

Learning Montages of Transformed Latent Images as Representations of Objects That Change in Appearance

Chris Pal¹, Brendan J. Frey², and Nebojsa Jojic³

¹ University of Waterloo, Dept. Computer Science, Waterloo, ON, N2L 3G1, Canada.
cjp@uwaterloo.ca

² University of Toronto, Dept. Electrical and Computer Engineering, Toronto, ON,
M5S 3G4, Canada. frey@psi.utoronto.ca

³ Microsoft Research, Redmond, WA, 98052, USA. jojic@microsoft.com

Abstract. This paper introduces a novel probabilistic model for representing objects that change in appearance as a result of changes in pose, due to small deformations of their sub-parts and the relative spatial transformation of sub-parts of the object. We call the model a *probabilistic montage*. The model is based upon the idea that an image can be represented as a montage using many, small transformed and cropped patches from a collection of latent images. The approach is similar to that which might be employed by a police artist who might represent an image of a criminal suspect's face using a montage of face parts cut out of a "library" of face parts. In contrast, for our model, we learn the library of small latent images from a set of examples of objects that are changing in shape. In our approach, first the image is divided into a grid of sub-images. Each sub-image in the grid acts as window that crops a piece out of one of a collection of slightly larger images possible for that location in the image. We illustrate various probability models that can be used to encode the appropriate relationships for latent images and cropping transformations among the different patches. In this paper we present the complete algorithm for a tree-structured model. We show how the approach and model are able to find representations of the appearance of full body images of people in motion. We show how our approach can be used to learn representations of objects in an "unsupervised" manner and present results using our model for recognition and tracking purposes in a "supervised" manner.

1 Introduction

In this paper we address the problem of learning representations of objects that change in appearance as a result of small non rigid deformations and changes in appearance arising when objects with rigid sub-components are translated in relative position. This paper introduces the *probabilistic montage*, illustrates some variations of the model, present a complete algorithm for a tree structured model and presents some results using a tree structured model for recognition tasks applied to full body images of people in motion. Importantly, the parameterization

of the probabilistic montages presented in this paper encode few assumptions specific to human figures. As such, probabilistic montages are applicable to various learning and classification tasks.

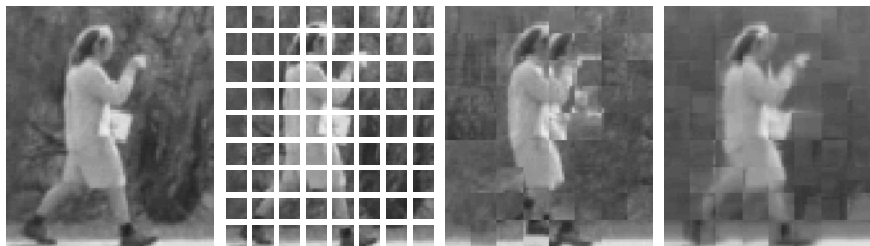


Fig. 1. (left to right) (a) A still image of a person walking, carrying a cup and a book. (b) The image is broken up into a grid. (c) A montage constructed from four possible latent images per grid cell *randomly selected* from different images in a set of similar images. (d) A montage constructed from four latent images *learned* from a set of images.

In our approach, we learn a collection of small latent images from a set of examples of objects that are changing in shape. First the image is divided into a grid of sub-images. Each sub-image in the grid acts as window that crops a piece out of one of the collection of slightly larger images possible for that location in the image. Additionally a rotation of the image for the patch can also be applied. Figure 1 illustrates our approach. Figure 1 (a) and (b) show an 80×90 image and a grid of sub-images. Figure 1 (c) and (d) show montages approximating the image using the best (in the maximum *a posteriori* or MAP sense) cropping and transformation of the best image from a small subset of slightly larger images. Figure 1 (c) shows how a montage might be created in a simple model where the collection of slightly larger latent images for each grid cell was gathered by randomly sampling four different frames of a video sequence. The frame in (a) was then reconstructed using the best cropping and transformation possible from the collection of possible latent images for each location. Figure 1 (d) illustrates the MAP montage when four latent images for each grid cell were learned from an image sequence. Notice that when latent images are selected randomly, the model often has “no choice” but to use a poor approximation (e.g. near the legs and the book).

