Learning Articulated Motion Structures with Bayesian Networks

Fabio T. Ramos, Hugh F. Durrant-Whyte, Ben Upcroft
ARC Centre of Excellence in Autonomous Systems (CAS),
Australian Centre for Field Robotics,
The University of Sydney, NSW 2006, Sydney, Australia.
{f.ramos, hugh, b.upcroft}@acfr.usyd.edu.au

***Abstract*— This paper presents a general methodology for learning articulated motions that, despite having non-linear correlations, are cyclical and have a defined pattern of behaviour. Using conventional algorithms to extract features from images, a Bayesian classifier is applied to cluster and classify features of the moving object. Clusters are then associated in different frames and structure learning algorithms for Bayesian networks are used to recover the structure of the motion. This framework is applied to the human gait analysis and tracking but applications include any coordinated movement such as multi-robots behaviour analysis.**

I. Introduction

A key challenge in robotics is learning a representation of an unstructured world given a set of sequential measurements. The environment can be dynamic, with multiple moving objects and subject to changes in illumination. As the robot moves, an object is seen from different perspectives and parts may be occluded by other objects. In addition, sensor measurements may be erroneous, so requiring a representation able to handle uncertainty. A possible approach to these problems is to employ a state estimator such as a Kalman filter (KF) or Extended Kalman Filter (EKF). These estimators describe the process of state transition and observation, and generate an estimate that minimises estimated mean square error. However, most applications of KFs consider only point targets or objects represented by a group of points with the same dynamic model. In this paper, we are interested in tracking the motion of complex structures, with correlations between parts of the same structure which may, nevertheless, execute separate but correlated motion. The techniques developed are applicable to problems such as tracking human motion or coordinated motions of sets of robots.

The human tracking problem and gait analysis have been widely studied in the past twenty years. Gait is an important feature of humans and can be used to identify individuals [1], [2], the sex of the person [3], or even to recognise groups of friends [4]. The computer vision community has addressed this problem by computing trajectories of body joints and creating temporal models of them [5], [6], [7]. It can be formulated in a probabilistic manner with two different approaches, one based on point features and another based on intensity. Feature-based approaches have the advantage of being able to employ many different algorithms for feature extraction and are generally more amenable to real-time implementation. However, they have additional problems in associating features from different image frames. The most similar approach to our is the one of Song et al [7]. In this work, a probabilistic framework is used to identify joints in the human body. Triangulated graphs are used to represent the structure of the body which can be learnt with an Expectation Maximisation (EM) algorithm. Labelling and classification of features is achieved through maximising the likelihood of the data given the decomposition represented by the triangulated graphs. In our approach, rather than labelling each feature using an existent structural model, we first cluster features using the EM algorithm and then learn the structural model by finding correlations between clusters. Features are extracted from a stream of frames with the Kanade-Lucas-Tomasi (KLT) algorithm [8] and contain positions and velocities. Then, EM is used to cluster these features under the assumption that positions and velocities are independent given the class. In other words, a Naive Bayes classifier [9], represented as a Bayesian network, is learnt with the class variable being hidden. Once parameters are learnt, the classifier can be applied in features of different frames, making the association task straightforward.

With features labelled in all frames, it is then possible to learn dependencies among clusters, so building a Bayesian network model of the motion. In complex structures, dependencies can be non-linear, i.e. variables may be function of a non-linear combination of its descendants. Unfortunately, learning a Bayesian network with continuous nodes and non-linear relations between variables, even assuming these to be Gaussian distributed, is a cumbersome task where Monte Carlo algorithms must generally be applied [10]. An alternative to tackle this problem is presented here by representing non-linear dependencies as a set of net structures, with linear Gaussians distributions. For each frame, a network structure is learnt along with its correlations with the previous frame. As motions are usually periodic, the learning process can stop when the structures have the same dependencies as those previously learnt.

This paper is organised as follows: in Section II we present formal definitions and a brief review of Bayesian networks. Section III shows how to cluster features in an unsupervised fashion using the EM algorithm. Section IV presents the

structure and parameters learning algorithms along with some experimental results. We conclude in Section V and present some ideas for future work.

