N}(e_{kn}|\mu_{k1}, \sigma_{k1})$$

where  $\mathcal{N}(\cdot|\mu, \sigma)$  denotes a Gaussian of mean  $\mu$  and covariance  $\sigma$ . We assume the second component of  $P(\mathbf{e}_k|\theta_{\mathbf{e}_k})$  as a mixture of two Gaussians to reflect the fact  $\mathbf{e}_k$  can separate one group of data from others when it is relevant:

$$p(e_{kn}|\theta_{e_{kn}}^2) = w_k \mathcal{N}(e_{kn}|\mu_{k2}, \sigma_{k2}) + (1-w_k) \mathcal{N}(e_{kn}|\mu_{k3}, \sigma_{k3})$$

where  $w_k$  is the weight of the first Gaussian in  $p(e_{kn}|\theta_{e_{kn}}^2)$ . There are 8 parameters required for describing the distribution of  $e_{kn}$ :  $\theta_{\mathbf{e}_k} = \{R_{\mathbf{e}_k}, \mu_{k1}, \mu_{k2}, \mu_{k3}, \sigma_{k1}, \sigma_{k2}, \sigma_{k3}, w_k\}$ . The maximum likelihood (ML) estimate of  $\theta_{\mathbf{e}_k}$  can be obtained using the following algorithm. First, the parameters of the first mixture component  $\theta_{e_{kn}}^1$  are estimated as  $\mu_{k1} = \frac{1}{N} \sum_{n=1}^N e_{kn}$  and  $\sigma_{k1} = \frac{1}{N} \sum_{n=1}^N (e_{kn} - \mu_{k1})^2$ . The rest 6 parameters are then estimated iteratively using Expectation Maximisation (EM) [1].

Since our relevance learning algorithm is essentially a local (greedy) searching method, the algorithm could be sensitive to parameter initialisation especially given noisy and sparse data [1]. To overcome this problem, our *a priori* knowledge on the relevance of each eigenvector can be utilised to set the initial value of  $R_{\mathbf{e}_k}$ . Specifically, we set  $\tilde{R}_{\mathbf{e}_k} = \bar{\lambda}_k$ , where  $\tilde{R}_{\mathbf{e}_k}$  is the initial value of  $R_{\mathbf{e}_k}$  and  $\bar{\lambda}_k \in [0, 1]$  is the normalised eigenvalue for  $\mathbf{e}_k$  with  $\bar{\lambda}_1 = 1$  and  $\bar{\lambda}_{K_m} = 0$ .

The estimated  $R_{\mathbf{e}_k}$  provides a continuous-value measurement of the relevance of  $\mathbf{e}_k$ . Since a ‘hard-decision’ is needed for dimension reduction, we simply eliminate the  $k$ th eigenvector  $\mathbf{e}_k$  among the  $K_m$  candidate eigenvectors if  $R_{\mathbf{e}_k} < 0.5$  and weight the relevant eigenvectors using  $R_{\mathbf{e}_k}$ . This gives us a new data set denoted as  $\mathbf{D}_r = \{\mathbf{y}_1, \dots, \mathbf{y}_n, \dots, \mathbf{y}_N\}$ . We model the distribution of  $\mathbf{D}_r$  using a Gaussian Mixture Model (GMM) for data clustering. The Bayesian Information Criterion (BIC) is then employed together with the learning of the GMM using EM to select the optimal number of components, corresponding to the optimal number of clusters  $K_o$ . Each feature vector in the training data set is then labelled as one of the  $K_o$  clusters using the learned GMM with  $K_o$  Gaussian components.