In this paper, we discuss how the distribution over latent images and which transformations and cropping locations to use can be characterized using different probabilistic representations. We then derive an EM algorithm for a tree structured model and present some results of modelling human figures walking while carrying and manipulating objects. We will assume that the objects of interest have been coarsely centered within the image. Figure 2 shows some 80×90 pixel greyscale images of a person walking to the left and to the right

while carrying a book and drinking a cup of coffee. We shall present results in which our model has clustered images of the walking human subject into poses in an "unsupervised" fashion and we shall present some "supervised" classification results. In this paper we are not focusing on the task of capturing the dynamics



Fig. 2. Examples of images used to train the model.

of human walking motion. We are interested in the general task of characterizing different configurations of an object (e.g. poses of a moving person or deformations of a face) and modelling sub-types of objects for which there may not be a single object undergoing a dynamic change of its sub-parts (e.g. the tops of buildings in aerial imagery). In this paper we have used video sequences but we wish to construct a model that could be used to learn representations from static images. However, such a model should allow images generated as a result of the motion of human body parts to be efficiently characterized despite the fact that no highly specific assumptions about human motion are built into the structure of the model.

2 Other Approaches

We are interested in learning models for the sub-parts of image objects in addition to the ways in which these sub-parts undergo relative translation and transformation from example data. Localized image transformations in examples of objects occur when objects undergo deformations, when classes of objects naturally vary in configuration and also occur when fairly rigid sub-parts of an object are able to move in relative position. Here we describe some examples of approaches to these problems that have been described in the literature.

One way to model changes in an objects appearance within an image is to directly model image transformations such as translations, rotations, shearing and warping. In [1], these transformations are modelled using random variables indicating *global* image transformations. Extending this approach to allow multiple pre-transformation *latent* images allows these methods to be used to extract

meaningful data clusters from images in the presence of background clutter. However, these techniques cannot easily model the fact that numerous spatially localized transformation of an object often occur simultaneously.

In [2], rigid and non rigid facial motions were represented based on local parametric models of image deformations and transformation. Here it was shown that modelling the majority of the face as a plane provided a reasonable model. The parameters of the deformation and translation of local image masks specified for each location were then found using robust regression techniques and gradient descent optimization. In related work [3], this model was extended to a cardboard person representation for articulated image motion such as human walking. Here again, the local images that were used to represent the sub-parts of the model were specified a priori. In contrast, in our approach, we learn local images.

In contrast to Active Contour Models [4] and Active Appearance Models [5], our approach requires less human intervention for the initialization of parameters encoding shape information. In Active Contour or Active Appearance modelling approaches relatively specialized parameterizations are used to encode contour and shape information. In our approach, appearance is modelled within spatially localized latent images. Information concerning an object's shape is encoded within the parameters of a probability model that specifies the distribution for allowable spatially localized transformations of local latent images and this parameterization does not need significant hand adjustment for new shapes.

In [6] a decomposition was proposed to model human dynamics in video sequences. In this work blobs of pixels corresponding to coherent objects in motion are found using mixtures of Gaussians applied to spatio-temporal image gradients and optionally, pixel color values. The motion of the blobs is then modelled hierarchically, first using a Kalman-Filter and then using Hidden Markov Models. In contrast to this and other blob based techniques we learn a number of latent images, their transformations and cropings for spatially localized positions in the image.

Further, numerous approaches have been described in which stick figure models are embedded into underlying models of human figures [7,8,9]. The explicit modelling of the human form used in these techniques allow fewer images to be required to estimate the parameters of these models. In contrast to these approaches we are interested in models that are more generally applicable to images that are not of human figures. But, we wish to be able to learn parameters characteristic of properties such as body joint locations.

