On the Binding Mechanism of Synchronised Visual Events

Jeffrey Ng and Shaogang Gong
Department of Computer Science
Queen Mary, University of London
London E1 4NS, UK
E-mail: {jeffng,sgg}@dcs.qmul.ac.uk

Abstract

We address the problem of interpreting visual surveillance data by learning appropriate spatio-temporal subspaces of active image regions caused by scene activities. We focus on identifying regions of sustained change for recognising key stages of a visual behaviour. Our behaviour representation is based on the asynchrony or delay patterns of occurrence among local events which need not be spatially connected. We use an automatic Normalised Cut structure discovery algorithm with a hybrid similarity criteria for simultaneously identifying relevant spatio-temporal subspaces and clustering similar behaviour patterns in those subspaces. We compare the automatically discovered classes with conceptual classes of behaviours in a semi-constrained "Shopping" scenario.

1. Introduction

Event and behaviour interpretation play a crucial role in automatic visual surveillance. The problem has been commonly addressed by monitoring temporal changes in states of variables or objects in the scene using the Automata, Hidden Markov Models or Bayesian Belief Networks. However, the states of these variables are often hidden. Furthermore, visual ambiguities give rise to many plausible object states. The ambiguities can be partially addressed by learning prior knowledge to provide contextual information of the scene [1] or by propagating a probability density of belief states in a user-defined graph over time [9].

However, what is missing is that instead of using temporal information to disambiguate object dynamics, temporal information should be directly used in the process of binding events occurring in local image regions into meaningful scene behaviours. In particular, asynchrony, defined as the delay patterns in which activities occur across pixels or image regions, can be used to infer the correlation between the events comprising a behaviour [11]. Spiking Neurons, a relatively recent development in machine learning [4], explicitly learn the delays which underpin our notion of asynchrony in automatic behaviour recognition. However, the training algorithms for spiking neurons are similar to vector quantisation and have not proved to be adequate for classifying complex spatio-temporal patterns of visual events in our experiments. To address this, we further adopt an automatic structure discovery approach using spectral graph theory and extend the automatic NCut technique to learn intrinsic classes in spatio-temporal subspaces of asynchrony.

In Section 2, we describe the use of sustained change as a means of detecting active regions and the use of 'delays' in the activation of the regions as a spatio-temporal asynchrony pattern for differentiating events. In Section 3, we extend the NCut automatic structure discovery and clustering process [6] to the problem of finding effective classes of spatio-temporal subspace and event categories. We also train RBF Spiking Neurons [4] for the classification of each spatio-temporal category. We provide experimental results on a "Shopping" scenario where people enter, browse a selection of soft-drink cans and make a purchase or leave in Section 4. We conclude in Section 5.

2. Representing Spatio-Temporal Activity

Reliable tracking of objects requires prior knowledge in the form of shape or appearance models [3]. Object dynamics can be retrieved by finding the correlation of detected objects across frames. However, visual ambiguities, mutual occlusion, non-rigid deformations and in-depth rotation often complicate the process. While further temporal reasoning can partially help in resolving the ambiguity problem, prior knowledge can be directly encoded at a much lower-level to first determine interesting occurrences of sustained change and then to learn both the image-region subspaces of change and the temporal characteristics of these changes. The emphasis is shifted from recognising object dynamics, in relation to the overall trajectory, to correlating sparse local events together to identify behaviours.

2.1. Activity Representation

The complexity of analysing scenes at high spatial resolution becomes prohibitive for processes attempting to find pixel or region-level correspondences such as computing optical flow. High resolution only provides finer detail for appearance analysis while a low resolution retains the structure of activity [8]. Noise is also more prominent at high rather than low resolutions, which is specifically true for background Gaussian mixture models. To detect scene activity, we adopt a probabilistic pixel-based background modelling technique based on temporal reasoning of predominant colour clusters over long-term time-scale [5]. After the background subtraction process, we adopt a simple thresholded voting scheme to both remove noise and reduce the dimensionality of the data to a spatial resolution adequate for identifying different events. We decompose the image space into a grid of 20×15 cells and the cell-regions are activated when the number of active pixels in a cell exceed a 40% threshold for a reasonable detection confidence.