II. Preliminaries

This section briefly reviews Bayesian networks and introduces necessary notation. Capital letters (X, Y, Z) are used to denote names of random variables, lowercase letters (x, y, z) to denote specific values taken by those variables, boldface capital letters ($\mathbf{X}, \mathbf{Y}, \mathbf{Z}$) to denote sets of random variables and boldface lowercase variables ($\mathbf{x}, \mathbf{y}, \mathbf{z}$) to denote values taken by those sets. A joint probability over a set $\mathbf{X} = \{X_1, X_2, ..., X_n\}$ is denote by $P(\mathbf{X})$.

A Bayesian network is defined as a tuple $\mathcal{B} = \langle \mathcal{G}, \Theta \rangle$ where $\mathcal{G}$ is a directed acyclic graph whose vertices represent random variables and Θ are the parameters that define the distributions. The main assumption encoded by a BN is that each variable X_i is conditionally independent of its non-parents given its parents. The joint probability is defined by:

$$P(\mathbf{X}) = \prod_i P(X_i|\mathbf{Pa}(X_i)), \tag{1}$$

where $\mathbf{Pa}(X_i)$ represent the parents of the variable X_i.

In this paper, Bayesian networks are used for two different tasks: 1) unsupervised classification of features and 2) learning and representation of the structure of the motion. Except for the class variable of the classifier, all other variables are assumed to have a normal (Gaussian) distribution with parameters μ and σ^2, with the distribution denoted by $\mathcal{N}\left(X; \mu, \sigma^2\right)$. Then, assuming linear relation between Gaussians and an order $X_1, \ldots, X_n$ of variables, it is possible to define linear conditional Gaussian distributions as:

$$P\left(X_i|X_1, \ldots, X_{i-1}\right) = \mathcal{N}\left(X_i; \beta_{i,0} + \sum_{j=1}^{i-1} \beta_{i,j} X_j, \sigma_i^2\right), \tag{2}$$

where $\beta_{i,0}$ and $\beta_{i,j}$ describe the linear combination of the variable X_i given its parents $X_1, \ldots, X_{i-1}$. When $\beta_{i,j} \neq 0$, there is an edge from X_j to X_i forming a graph. This definition thus brings linear Gaussian distributions into Bayesian networks. If $\beta_{i,j} = 0$ for every i and j, the variable X_i is a root node with a univariate Gaussian distribution. The joint probability distribution with all variables being Gaussian is then $\mathcal{N}\left(\mathbf{X}; \mu, \Sigma\right)$, defined as:

$$P\left(\mathbf{X}\right) = \frac{1}{\left(2\pi\right)^{n/2} \left|\Sigma\right|^{1/2}} \exp\left(-\frac{1}{2}\left(\mathbf{x} - \mu\right)^T \Sigma^{-1} \left(\mathbf{x} - \mu\right)\right), \tag{3}$$

where μ is a vector of size n and Σ is a symmetric positive-definite matrix of size $n \times n$.

Making inferences in a Bayesian network is the task of computing posterior probabilities given some observed values. That is, given a set of *query* variables $\mathbf{X}_q$ and a set of *evidence* $\mathbf{X}_e = \mathbf{x}_e$, we compute $P\left(\mathbf{X}_q|\mathbf{X}_e = \mathbf{x}_e\right)$ which, with continuous distributions, is proportional to the marginalisation of the joint probability over variables $\mathbf{X}_z$, where $\mathbf{X}_z = \mathbf{X} \setminus \left(\mathbf{X}_q \cup \mathbf{X}_e\right)$:

$$P\left(\mathbf{X}_q|\mathbf{X}_e\right) \propto \int \prod_i P(X_i|\mathbf{Pa}(X_i))\, d\mathbf{X}_z. \tag{4}$$

Algorithms for inference in these models are discussed in [11], [12], [13]. These algorithms describe Gaussian distributions using *canonical characteristics* and perform message-propagation in a *juction tree* [14] to calculate marginal distributions. A limitation exists when there are deterministic relations between variables since the covariance matrix Σ is not invertible, and the canonical form needs to invert the covariance matrix to calculate one of its terms. To overcome this problem, it is possible to use conditional forms [15] which are also more numerically stable than canonical forms when the net has both discrete and continuous variables. A deeper discussion of inference with linear Gaussian distributions is beyond the scope of this paper.

In a frequentist approach, the parameters of linear Gaussian models can be learnt using maximum-likelihood techniques. See [16], [17] for details.