3 Montages of Transformed Mixtures of Gaussians

In our approach we extend the Transformed Mixtures of Gaussians (TMG) model in a number of ways. We briefly review the TMG model as we describe these extensions. First, we break an image up into a grid of smaller images or patches. We shall use i and j to denote the locations of these smaller images within

the rows and columns of the grid. Within each cell of the grid there exists a TMG model. Each grid location has a model with C_{ij} clusters that consist of discrete variables indexed by $c_{ij} \in \{1, \dots, C_{ij}\}$. For a basic TMG formulation of the models in each of these grid cells, each cluster c_{ij} would have mixing proportions $P(c_{ij}) = \pi_{c_{ij}}$. Each class c then indexes a probability distribution over a latent image \mathbf{z}_{ij}

$$p(\mathbf{z}_{ij}|c_{ij}) = \mathcal{N}(\mathbf{z}_{ij}; \boldsymbol{\mu}_{c_{ij}}, \boldsymbol{\Phi}_{c_{ij}}) \quad (1)$$

Thus, one can consider each mean $\boldsymbol{\mu}_{c_{ij}}$ as a possible latent image for this grid position in the montage, with $\boldsymbol{\Phi}_{c_{ij}}$ representing a diagonal covariance matrix specifying the variability of each pixel in the latent image. Each position in the grid also has associated a transformation index $l_{ij} \in \{1, \dots, L_{ij}\}$ representing a set of sparse transformation matrices $\boldsymbol{\Gamma}_1, \dots, \boldsymbol{\Gamma}_L$. These matrices model the transformations and cropping from the slightly larger latent image clusters $\boldsymbol{\mu}_{ij}$. We shall only encode vertical and horizontal translation transformations prior to cropping of the latent image. However, other more complex transformations such as rotation and shearing can also be encoded using these sparse transformation matrices. The conditional density of the observed image x_{ij} given the latent image z_{ij} and transformation l_{ij} can then be written as

$$p(\mathbf{x}_{ij}|\mathbf{z}_{ij}, l_{ij}) = \mathcal{N}(\mathbf{x}_{ij}; \boldsymbol{\Gamma}_{l_{ij}} \mathbf{z}_{ij}, \boldsymbol{\Psi}_{ij}), \quad (2)$$

where $\boldsymbol{\Psi}_{ij}$ is a diagonal covariance matrix that specifies a noise model on the observed pixels for the patch at location i, j . These transformation matrices shall be held constant. Finally, a uniform probability distribution is assigned to each of the possible transformation l . We shall use the following notation $p(l_{ij}) = p_{l_{ij}}$. In this simple montage model, the joint distributions for each location in the grid are independent and take the form

$$\begin{aligned} P(\mathbf{x}, l, \mathbf{z}, c)_{ij} &= (p(\mathbf{x}|l, \mathbf{z})p(l)p(\mathbf{z}|c)p(c))_{ij} \\ &= \pi_{c_{ij}} p_{l_{ij}} \mathcal{N}(\mathbf{z}_{ij}; \boldsymbol{\mu}_{c_{ij}}, \boldsymbol{\Phi}_{c_{ij}}) \mathcal{N}(\mathbf{x}_{ij}; \boldsymbol{\Gamma}_{l_{ij}} \mathbf{z}_{ij}, \boldsymbol{\Psi}_{ij}) \end{aligned} \quad (3)$$

Figure 3 illustrates the TMG model as a Bayesian network [10] and contrasts this with a montage of TMG models.

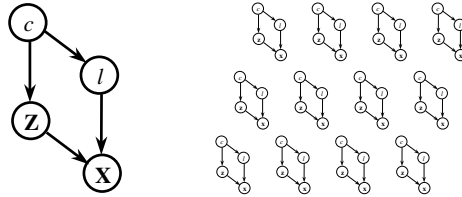


Fig. 3. Bayesian Networks for a single TMG model vs. a montage of independent TMG models.