Instantaneous activation of the cells can be caused by the movement of foreground objects in the scene. Correlating the activation of cells across time could yield the dynamics and trajectory of the object. In this paper, we are interested in detecting occurrences of sustained local change in the scene. Sustained change may indicate an activity which is dependent on the propositional semantics of the particular region such as an interaction of a foreground object with a background feature or an interaction between more than one foreground object. We modify the Motion History Image [2] representation to include the capability to accumulate evidence in the presence of change in addition to the original decay in the absence of change.This is in order to detect sustained change. Given the function $A^{x,y}(t)$ indicating the activation of cell (x, y) at time t, the Accumulative MHI (AMHI) and Sustained Change (SC) function are given as,

$$\mathrm{AMHI}^{x,y}(t) = \begin{cases} 0 & \text{if } t = 0, \\ min[\alpha, \mathrm{AMHI}^{x,y}(t-1)+1] & \text{if } A^{x,y}(t) = 1, \\ max[0, \mathrm{AMHI}^{x,y}(t-1)-1] & \text{otherwise} \end{cases} \tag{1}$$

$$SC^{x,y}(t) = \{\begin{cases} 1 & \text{if } AMHI^{x,y}(t) > thresh \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where α is the maximum value for the AHMI representation and $thresh$ is a threshold above which accumulated change is considered sustained. These two parameters are control the time-scale over which change is considered sustained.

We define asynchrony as the delay pattern of the occurrence of local cell-activity events. We propose that the order and delay of the local events can be used as a binding criteria to correlate the events into a higher-order structure as a behaviour. As suggested in the psychophysical literature [11], asynchrony is only used to bind events over a fixed time-window to reduce the load of correlating spatially diverse events. Hence, we define the delay-based asynchrony representation of spatio-temporal events for cell x, y at time t with a temporal window of size w as,

$$syn^{x,y}(t) = \begin{cases} t - m^{x,y}(t), & \mathrm{if\ t - m^{x,y}(t) < w} \\ \infty, & otherwise \end{cases} \tag{3}$$

where $m(x, y, t)$ is the last activation time of cell (x, y) from time t, i.e. $m(x, y, t) = argmax_i(SC(x, y, i) = 1)$.

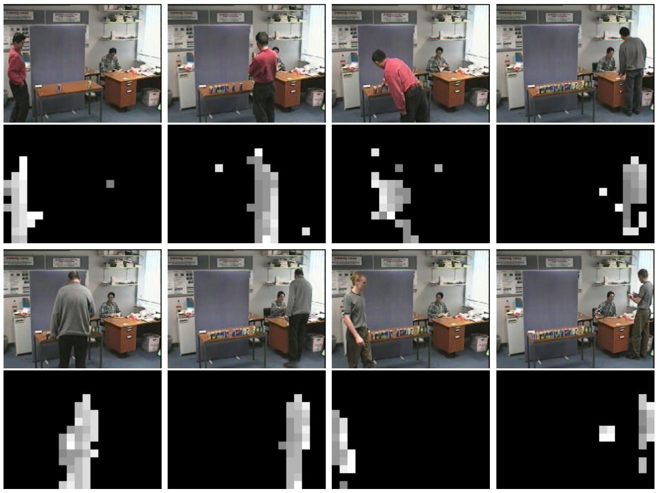


Figure 1. Examples from the "Shopping" scenario and corresponding asynchrony of sustained event patterns. Absence of events is denoted by black squares while relative delays of the occurrence of region-events are encoded from gray (least recent) to white (most recent).

2.2. Spatio-temporal Activity Similarity Measure

Spatio-temporal asynchrony patterns encode the correlation of different region-level events and bind them into a higher-order structure. However, asynchrony patterns are only relevant in relation to the distribution of other patterns in the scene. Frequently re-occurring patterns have a higher probability of having some semantical meaning and can be considered as important behaviours instead of spurious noise. Hence, we investigate the structure of the asynchrony patterns in the context in which they occur.

Conventional clustering techniques such as Vector Quantisation, k-means, Expectation Maximisation and Entropy Minimisation work on feature vectors of fixed length, treating all features, here delays, as equally valid. The feature vectors of spatio-temporal asynchrony delay patterns contain (in Equation (3)) a special token ∞ for the absence of activity in spatial cells, corresponding to the absence of correlation for these cells. Absent delays may be replaced by white noise but a high ratio of absent to available delays would result in more noise than usable information and greatly increase the difficulty of the learning process. It is even likely that subspace clustering techniques like hierarchical PCA [7] can become confused.