### 3. Image Segmentation

Our relevance learning based spectral clustering algorithm has been applied to image segmentation. A pixel-pixel pairwise affinity matrix  $A$  is constructed for an image based on the Intervening Contours method introduced in [6]. First, for the  $i$ th pixel on the image the magnitude of the orientation energy along the dominant orientation is computed as  $OE(i)$  using oriented filter pairs [6]. The value of  $OE(i)$  ranges between 0 and infinity. A probability-like variable  $p_{con}$  is then computed as  $p_{con} = 1 - \exp(-OE(i)/\sigma)$ . The value of  $\sigma$  is related to the noise level of the image. It is set as 0.02 in this paper. The value of  $p_{con}$  is close to 1 when the orientation energy is much greater than the noise level, indicating the presence of a strong edge. Second, given any pair of pixels in the image, the pixel affinity is computed as

$$A_{ij} = 1 - \max_{x \in M_{ij}} P_{con}(x)$$

where  $M_{ij}$  are those local maxima along the line connecting pixels  $i$  and  $j$ . The dissimilarity between pixels  $i$  and  $j$  is high ( $A_{ij}$  is low) if the orientation energy along the line between the two pixels is strong (i.e. the two pixels are on the different sides of a strong edge).

Our algorithm has been tested on a variety of natural images. Fig. 1&2 show typical segmentation results.  $K_m$  was set to 20 in our experiments. Our results show that (1) Regions corresponding to objects or object parts are clearly separated from each other, and (2) The optimal numbers of image segments estimated by our algorithm reflect the complexity of the images very well. The results verify our argument that selecting the relevant eigenvectors is critical for spectral clustering. In particular, it is discovered by our experiments that performing spectral clustering without relevance learning caused the number of image segments being severely underestimated. A typical image segmentation process is shown in details in Fig. 1. Fig. 1(k) shows the learned relevance for each of the largest 20 eigenvectors for an image shown in Fig. 1(a). The most relevant and irrelevant eigenvectors, shown in Fig. 1(b)-(e) and Fig. 1(f)-(i)