III. Unsupervised Feature Classification

We start our discussion about learning motion structures by analysing the problem of feature association. Given a set of features extracted by an optical-flow based algorithm (KLT), the first step towards structure reconstruction is to associate features from different frames. In a complex environment with changes in luminosity, occlusions, rotations and translations of objects, features can appear and disappear from frame to frame. If there is no predefined dynamic model describing the behaviour of such features, the problem of predicting the position of a particular occluded feature becomes very complex. In the same way, association of those features fails due to lack of observability. In this work, instead of trying to track individual features fixed in an object, features are clustered using probabilistic methods and only the created clusters are tracked. We advocate that this method is more robust in dealing with occlusions and inaccurate information from the feature extraction algorithm than methods that try associate features individually.

To classify and cluster features we use the well-known Naive Bayes classifier. The Naive Bayes classifier [9] assumes that the attributes are conditionally independent given the class. This assumption is quite reasonable in our problem whose attributes are positions and velocities for the features extracted. Note that at this point, there is no association between features in consecutive frames so that velocities and positions are independent. In the Naive Bayes model, the probability of a specific label c, given the observed attributes, is given by:

$$P\left(c|x, y, \dot{x}, \dot{y}\right) = P\left(x|c\right) P\left(y|c\right) P\left(\dot{x}|c\right) P\left(\dot{y}|c\right) P\left(c\right). \tag{5}$$

A feature will belong to the label that maximises its posterior probability. Figure 1 shows the Bayesian network representing the Naive Bayes classifier used to cluster features.

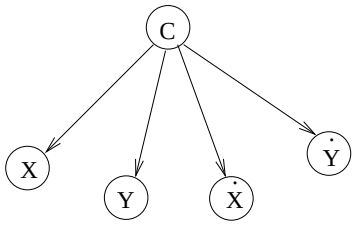

Fig. 1. The Naive Bayes classifier to cluster features.

Fig. 2. An example of a sequence of frames from a person walking from the right to the left side of the scene.

An alternative way to classify features is through a dynamic Naive Bayes classifier. In this case it is assumed that the class describes a stochastic process $\{C(t), t \in T\}$ where t is a time slice - or a frame - in the stream. If assumed that this process is stationary, the dynamic classifier can be represented as a dynamic Bayesian network with transitions given by $P(C(t)|C(t-1))$.

In the unsupervised approach, a Naive Bayes classifier can be learnt using maximum-likelihood techniques such as the EM algorithm [18], [19]. The main idea of the EM algorithm is to apply the Jensen's inequality [20] to simplify the computation of the log-likelihood. At each interaction, the EM computes the expected value of the hidden variables given the current data and parameters (E-Step). Then, it finds new values for the parameters that maximise the likelihood (M-Step). The only parameter that has to be defined *a priori* is the number of categories that the class variable can have. This value is equal or larger than the number of clusters identified with EM - it is larger if EM finds no feature for a particular cluster. 10 categories are used in our experiments.

EM is known to suffer from the overfitting problem [1] and convergence is guaranteed only to a local maximum. We overcome this problem by initialising it properly. As positions and velocities in the image have a known order of magnitude — positions vary only from 1 to 320 and 1 to 240 for x and y and velocities from -20 pixels per frame to 20 pixels per frame — a random initialisation followed by a couple of iterations of K-nearest neighbours (KNN) [21] are enough to guarantee acceptable convergence properties for EM. In the tests performed the overfitting problem never occurred and in the vast majority of cases, clusters were correctly identified.

Using EM, the classifier can be trained with the features extracted by the KLT algorithm whose attributes are positions and velocities for all features detected, regardless of which frame they come from. To do so, it is necessary to remove possible translations from the position variables. For example, suppose that the motion recorded in a video is of a person walking from the left to the right side of the screen, with the camera remaining fixed during the whole video acquisition. Figure 2 shows five frames of this example grabbed with a camera of 320 x 240 pixel resolution. As the person walks, the x position of the detected features changes accompanying the body motion. In order to make the Gaussian assumption reasonable, the translation is removed by subtracting the mean of the x positions of all features detected in a particular frame from the x position of each feature in that frame. For each feature i detected in the frame t its corrected $x_i(t)$ position is given by:

$$x_i(t) = x_i(t) - \mu_x(t),$$

where $\mu_x(t)$ is the mean of the x position of all features detected in frame t.