We therefore adopt a similarity representation that can cope with explicitly defined absent features in certain dimensions of asynchrony feature vectors and still retain the correlation between two asynchrony patterns with mismatched dimensions. Using a structure discovery algorithm on a similarity representation provides the additional benefit that the similarity criteria affects the structure of the discovered classes and can be made to emphasise certain spatial or temporal characteristics more than others. The asynchrony delay patterns possess both (a) a spatial component $spatialSimil$ in the dimensions which contain delays and (b) a temporal component $temporalSimil$ in the delay values contained in those dimensions. However, two different asynchrony patterns s_i and s_j have a set $d_{i,j}$ of common features, a set $e_{i,j}$ of mismatched dimensions with data only in one pattern and a set $f_{i,j}$ of common dimensions with available delays. We define the temporal component of the similarity criteria with an exponential function to convert the root-mean-squared difference of the temporal delay values into similarities for pairs of patterns for which $n(f_{i,j}) > 0$ (illustrated in Fig. 2(a)),

$$temporalSimil(s_i, s_j) = \begin{cases} 0 & \text{if } n(f_{i,j}) = 0, \\ exp(-\frac{\mathrm{ATD}(s_i,s_j)^2}{2\sigma^2}) & \text{otherwise} \end{cases} \tag{4}$$

where σ is the standard deviation of the Average Temporal Distance $\mathrm{ATD}(s_i, s_j)$ which is defined as,

$$\mathrm{ATD}(s_i, s_j) = \sqrt{\frac{\sum_{k \in f_{i,j}} (s_i^k - s_j^k)^2}{n(f_{i,j})}} \tag{5}$$

We define the spatial component of the similarity criteria as the ratio of the number of common dimensions with available delays to the total number of dimensions with at least a temporal delay present in either pattern as in Equation (6). However, owing to the large variations in the spatial regions occupied by an object, the spatial component may not be strong enough to provide a clear structure for structure discovery. The joint spatio-temporal similarity, illustrated in Fig. 2(b), of the two previous components results in a sparse similarity structure and a degenerate partitioning into small unitary sets. We apply a multiplication and thresholding function $spatialSimil'$ to increase the spatial similarity (Fig. 2(c)) and the joint spatio-temporal similarity (Fig. 2(d)), for a coherent similarity structure,

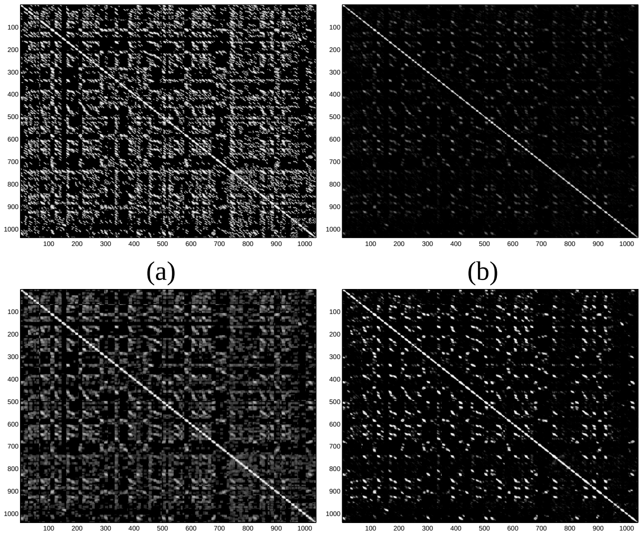
$$spatialSimil(s_i, s_j) = \frac{n(f_{i,j})}{n(f_{i,j} \cup e_{i,j})} \tag{6}$$

$$spatialSimil'(s_i, s_j) = (min(spatialSimil(s_i, s_j) {*} ef, 1))^2 \tag{7}$$

where ef is the enhancement factor which we set to 4.

The affinity matrix W comprising the similarity $w_{i,j}$ of asynchrony pattern i with another pattern j is defined by (illustrated in Fig 2(f)),

$$w_{i,j} = temporalSimil(s_i, s_j) \times spatialSimil'(s_i, s_j) \tag{8}$$



(a) (b)

(c) (d)

Figure 2. From top to bottom: (a) The temporal component of the pairwise similarity of asynchrony patterns, (b) the corresponding combined spatio-temporal affinity matrix with weak structure, (c) the enhanced spatial component $spatialSimil'$, and (d) the corresponding more coherent spatio-temporal affinity matrix. Intensity encodes similarity.