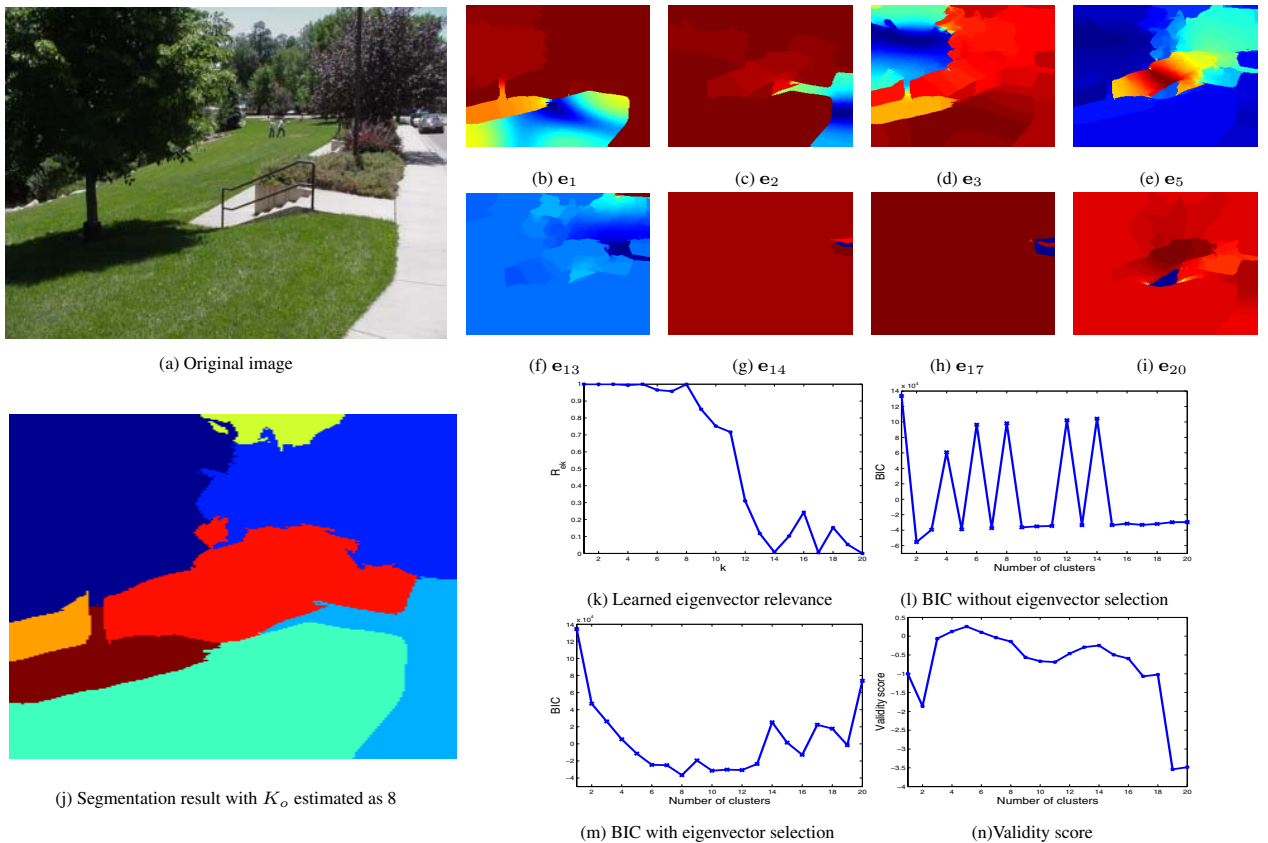


Figure 1: An image shown in (a) is segmented as shown in (j). The learned relevance for the 20 largest eigenvectors is shown in (k). (b)-(e) and (f)-(i) show the top 4 most relevant and irrelevant eigenvectors among the 20 largest eigenvectors respectively. (l) and (m) show that  $K_o$  was estimated as 2 and 8 without and with relevant eigenvector selection respectively using BIC. (n) shows that  $K_o$  was estimated as 5 using the validity score [8].

respectively, demonstrate that our relevance learning algorithm is able to select those eigenvectors that are most capable of grouping pixels into meaningful regions. The number of segments in the image was determined as 8 by BIC using only the relevant eigenvectors (see Fig. 1(m)). It was determined as 2 and 5 by BIC and the validity score [8] respectively without relevance learning (see Fig. 1(l)&(n)).

#### 4. Video Based Behaviour Profiling

Our spectral clustering algorithm has also been applied to video behaviour profiling which can be considered as a more challenging problem compared to image segmentation due to the often very sparse and noisy data available. A continuous video  $\mathbf{V}$  is segmented into  $N$  segments  $\mathbf{V} = \{\mathbf{v}_1, \dots, \mathbf{v}_n, \dots, \mathbf{v}_N\}$  so that each segment contains approximately a single behaviour pattern. Video segmentation can be performed by either utilising non-activity gaps between two consecutive behaviour patterns for a not-too-busy scenario, or employing an on-line segmentation algorithm [11] for a busy scenario. Alternatively, the video can also be simply sliced into overlapping segments with a fixed

time duration [13]. A discrete event based approach is then adopted for behaviour representation [3]. First, an adaptive Gaussian mixture background model is used to detect foreground pixels. Second, the foreground pixels in a vicinity are grouped into a blob using the connected component method. Each blob with its average pixel-change-history value greater than a threshold is then defined as an event. An event is represented as a 7-dimensional feature vector capturing location, shape and motion information. Third, classification is performed in a 7D feature space using a Gaussian Mixture Model (GMM). The number of event classes  $K_e$  is determined automatically using BIC. The learned GMM is used to classify each detected event into one of  $K_e$  event classes. Finally, the behaviour pattern captured by the  $n$ th video segment  $\mathbf{v}_n$ , consisting of  $T_n$  image frames, is represented as a behaviour pattern feature vector  $\mathbf{P}_n = [\mathbf{p}_{n1}, \dots, \mathbf{p}_{nt}, \dots, \mathbf{p}_{nT_n}]$ , where the  $t$ th element  $\mathbf{p}_{nt}$  is a  $K_e$  dimensional variable:  $\mathbf{p}_{nt} = [p_{nt}^1, \dots, p_{nt}^k, \dots, p_{nt}^{K_e}]$ ;  $\mathbf{p}_{nt}$  is computed from the  $t$ th image frame of  $\mathbf{v}_n$  where  $p_{nt}^k$  is the posterior probability that an event of the  $k$ th event