The data set for the Naive Bayes classifier is thus a set $\mathcal{D} = \{\mathbf{d}_{1,1}, \mathbf{d}_{1,2}, \ldots, \mathbf{d}_{1,N_1}, \ldots, \mathbf{d}_{T,1}, \mathbf{d}_{T,2}, \ldots, \mathbf{d}_{T,N_T}\}$, where T is the number of frames in the stream and N_i is the number of features detected in the frame i with each sample $\mathbf{d}_{i,j} \in \mathbb{R}^4$ and $\mathbf{d}_{i,j} = \{x, y, \dot{x}, \dot{y}\}^T$.

With all parameters determined, features are clustered by making inferences on the Bayes net of Figure 1. Resulting clustered features are classified with a unique label. Figure 3 shows the result of this process on the sequence of Figure 2. Note that features with velocities close to nil are represented with light gray unfilled squares. They move to the foot in contact with the ground since velocities in this region are zero. Another interesting cluster is the one represented by dark gray unfilled squares. Features of this cluster are associated with the movement of the head and remain accompanying it during the whole sequence.

By making inferences with evidence from features detected across frames, it is possible to associate clusters in the whole stream. The samples in the data set are then modified to incorporate one more dimension representing their labels. Thus, $\mathbf{d}_{i,j} \in \mathbb{R}^5$ and $\mathbf{d}_{i,j} = \{x, y, \dot{x}, \dot{y}, c\}^T$ where c is the label or cluster that the feature belongs.

IV. Learning the Motion Structure

With a group of samples for each cluster, in each frame (time slice) the motion structure can be learnt using structure learning algorithms for Bayesian networks. However, complex motions may have non-linear dependencies, and a linear Gaussian network may not be directly applicable. Our strategy to tackle this problem is to approximate non-linear relations to

[1]The overfitting problem occurs when one of the classes gets associated with few data samples. This makes the covariance matrix of that particular class to decrease its magnitude and the likelihood to increase in the same proportion up to a point where numerical problems stop the algorithm the proceed.

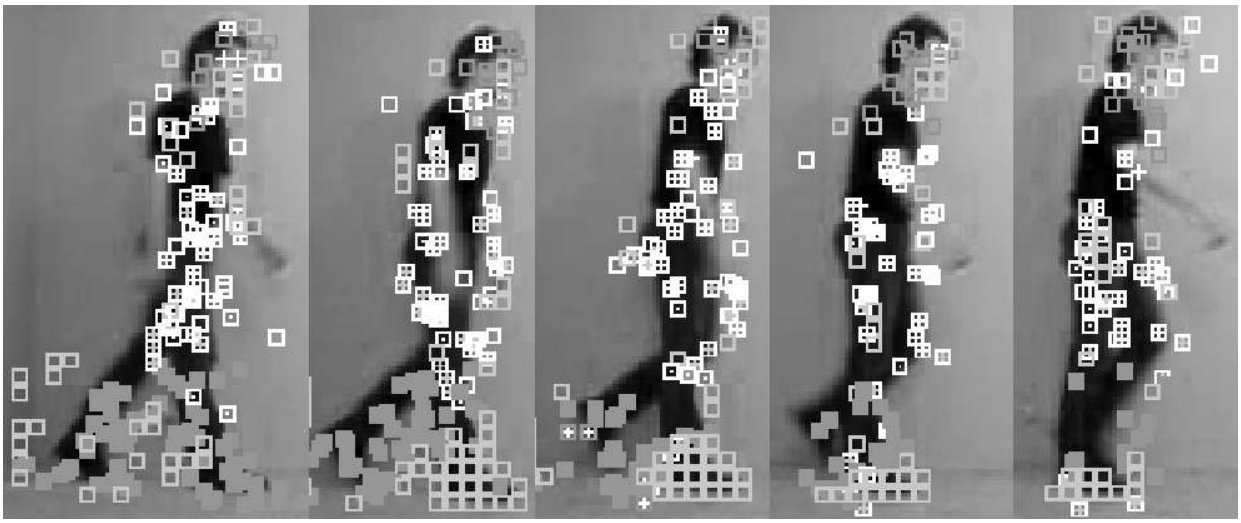

Fig. 3. Features clustered using the learnt Naive Bayes. Features represented with the same symbol belong to the same cluster.

linear relations, learning a different structure for each time slice, until structures start repeating. Our assumption is that, even in complex motions like a human body walking, there exists a pattern that is repeated over some (unknown) time interval. The idea is to try to learn this pattern and then construct a Bayesian network to describe it. As conditional probabilities and dependencies change with time, a dynamic Bayesian network, as it is normally defined, would not be appropriate to represent the model. Nonetheless, it is possible to consider the whole motion pattern learnt with a Bayesian network as the structure repeated in a dynamic Bayesian network (DBN). Thus, with a slight change in the definition of DBNs, the problem can be described in the form of DBN structure learning.