4 Extending an Independent TMG Montage Model

Here we discuss a number of ways to relax the independence assumption for the TMG models in the image grid. We would like to couple the TMG models in each cell of the image grid so that the transformations and classes are not drawn from independent distributions. Such prior models are necessary to encode constraints on the global shape of an object. For this procedure, one could think of grouping the class and transformation index into one composite variable (c_{ij}, l_{ij}) as well as integrating out \mathbf{z}_{ij} , within each point on the grid. Additionally, to reduce the number of free parameters in the coupling we propose introducing a "coarse scale" version of (c_{ij}, l_{ij}) labelled $(c'_{ij}, l'_{ij})'$. We use a convolution of a smoothing filter

Additionally, to reduce the number of free parameters involved with representing relationships between classes and transformations within differing grid locations, we introduce variables $(c, l)_{ij}$ representing *coarse scale* spatial translation variables for finer scale $(c, l)_{ij}$. In our model we hold the conditional distribution $p((c, l)_{ij} | (c, l)'_{ij})$ fixed so that this distribution acts as a *smoothing* filter encoding a Gaussian or some other finite extent window. One can think of this operation as *sub-sampling* l_{ij} , the variables indicating the position of the sift invariant feature c_{ij} . In this way, fewer parameters are needed to encode the conditional distribution $p((c, l)'_{ij} | t)$ as opposed to a conditional distribution $p((c, l)_{ij} | t)$ on the finer scale variables $(c, l)_{ij}$ requiring more parameters. This strategy thus reduces the number of images needed as examples to estimate the parameters in the model. Figure 4 illustrates graphically (a) the initial TMG formulation, (b) an illustration of the model used for inference in a standard TMG, (c) the introduction of a coarse scale variable and (d) a simplified form of (c) that shall be used to more clearly present our illustrations of how the independent montage model may be coupled spatially.

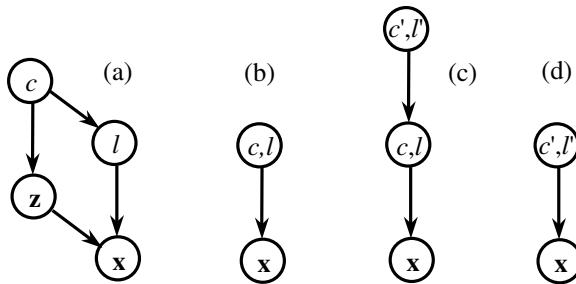


Fig. 4. (a) The initial TMG model. (b) Grouping of c, l and integrating out \mathbf{z} . (c) Introducing a coarse scale position variable. (d) Further simplifying the graph for a clearer presentation of subsequent coupling techniques.

Our approach of using a shift invariant, spatially localized model followed by a sub-sampling of the "likelihood" for the position variable l_{ij} for each c_{ij} is

similar to the approach taken in [11]. However in [11], spatially localized feature detectors and the sub-sampling of the feature detector’s outputs are employed within a multilayer perceptron architecture. Similar techniques employing this smoothing approach have also been used in the earlier work of [12]. Our approach differs in that we use formally defined, graphical probability models [13] as the underlying architecture. This allows partial computations in the graph to be interpreted as likelihoods.

There are a number of ways to link the hidden variables of independent TMGs arranged in a grid using a graphical model. Figure 5 illustrates some possible types of models. The upper right corner of Figure 5 illustrates the coupling as a Markov Random Field (MRF) [14]. The lower left and right corners of Figure 5 illustrate tree structured coupling with various degrees of sub-structure. In the next section we derive the update equations for an EM algorithm in a tree structured model.