Figure 2: Examples of image segmentation. Left: original images; Right: segmentation results. From top to bottom, the optimal number of segments  $K_o$  were automatically determined as 7, 7, 8, 5, 9, and 4 respectively.

class has occurred in the frame given the learned GMM.

Consider a training data set  $\mathbf{D} = \{\mathbf{P}_1, \dots, \mathbf{P}_n, \dots, \mathbf{P}_N\}$  consisting of  $N$  behaviour patterns, where  $\mathbf{P}_n$  is the  $n$ th behaviour pattern feature vector as defined above. We aim to first discover the natural grouping of the training behaviour patterns upon

which a behaviour model can be based. This is an unsupervised clustering problem with the number of clusters unknown. However, there are two characteristics of the behaviour feature vectors that make the clustering problem challenging: (1) Each feature vector can be of different length therefore requires dynamic warping before they can be compared with. (2) A definition of a distance/affinity metric among these variable length feature vectors is not simply Euclidean therefore requires a nontrivial string similarity measure.

We propose to utilise Dynamic Bayesian Networks (DBNs) to provide a dynamic representation of each behaviour pattern feature vector in order to both address the need for dynamic warping and provide a string similarity metric. More specifically, each behaviour pattern in the training set is modelled using a DBN. To measure the affinity between two behaviour patterns represented as  $\mathbf{P}_i$  and  $\mathbf{P}_j$ , two DBNs denoted as  $\mathbf{B}_i$  and  $\mathbf{B}_j$  are trained on  $\mathbf{P}_i$  and  $\mathbf{P}_j$  respectively using the EM algorithm [1, 5]. The affinity between  $\mathbf{P}_i$  and  $\mathbf{P}_j$  is then computed as:

$$A_{ij} = \frac{1}{2} \left\{ \frac{1}{T_j} \log P(\mathbf{P}_j | \mathbf{B}_i) + \frac{1}{T_i} \log P(\mathbf{P}_i | \mathbf{B}_j) \right\}, \quad (1)$$

where  $P(\mathbf{P}_j | \mathbf{B}_i)$  is the likelihood of observing  $\mathbf{P}_j$  given  $\mathbf{B}_i$ , and  $T_i$  and  $T_j$  are the lengths of  $\mathbf{P}_i$  and  $\mathbf{P}_j$  respectively. DBNs of different topologies can be used. In this work, we employ a Multi-Observation Hidden Markov Model (MOHMM) [3]. The number of hidden states for each hidden variables in the MOHMM is set to  $K_e$ , i.e. the number of event classes <sup>1</sup>.

C1	From the office area to the near end of the corridor
C2	From the near end of the corridor to the office area
C3	From the office area to the side-doors
C4	From the side-doors to the office area
C5	From the near end of the corridor to the side-doors
C6	From the side-doors to the near end of the corridor

Table 1: Six classes of commonly occurred behaviour patterns in the entrance scene.