3. Binding Structure of Synchronised Events

Hierarchical dataset partitioning techniques related to Spectral Graph Theory have been reported to yield good results on image segmentation tasks [10]. The clustering process exploits the eigen-decomposition of the pairwise affinity (similarity) matrix of elements to split their membership into clusters with very high between-cluster similarity and very high intra-cluster dissimilarity, which is in essence similar to LDA. The Normalised Cut algorithm of Shi-Malik [10] has been mathematically proven to yield such a result when the elements of the second generalised eigenvector of the affinity matrix are integer-valued. However, for computational tractability, the constraint must be relaxed to allow for real values. Ng and Gong [6] used a cost function to separately re-evaluate the between-cluster and intra-cluster similarities in terms of the free parameter of the Shi and Malik NCut algorithm. This has been shown to successfully perform automatic model order selection on a dataset of warp-free trajectories. In this section, we describe the automatic NCut structure discovery algorithm and provide a brief overview of RBF Spiking Neurons.

3.1. Automatic Normalised Cut

Formulating the discovery of the binding structure of temporal events under spectral graph theory, the asynchrony patterns are considered as nodes in a graph while the affinity or similarity between the asynchrony patterns are considered as connecting edges. The term ‘binding’ is used for both the correlation of multiple asynchrony patterns together and the correlation which bind the region-level events into an asynchrony pattern. The two concepts are linked as finding the groups of asynchrony patterns also reveal which correlations of region-level events are important in the scene context. The normalised cut [10] of a graph is defined as the partitioning from the asynchrony-distribution graph into two sub-graphs which minimise the sum of broken edge connections relative to the sum of the edge connections from the subgraph to the whole graph.

Shi and Malik [10] have proven that thresholding the second generalised eigenvector of the affinity matrix $(\mathbf{D} - \mathbf{W})x = \lambda \mathbf{D} y$ results in a minimal NCut partitioning. The asynchrony-distribution graph is recursively partitioned into binary sets until the NCut value exceeds a certain threshold n or the second generalised eigen-vector becomes unstable with continuous values which are hard to threshold. Ng and Gong [6] have formulated a cost function which explicitly re-evaluates the trade-off between high intra-cluster similarity $(I(n))$ and low between-cluster similarity $(B(n))$ for the whole hierarchical partitioning instead of just one binary partitioning step as in the case of Shi and Malik,

$$f(n) = -ln(I(n)) + ln(B(n)) \tag{9}$$

where root-mean-square functions for the intra-cluster $I(n)$ and between cluster affinities $B(n)$ are computed from the partitioning resulting from parameter n. Minimising $f(n)$ partitions the asynchrony pattern distribution into intrinsic binding sets which share common spatio-temporal correlation. The minimisation is simplified as only discreet steps in the threshold n yields different partitioning solutions and all solutions under the NCut framework can be explored.

3.2. Learning Classes with RBF Spiking Neurons

Having discovered the intrinsic classes of asynchrony patterns in the context of scene activities, the binding of the region-level events into behaviours can be learned by a network of RBF-like Spiking Neurons [4]. The network of Spiking Neurons use the same basic representation of time-delays as our asynchrony patterns to encode information. More specifically, the information is encoded as the relative delays in the firing time of spikes from an input layer [4]. Given a asynchrony pattern $s_i = \{s_i^1, s_i^2, ..., s_i^p\}$, a series of spikes, called a spike train, is generated at the input layer according to $z_i = \{maxt - z_i^1, maxt - z_i^2, ..., maxt - z_i^p\}$ where $maxt$ is the maximum firing-time and thus the maximum delay allowed by the temporal-window used in Section 2. The learning goal for the spiking network is to obtain a further set of delays in the synapses connecting the input neurons to an output neuron, which when added to the firing-time of the input layer causes all the spikes to arrive simultaneously to an output neuron. Thus trained synapses learn the inverse of the asynchrony delay patterns. A simple thresholded leaky integrate-and-fire mechanism on the potentials of the synapses is enough to cause the output neuron to fire. However, only the weight of the synapses can normally be changed, not the delays. Thus the connection from an input neuron i_x to an output neuron o_y actually consists of a set of synapses each with predefined delays $d_1 = 1, d_2 = 2, ..., d_l = l$ and weights $\{w_1, w_2, ..., w_l\}$. The output potential Op at time t of the output neuron connected to u input neurons is given by,

$$Op_y(t) = \sum_{x=1}^{u} \sum_{c=1}^{l} w_c^{x,y} . \frac{t - (z_x + d_c^{x,y})}{t_p} exp(1 - \frac{t - (z_x + d_c^{x,y})}{t_p}) \tag{10}$$

where the constant $t_p = 3$ for the shape of the output potential function and the potential $Op_y(t)$ is thresholded for determining the firing time of the neuron (we used 1.0).