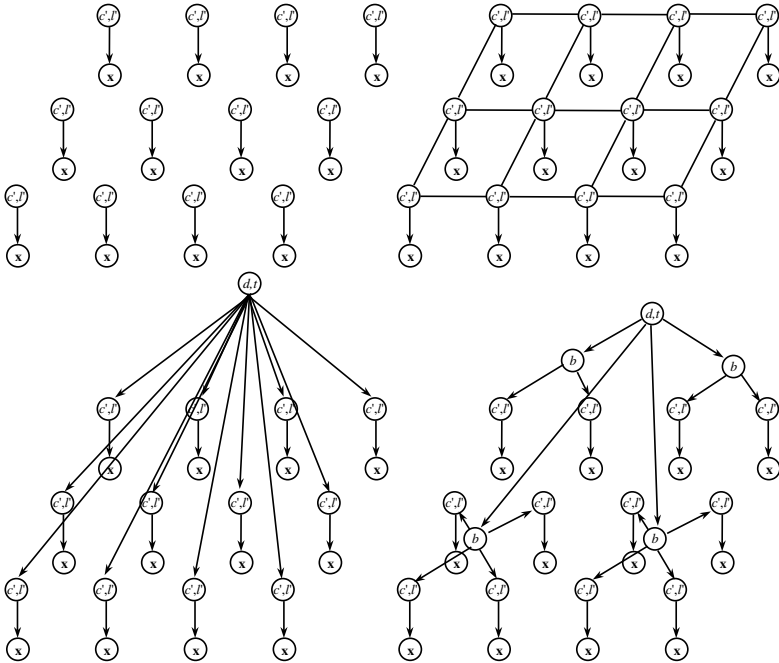


Fig. 5. A graphical comparison of an independent TMG montage model with various prior models coupling the independent TMG models.

5 An EM Algorithm for a Tree Structured TMG Montage Model

Consider the Bayesian network in the lower left corner of Figure 5 describing a tree model. Such a model describes a joint distribution and in this particular tree we shall write this distribution as:

$$p(\mathbf{X}, \mathbf{z}_{ij}, (c, l)_{ij}, (c, l)'_{ij}, t, d) \\ = \prod_{ij} p(\mathbf{x}_{ij} | (l, \mathbf{z})_{ij}) p(\mathbf{z}_{ij} | c_{ij}) p((c, l)_{ij} | (c, l)'_{ij}) p((c, l)'_{ij} | t) p(t | d) P(d). \quad (4)$$

In this model there is a discrete variable d at the root of the tree encoding the idea that the entire composite image composed of the leaves of the tree could be assigned the class d . The variable t encodes the notion that there can be sub-classes or variations of this main class. For example, left, right front and back profiles of a human subject. We shall use \mathbf{X} to denote the complete image and use \mathbf{x}_{ij} and \mathbf{z}_{ij} to denote the observed and latent images or variables for a patch at location (i, j) in the grid. $(c, l)_{ij}$ is a composite class and position variable. c_{ij} can be thought of as the indices for the possible latent images at each location (i, j) . While the variables l_{ij} encode the location of the window that crops a view out of the latent images encoded within $\boldsymbol{\mu}_{c_{ij}}$, the mean parameters for $p(\mathbf{z}_{ij} | c_{ij})$. Finally, $(c, l)'_{ij}$ are the coarser scale versions of $(c, l)_{ij}$ as described in the previous section. We shall now derive the Expectation Maximization (EM) equations to update the variables in this model.