Using our spectral clustering algorithm described in Section 2, the  $N$  behaviour patterns in the training set are classified into  $K_o$  behaviour pattern classes. To build a model for the observed/expected behaviour, we first model the  $k$ th behaviour class using a MOHMM  $\mathbf{B}_k$ . The parameters of  $\mathbf{B}_k$ ,  $\theta_{\mathbf{B}_k}$  are estimated using all the patterns in the training set that belong to the  $k$ th class. A behaviour model  $\mathbf{M}$  is then formulated as a mixture of the  $K_o$  MOHMMs. Given an unseen behaviour pattern, represented as a behaviour pattern feature vector  $\mathbf{P}$ , the likelihood of observing  $\mathbf{P}$  given  $\mathbf{M}$  is  $P(\mathbf{P} | \mathbf{M}) = \sum_{k=1}^{K_o} \frac{N_k}{N} P(\mathbf{P} | \mathbf{B}_k)$ , where  $N_k$  is the number of training behaviour patterns that belong to the  $k$ th behaviour class. An unseen behaviour pattern is detected as

<sup>1</sup>  $K_e$  reflects the complexity of the behaviour patterns, so is the number of hidden states. So it is appropriate to set these two to be equal.

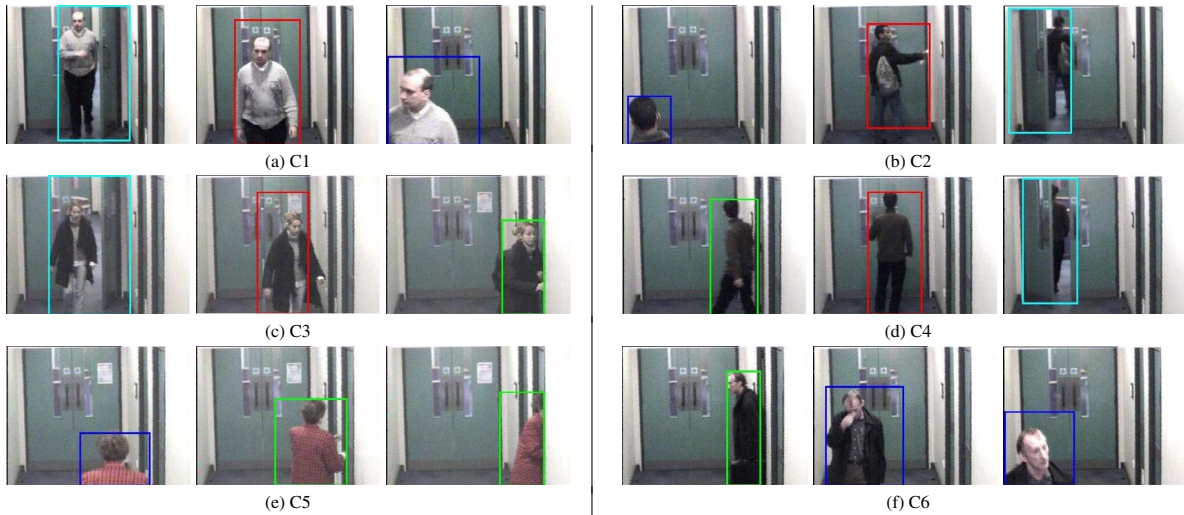


Figure 3: Behaviour patterns in the entrance scene. (a)–(f) show image frames of typical behaviour patterns belonging to the 6 behaviour classes listed in Table 1. Events detected during each behaviour pattern are shown by colour-coded bounding boxes in each frame.