Given the data set with labelled samples, the algorithm works as follows. Suppose that in the frame t there are n clusters, $C_t^1, \dots C_t^n$, identified with at least m features per cluster, using the procedure described in Section III. In the next frame $t+1$, the same n clusters are identified, $C_{t+1}^1, \dots, C_{t+1}^n$, with m samples per cluster[2]. The first step of the algorithm is to learn the structure represented by the clusters in the first frame. Structure learning in a Bayes net involves a search over the set of all possible directed acyclic graphs, scored by a determined scoring function. As the number of possible graphs grows super-exponentially with the number of variables, a heuristic strategy must be used. In this work, the scoring function used is the well known Bayesian Information Criterion (BIC) [22] which is equivalent to the Minimum Description Length (MDL) approach [23]. Essentially the BIC has one term that is exactly the log-likelihood, measuring how well the model predicts the data, and one term to penalise the complexity of the model:

$$\mathrm{BIC}\left(\mathcal{G}\right) = \sum_n \log P\left(D^n | \hat{\theta}, \mathcal{G}\right) - \frac{np}{2} \log N, \qquad (6)$$

where np is the total number of parameters of the Bayes net and N is the number of samples. A greedy search is used to find the graph that maximises the scoring function. The search starts with a fully connected graph and operations of adding, removing and reverting edges are performed until a local maximum is obtained.

Having learnt the structure for frame t, the same procedure is repeated for frame $t+1$, and another structure is learnt. With two consecutive structures, correlations between clusters in different frames are discovered. This can be done using the same greedy search heuristic, under the constraint that clusters in frame $t+1$ cannot be parents of clusters in frame t. This ensures a Markov assumption where variables are independent of the past given the present. Figure 4 shows an example of a learnt structure for two consecutive frames. Note that in contrast to a Dynamic Bayesian Network where the net has the same structure for every time step t with $t \neq t_0$, the network structure of Figure 4 differs in the two consecutive frames. The algorithm continues learning structures and inter-frame dependencies until the new learnt structures start being *similar* to those previously learnt, indicating that the cycle has finished. *Similar* in this case is understood in terms of relative entropy or Kullback-Leibler (KL) divergence [20]. Thus, the learning process stops when a defined threshold of KL divergence is achieved. A sketch of the algorithm is shown in Algorithm 1.

Results from the complete algorithm are presented in Figure 5 for the first five frames of a motion pattern. The motion pattern has 19 inter-connected structures representing the whole cycle of a typical human gait. Edges represented with solid lines indicate conditional dependencies between clusters in the same frame, while edges represented with dash lines show the

[2]In the case that more than m features were identified, some of them can be excluded by selecting the m features that have higher probability of belonging to that particular cluster.

Algorithm 1 A pseudo-code for Motion Structure Algorithm.

Inputs: A set of labelled features $\mathcal{D}$;
KL threshold, k.
Output: Learnt BN encoding the motion pattern, $\mathcal{B}$.
While $stop > k$ do
$B_t \leftarrow$greedy_search(D, t)
$B_{t+1} \leftarrow$greedy_search$(D, t+1)$
$B_t^{t+1} \leftarrow$greedy_search$(D, t$,$t+1)$ //inter dep.
$\mathcal{B} \leftarrow \mathcal{B} + \langle B_t, B_{t+1}, B_t^{t+1} \rangle$
$t \leftarrow t+1$
$stop \leftarrow KL(B_t; B_{t_0})$
End