The RBF nature of spiking neurons is contained in the synaptic potentials connecting the input to the output layers. The summation of the non-linear exponential functions controlling the potentials across the delayed synapses from an input to an output neuron results in a Gaussian response. Thus the variance of asynchrony delays in each common dimension is learned. The output neuron which fired the earliest in response to an input spike pattern has the closest stored pattern to the input pattern. Its weights are thus adapted with a Hebbian learning rule to reinforce the similarity of the asynchrony pattern to the pattern already stored in the weights of the synapses while the weights of dissimilar delays are reduced. The learning function is defined as $L(\Delta T) = (1 - b) \cdot exp(-(\Delta t - c)^2/\beta)$ where $b = -0.11$, $\beta = 1.11$, $c = -2$ and $\Delta T =$ delay between the arrival of a spike at the output neuron and its firing time [4]. For classification, the output neuron which fires the earliest has recognised the input asynchrony pattern to belong to its class.

4. Experiments

In this section, we provide experimental results on a “Shopping” sequence. The scenario occurs in a typical indoor environment with an entrance on the left, a table with a selection of soft-drink cans in the middle and a shopkeeper on the right of the camera. Potential customers come in, browse the selection of cans and at this point can leave, pickup a can and pay for it, or pick up a can and leave. The semi-unconstrained way in which the customers move in the sequence have already caused trajectory analysis to report very few usable classes [6]. We thus investigate how spatio-temporal asynchrony analysis of the patterns of motion in occurrences of sustained change can extract important cues about the instantaneous state of visual events and their correlations that give rise to meaningful behaviours. We have obtained 1039 asynchrony patterns of sustained changed from the sequence of over 5000 frames.

We have manually organised the asynchrony patterns of sustained change into three overall conceptual categories to provide ground truth. The broad categories were aligned on the spatial regions in which the events occurred, namely (a) entrance at the door (b) can area (c) shopkeeper area. We created further sub-categories depending on the spatio-temporal characteristics as shown in Table 1. The criteria of sustained change allowed the detection of occurrences of people stopping to perform an action such as browsing fizzy drinks or effecting a purchase with the shopkeeper.

The warp-free temporal trajectory data used by Ng and Gong was well-structured from the pre-defined gesture categories and resulted in a nice binary hierarchical partitioning. However, the asynchrony patterns among temporal behaviour addressed in this work depend on the spatio-temporal characteristics of the activities in the scene. Although the broad categories of interaction were pre-defined, the order in which behaviour interactions were performed and the personal idiosyncrasies were left to the performers.

We compare the correlation of the classes obtained from the automatic NCut algorithm with the enhanced spatio-temporal similarity to the conceptual categories in Table 2. We can see the discovered classes have been separated mainly according to their physical location in the scene. Further discrimination based on the temporal characteristics is possible in the case of classes 1, 3 and 4. However, because some of these behaviours occur soon after each other, the continuity caused the sub-categories to be merged into the same class. The structure discovery also reserved a few classes for noisy asynchrony patterns which are very different from the others. We then use the automatic classes to train a network of spiking neurons with an output neuron assigned to each class. We show the results of the classification of the asynchrony pattern dataset with the trained spiking network in Table 3. The classification by the spiking network is quite similar to the intrinsic classes discovered by automatic normalised cut although the detectable occurrences are less. Neuron 7 appears to be too broad and merges many of the spatio-temporal asynchrony classes with subtle differences. Please note that neuron 8-10 are not shown as they have zero classification results.

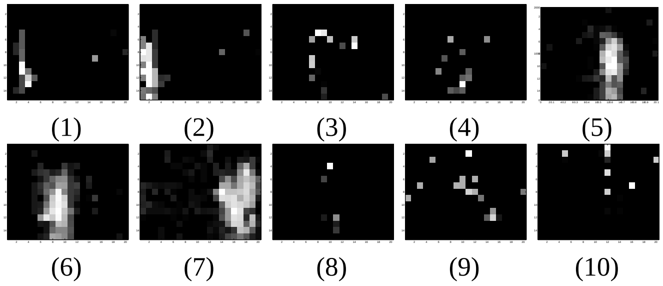


Figure 3. The average asynchrony classes discovered in the “Shopping” scenario.