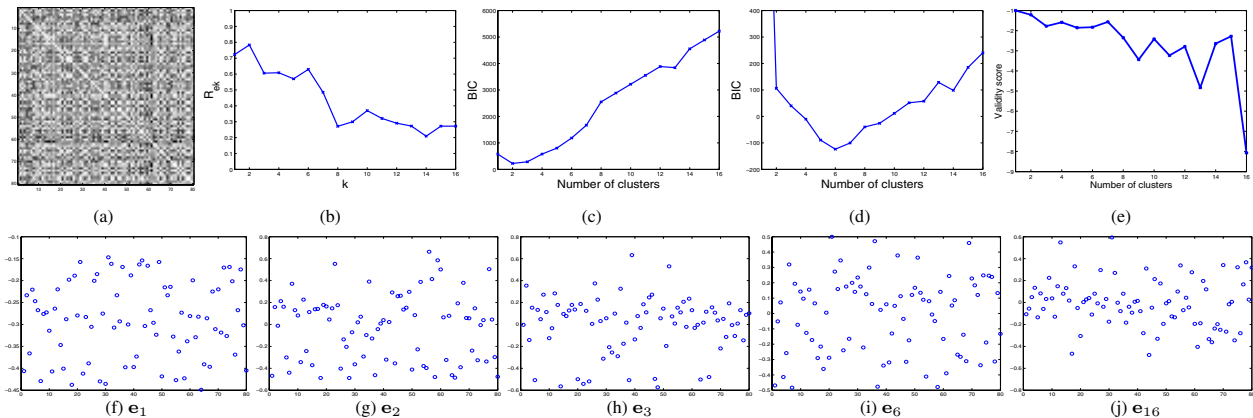


Figure 4: An typical example of model training. (a): The normalised behaviour affinity. (b): the learned relevance for the  $K_m$  largest eigenvectors. (c) and (d) show the BIC model selection results without and with relevant eigenvector selection respectively. (e) shows that  $K_o$  was estimated as 1 using the validity score. (f)–(j): the distributions of some eigenvectors.

abnormal if  $P(\mathbf{P}|\mathbf{M}) < Th_A$  where  $Th_A$  is a threshold. When an unseen behaviour pattern is detected as normal, the normal behaviour model  $\mathbf{M}$  can also be used for recognising it as one of the  $K$  behaviour pattern classes learned from the training set. More specifically, an unseen behaviour pattern is assigned to the  $\hat{k}$ th behaviour class when  $\hat{k} = \arg \max_k \{P(\mathbf{P}|\mathbf{B}_k)\}$ .

Experiments were conducted on an entrance surveillance scenario. A CCTV camera was mounted on the ceiling of an office entry corridor, monitoring people entering and leaving the office area (see Fig. 3). The office area is secured by an entrance-door which can only be opened by scanning an entry card on the wall next to the door (see middle frame in Fig. 3(b)). Two side-doors were also located at the right hand side of the corridor. Typical behaviours occurring in

the scene would be people entering or leaving either the office area or the side-doors, and walking towards the camera. Each behaviour pattern would normally last a few seconds. For this experiment, a data set was collected over 5 different days consisting of 6 hours of video totalling 432000 frames captured at 20Hz with  $320 \times 240$  pixels per frame. This data set was then segmented into sections separated by any motionless intervals lasting for more than 30 frames. This resulted in 142 video segments of actual behaviour pattern instances. Each segment has on average 121 frames with shortest 42 and longest 394.

**Model training** — Each training set consist of 80 randomly selected video segments without any behaviour class labelling of the video segments. The remaining 62 segments were used for testing later. This model training exercise was



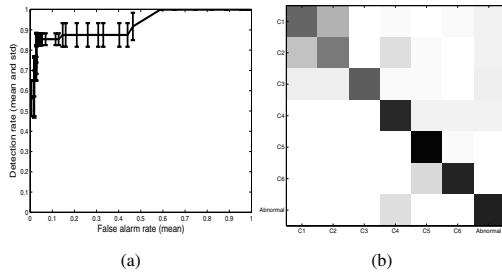


Figure 5: The performance of abnormality detection and behaviour recognition for the corridor scene. (a): The mean and  $\pm 1$  standard deviation of the ROC curves for abnormality detection obtained over 20 trials. (b): Confusion matrix for behaviour recognition. Each row represents the probabilities of that class being confused with all the other classes averaged over 20 trials. The main diagonal of the matrix shows the the fraction of patterns correctly recognised and is as follows: [.68 .63 .72 .84 .92 .85].

repeated 20 times and in each trial a different model was trained using a different random training set. Given each training set, 4 classes of discrete events were detected and classified using automatic model order selection in clustering (see Figure 3). Over the 20 trials, on average 6 eigenvectors were automatically determined as being relevant for clustering with smallest 4 and largest 9.  $K_m$  was set to 16 in the experiments. Fig. 4 shows an typical example of the model training. The number of clusters for each training set was determined automatically as 2 and 6 in every trial without and with relevant eigenvector selection respectively (see Fig. 4(c)&(d)). In the meantime, the number of clusters was severely under-estimated using the validity score (see Fig. 4(e)). By observation, each discovered data cluster mainly contained samples corresponding to one of the 6 behaviour classes listed in Table 1.