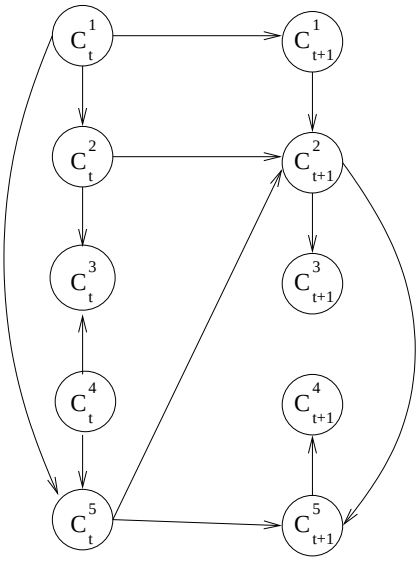

Fig. 4. Structure learnt from samples of 10 clusters into 2 consecutive frames.

inter-frame correlation. From this figure it is possible to note that clusters associated with the trunk, such as $C4$, $C5$ and $C6$, are normally the parents of other clusters in inter-frame correlations. They represent the centre of the body where the movement of other parts are based on and therefore, tend to have more correlations. Besides, the trunk has a movement closer to linear than other parts such as the limbs. Thus, it is expected that they have similar inter-frame correlations between themselves.

V. Conclusions and Future Work

The algorithm described in this paper provides a general methodology to learn complex motion structures that have specific patterns. With a set of features extracted from a video, clusters are identified and tracked. These represent characteristics of the object being tracked whose dependencies can be analysed and learnt. Once a sequence of structures and their correlations are obtained, the built network can be used to predict positions and velocities or the general behaviour of the model. Experiments were undertaken using a video of a walking human, however the techniques presented here can be used for more general proposes such as recovering the behaviour of a group of robots whose actions have some coordination. The learning algorithm can be implemented online and can incorporate techniques to select samples - similar to that presented in Section III.

One of the drawbacks of the proposed algorithm is that it is necessary to store the whole BN encoding the pattern. If the cycle of the motion is long, then the network will grow, possibly becoming intractable for exact inference algorithms. Alternatives to tackle this problem are non-linear regression methods that, by learning non-linear correlations, can incorporate sequences of motions in one structure.

Acknowledgements

This work is supported by the ARC Centre of Excellence programme, funded by the Australian Research Council (ARC) and the New South Wales State Government.

References

[1] J. J. Little and J. E. Boyd, "Recognizing people by their gait: the shape of motion," in *Videre*, vol. 1(2), 1998.
[2] G. V. Veres, L. Gordon, J. N. Carter, and M. S. Nixon, "A multi-view method for gait recognition using static body parameters," in *The 3rd International Conference on Audio and Video Based Biometric Person Authentication*, 2001.
[3] L. Kozlowski and J. Cutting, "Recognizing the sex of a walker from a dynamic point-light display," in *Perception and Psychophysics*, vol. 21, 1977, pp. 575–580.
[4] J. Cutting and L. Kozlowski, "Recognizing friends bu their walk: Gait perception without familiarity cues," in *Bulletin of the Psychonomic Society*, vol. 9, 1977, pp. 353–356.
[5] G. V. Veres, L. Gordon, J. N. Carter, and M. S. Nixon, "What image information is important in silhouette-based gait recognition," in *Proc. of Conf. on Computer Vision and Pattern Recognition*, vol. 2, 2004, pp. 776–782.
[6] R. Tanawongsuwan and A. Bobick, "Gait recognition from time-normalized joint-angle trajectories in the walking plane," in *Proc. of Conf. on Computer Vision and Pattern Recognition*, 2001.
[7] Y. Song, L. Goncalves, and P. Perona, "Unsupervised learning of human motion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 7, pp. 814–827, July 2003.
[8] C. Tomasi and T. Kanade, "Detection and tracking of point features," School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, Tech. Rep. CMU-CS-91-132, 1991.
[9] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifi ers," *Machine Learning*, vol. 29, no. 2-3, pp. 131–163, 1997. [Online]. Available: citeseer.ist.psu.edu/friedman97bayesian.html
[10] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. New York: Springer-Verlag, 2001.
[11] S. L. Lauritzen, "Propagation of probabilities, means, and variances in mixed graphical association models," *Journal of the American Statistical Association*, vol. 87, no. 420, pp. 1098–1108, 1992. [Online]. Available: citeseer.ist.psu.edu/lauritzen92propagation.html
[12] K. P. Murphy, "Inference and learning in hybrid Bayesian networks," Computer Science Division, University of California, Berkeley, CA 94720, Tech. Rep. UCB/CSD-98-990, 1998.
[13] U. N. Lerner, "Hybrid Bayesian networks for reasoning about complex systems," Ph.D. dissertation, Department of Computer Science, Stanford University, October 2002.
[14] C. Huang and A. Darwiche, "Inference in belief networks: A procedural guide," *International Journal of Approximate Reasoning*, vol. 15(3), pp. 225–263, 1996.
[15] S. L. Lauritzen and F. Jensen, "Stable local computation with conditional Gaussian distributions," Department of Mathematical Sciences, Aalborg University, Aalborg, Denmark, Tech. Rep. R-99-2014, September 1999.
[16] K. P. Murphy, "Dynamic Bayesian networks: Representation, inference and learning," Ph.D. dissertation, Computer Science Division, University of California, Berkeley, 2002.
[17] S. L. Lauritzen, *Graphical Models*. Oxford: Clarendon Press, 1996.