5. Conclusions

In this work, we proposed a method for automatically binding visual events into behaviours using spatio-temporal asynchrony of sustained autonomous visual change. We consider this technique to provide better cues for selecting more relevant events in a given scene context. This is achieved without traditional manual labelling and hypotheses. We further exploit the use of spatio-temporal asynchrony to automatically analyse the correlation of relevant events in order to model behaviour. In sparse to medium busy scenes, all the regions of the scene will not show simultaneous activity and activity can be restricted to a small spatio-temporal subspace. We therefore used the automatic NCut clustering algorithm with a modified similarity criteria to perform subspace clustering of spatio-temporal asynchrony patterns from the “Shopping” sequence. We have shown that ordered structures of “sustained activity” can be discovered and learned by our model. Future work will be on learning higher-order correlation of events into a temporal framework for behaviour interpretation.

References

[1] D. Ayers and S. Mubarak. Monitoring human behavior from video taken in an office environment. *IVC*, 19(12):833–846, 2001.

[2] J.W. Davis and A.F. Bobick. The representation and

Table 1. Conceptual categories and sub-categories of spatio-temporal asynchrony patterns.

Entrance	Can Area	Shopkeeper
1a.Enter 1b.Leave 1c.Loiter	2a.Browse left 2b.Loiter 2c.Browse right 2d.Go to shopkeeper	3a.Purchase 3b.Go to table

Table 2. Correlation between discovered asynchrony-pattern classes and conceptual categories.

class/category	1a	1b	1c	2a	2b	2c	2d	3a	3b	noise
1	0	0	**11**	0	0	0	0	0	0	0
2	**16**	0	**10**	0	0	0	0	0	0	0
3	0	0	0	0	**11**	0	0	0	0	0
4	0	0	0	0	**3**	0	0	0	0	0
5	0	0	0	**45**	**26**	**4**	**43**	0	18	24
6	0	0	0	**14**	**79**	**37**	0	0	0	44
7	0	12	7	2	11	1	**64**	**341**	**75**	113
8	0	0	0	0	0	0	0	0	0	2
9	0	0	0	0	0	0	0	0	0	17
10	0	0	0	0	0	0	0	0	0	9

Table 3. The classification of the asynchrony patterns with a trained spiking network.

output neuron	1a	1b	1c	2a	2b	2c	2d	3a	3b	noise
1	**5**	0	**3**	0	0	0	0	0	0	0
2	0	0	0	0	**14**	0	0	0	0	0
3	0	0	0	**5**	**35**	**12**	0	0	0	19
4	0	0	0	0	**9**	0	0	0	0	0
5	0	0	0	**31**	**8**	0	**20**	0	**7**	14
6	0	0	0	0	0	1	**28**	**136**	**16**	36
7	11	12	25	25	64	29	59	205	70	120

recognition of human movement using temporal templates. In *CVPR*, pages 928–934, 1997.

[3] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *ECCV*, pages 343–356, 1996.

[4] T. Natschlger and B. Ruf. Spatial and temporal pattern analysis via spiking neurons. *Network: Computation in Neural Systems*, 9:319–332, 1998.

[5] J. Ng and S. Gong. Learning pixel-wise signal energy for understanding semantics. In *BMVC*, pages 695–704, U.K., 2001.

[6] J. Ng and S. Gong. Learning intrinsic video content using levenshtein distance in graph partitioning. In *ECCV*, pages 670–684, 2002.

[7] E.J. Ong and S. Gong. A dynamic 3d human model using hybrid 2d-3d representations in hierarchical pca space. In *BMVC*, volume 1, pages 33–42, UK, 1999.

[8] A. Robles-Kelly and E.R. Hancock. An em-like algorithm for motion segmentation via eigendecomposition. In *BMVC*, pages 123–132, U.K., 2001.

[9] J. Sherrah and S. Gong. Resolving visual uncertainty and occlusion through probabilistic reasoning. In *BMVC*, pages 252–261, UK, 2000.

[10] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE PAMI*, 22(8):888–905, August 2000.

[11] M. Usher and N. Donnelly. Visual synchrony affects binding and segmentation in perception. *Letter to Nature*, 394:179–182, 1998.