5.1 The E Step

It is helpful to think of the tree model in terms of a Bayesian Network, so that the E-step of an EM algorithm can be thought of as an instance of probability propagation within the graph. The resulting computations can then be decomposed as follows. For each patch on the grid we can compute the likelihoods

$$p(\mathbf{x}_{ij} | (c, l)_{ij}) = \int_{\mathbf{z}_{ij}} p(\mathbf{x}_{ij} | (c, l)_{ij}, \mathbf{z}_{ij}) d\mathbf{z}_{ij} \\ = \int_{\mathbf{z}_{ij}} \mathcal{N}(\mathbf{x}_{ij}; \boldsymbol{\Gamma}_{l_{ij}} \mathbf{z}_{ij}, \boldsymbol{\Psi}_{ij}) \mathcal{N}(\mathbf{z}_{ij}; \boldsymbol{\mu}_{c_{ij}}, \boldsymbol{\Phi}_{c_{ij}}) d\mathbf{z}_{ij} \quad (5) \\ = \mathcal{N}(\mathbf{x}_{ij}; \boldsymbol{\Gamma}_{l_{ij}} \boldsymbol{\mu}_{c_{ij}}, \boldsymbol{\Gamma}_{l_{ij}} \boldsymbol{\Phi}_{c_{ij}} \boldsymbol{\Gamma}_{l_{ij}}' + \boldsymbol{\Psi}_{ij})$$

Given an image and a label d_k we can then compute $p(t, d_k, \mathbf{X}_k)$

$$p(t, d_k, \mathbf{X}_k) = \sum_{(c, l)'_{ij}} \sum_{(c, l)_{ij}} p(t, d_k, \mathbf{X}_k, (c, l)_{ij}, (c, l)'_{ij}) \\ = p(t | d_k) \sum_{(c, l)'_{ij}} p((c, l)'_{ij} | t) \sum_{(c, l)_{ij}} p((c, l)_{ij} | (c, l)'_{ij}) \prod_{ij} p(\mathbf{x}_{ij_k} | (c, l)_{ij}) \quad (6)$$

Using probabilistic message passing [10], one can also efficiently compute $p((c, l)'_{nm}, t, d_k, \mathbf{X}_k)$ for each position $(i, j) = (n, m)$ in the grid.

$$p((c, l)'_{nm}, t, d_k, \mathbf{X}_k) = p(t|d_k)p((c, l)'_{nm}|t) \sum_{(c, l)'_{ij} \neq nm} p((c, l)'_{ij}|t) \sum_{(c, l)_{ij}} p((c, l)_{ij}|(c, l)'_{ij}) \prod_{ij} p(\mathbf{x}_{ij_k}|(c, l)_{ij}) \quad (7)$$

We can also compute $p((c, l)_{nm}, \mathbf{X}_k)$ efficiently using message passing as follows:

$$p((c, l)_{nm}, \mathbf{X}_k) = \sum_t p(t|d_k) \sum_{(c, l)'_{ij}} p((c, l)'_{ij}|t) \sum_{(c, l)_{ij} \neq nm} p((c, l)_{ij}|(c, l)'_{ij}) \prod_{ij} p(\mathbf{x}_{ij_k}|(c, l)_{ij}) \quad (8)$$

From (8) we can obtain:

$$\begin{aligned} p(c_{nm}, \mathbf{X}_k) &= \sum_{l_{nm}} p((c, l)_{nm}, \mathbf{X}_k), \\ p(\mathbf{X}_k) &= \sum_{(c, l)_{nm}} p((c, l)_{nm}, \mathbf{X}_k), \end{aligned} \quad (9)$$

$$\begin{aligned} p((c, l)_{nm}|\mathbf{X}_k) &= p((c, l)_{nm}, \mathbf{X}_k)/p(\mathbf{X}_k), \\ p(c_{nm}|\mathbf{X}_k) &= p(c_{nm}, \mathbf{X}_k)/p(\mathbf{X}_k), \\ p(l_{nm}|c_{nm}, \mathbf{X}_k) &= p((c, l)_{nm}, \mathbf{X}_k)/p(c_{nm}, \mathbf{X}_k). \end{aligned}$$

Additionally, in the M-step we shall need to compute $\boldsymbol{\Omega}_{(cl)_{ij}}$ the covariance matrices of \mathbf{z} given (c, l) and \mathbf{x} for each grid location. Importantly, these matrices are independent of \mathbf{X} and can thus be computed once before each E-step.