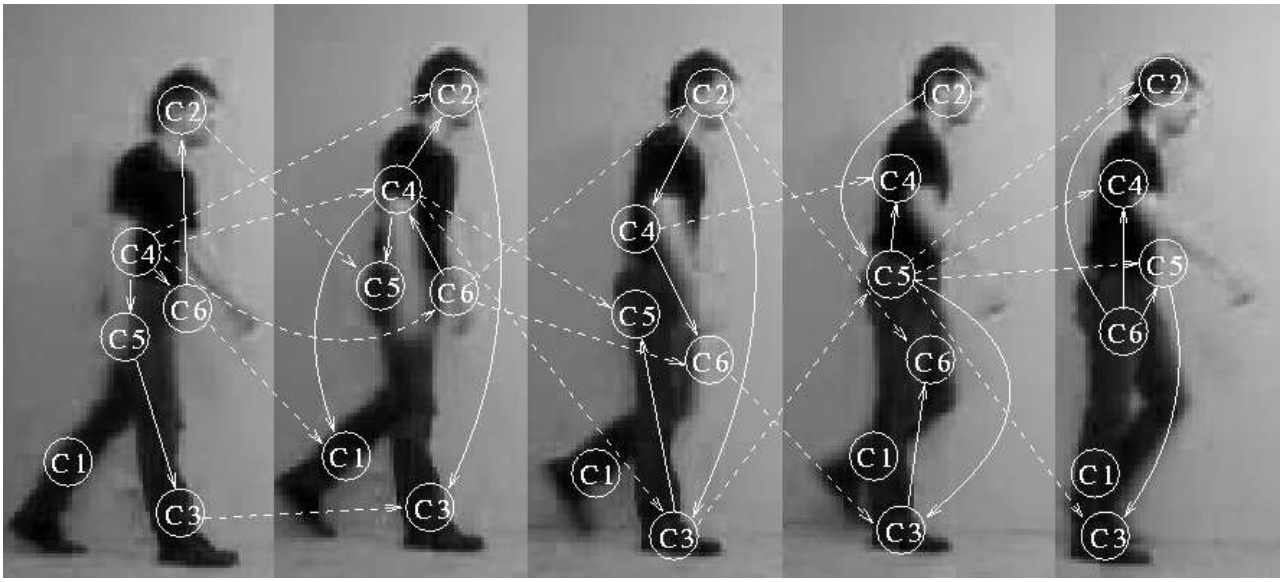

Fig. 5. This picture shows the first five frames of a typical human gait and the learnt structures. The positions of the nodes - representing clusters - were calculated by taking the average of the labelled feature positions. Edges in the same frame are represented with solid lines while inter-frame correlations are represented with dash lines.

[18] A. P. Dempster and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[19] R. M. Neal and G. E. Hinton, "A new view of the EM algorithm that justifies incremental and other variants," *Submitted to Biometrika*, 1993.

[20] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: John Wiley & Sons, Inc, 1991.

[21] D. J. MacKay, *Information Theory, Learning and Inference*. Cambridge University Press, 2003.

[22] D. Heckerman, "A tutorial on learning with Bayesian networks," Advanced Technology Division, Microsoft Corporation, Redmond, WA 98052, Tech. Rep. MSR-TR-95-06, 1996.

[23] J. Suzuki, "Learning Bayesian belief networks based on the MDL principle: An efficient algorithm using the branch and bound technique," *IEICE Transactions on Information and Systems*, vol. E81-D, no. 12, December 1998. [Online]. Available: citeseer.ist.psu.edu/article/suzuki96learning.html