**Abnormality detection** — To measure the performance of the learned models on abnormality detection, each behaviour pattern in the testing sets was manually labelled as normal if there were similar patterns in the corresponding training sets and abnormal otherwise. On average, there were 7 abnormal behaviour patterns in each testing set. The detection rate and false alarm rate of abnormality detection are shown in the form of a ROC curve. Fig. 5(a) shows that high detection rate and low false alarm rate can be achieved.  $Th_A$  was set to  $-0.2$  in the rest results unless otherwise specified, which gave an abnormality detection rate of  $85.4 \pm 2.9\%$  and false alarm rate of  $6.1 \pm 3.1\%$ .

**Recognition of normal behaviours** — To measure the performance of behaviour recognition results, the normal behaviour patterns in the testing sets were manually labelled into different behaviour classes. A normal behaviour pattern was recognised correctly if it was detected as normal and classified into the right behaviour class. The behav-

our recognition results is illustrated as a confusion matrix shown in Fig. 5(b). Overall, the recognition rates had a mean of 77.9% and standard deviation of 4.8% for the 6 behaviour classes over 20 trials. Fig. 5(b) also shows that when a normal behaviour pattern was not recognised correctly, it was most likely to be recognised as another class of normal behaviour pattern instead of being detected as an abnormality. This is a desirable feature from the perspective of achieving low false alarm rate for abnormality detection.

## 5. Conclusions

We have shown in this paper that selecting relevant eigenvectors is key for robust spectral clustering without knowing the number of clusters and proposed a novel algorithm for relevance learning. The effectiveness and robustness of our algorithm have been demonstrated through experiments on image segmentation and video based behaviour profiling.

## References

- [1] A. Dempster, N. Laird, and D. Rubin. Maximum-likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society B*, 39:1–38, 1977.
- [2] J. Dy, C. Brodley, A. Kak, L. Broderick, and A. Aisen. Unsupervised feature selection applied to content-based retrieval of lung images. *PAMI*, pages 373–378, 2003.
- [3] S. Gong and T. Xiang. Recognition of group activities using dynamic probabilistic networks. In *ICCV*, 2003.
- [4] M. Law, M.A.T. Figueiredo, and A.K. Jain. Simultaneous feature selection and clustering using mixture model. *PAMI*, 26(9):1154–1166, 2004.
- [5] L.R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.
- [6] J. Malik, S. Belongie, T. Leung, and J. Shi. Contour and texture analysis for image segmentation. *IJCV*, pages 7–27, 2001.
- [7] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, 2001.
- [8] F. Porikli and T. Haga. Event detection by eigenvector decomposition using object and frame features. In *CVPRW*, pages 114–121, 2004.
- [9] P. Smyth. Clustering sequence with hidden markov models. In *Advances in Neural Information Processing Systems*, pages 648–654, 1997.
- [10] Y. Weiss. Segmentation using eigenvectors: a unifying view. In *ICCV*, pages 975–982, 1999.
- [11] T. Xiang and S. Gong. Activity based video content trajectory representation and segmentation. In *BMVC*, 2004.
- [12] S. Yu and J. Shi. Multiclass spectral clustering. In *ICCV*, pages 313–319, 2003.
- [13] H. Zhong, J. Shi, and M. Visontai. Detecting unusual activity in video. In *CVPR*, pages 819–826, 2004.