$$\boldsymbol{\Omega}_{cl,ij} = (\boldsymbol{\Phi}_{c_{ij}}^{-1} + \boldsymbol{\Gamma}_{l_{ij}}' \boldsymbol{\Psi}_{ij}^{-1} \boldsymbol{\Gamma}_{l_{ij}})^{-1} \quad (10)$$

We then compute the following expectations:

$$E[\mathbf{z}_{ij}|(c, l)_{ij}, \mathbf{X}_t] = \boldsymbol{\Omega}_{(cl)_{ij}} \boldsymbol{\Gamma}_{l_{ij}}' \boldsymbol{\Psi}_{ij}^{-1} \mathbf{x}_{t_{ij}} + \boldsymbol{\Omega}_{(cl)_{ij}} \boldsymbol{\Phi}_{c_{ij}}^{-1} \boldsymbol{\mu}_{c_{ij}}, \quad (11)$$

$$E[\mathbf{z}_{ij}|c_{ij}, \mathbf{X}_t] = \sum_{l_{ij}} p(l_{ij}|c_{ij}, \mathbf{X}_k) E[\mathbf{z}_{ij}|(c, l)_{ij}, \mathbf{X}_k], \quad (12)$$

$$\begin{aligned} E[(\mathbf{z}_{ij} - \boldsymbol{\mu}_{c_{ij}}) \circ (\mathbf{z}_{ij} - \boldsymbol{\mu}_{c_{ij}})|c_{ij}, \mathbf{X}_k] &= \sum_{l_{ij}} p(l_{ij}|c_{ij}, \mathbf{X}_k) \\ &\left[(E[\mathbf{z}_{ij}|(c, l)_{ij}, \mathbf{X}_k] - \boldsymbol{\mu}_{c_{ij}}) \circ (E[\mathbf{z}_{ij}|(c, l)_{ij}, \mathbf{X}_k] - \boldsymbol{\mu}_{c_{ij}}) + \text{diag}(\boldsymbol{\Omega}_{(cl)_{ij}}) \right], \end{aligned} \quad (13)$$

and

$$\begin{aligned}
E[(\mathbf{x}_{ij} - \boldsymbol{\Gamma}_{l_{ij}} \mathbf{z}_{ij}) \circ (\mathbf{x}_{ij} - \boldsymbol{\Gamma}_{l_{ij}} \mathbf{z}_{ij}) | \mathbf{X}_k] &= \sum_{(cl)_{ij}} p((c, l)_{ij} | \mathbf{X}_k) \\
&\left[(\mathbf{x}_{ij} - \boldsymbol{\Gamma}_{l_{ij}} E[\mathbf{z}_{ij} | (c, l)_{ij}, \mathbf{X}_k]) \circ (\mathbf{x}_{ij} - \boldsymbol{\Gamma}_{l_{ij}} E[\mathbf{z}_{ij} | (c, l)_{ij}, \mathbf{X}_k]) \right. \\
&\quad \left. + \text{diag}(\boldsymbol{\Gamma}_{l_{ij}} \boldsymbol{\Omega}_{(cl)_{ij}} \boldsymbol{\Gamma}_{l_{ij}}') \right] \quad (14)
\end{aligned}$$

5.2 The M Step

Using the $\langle \cdot \rangle$ notation to denote $\frac{1}{K} \sum_k (\cdot)$ we can update the parameters of the tree model in the following way.

$$\tilde{p}((c, l)_{nm}' | t) = \frac{\langle p((c, l)_{nm}', t, d_k, \mathbf{X}_k) \rangle}{\langle p(t, d_k, \mathbf{X}_k) \rangle} \quad (15)$$

$$\tilde{p}(t | d_k) = \frac{\langle p(t, d_k, \mathbf{X}_k) \rangle}{\langle p(d_k, \mathbf{X}_k) \rangle} \quad (16)$$

The updates of the other parameters follow in a similar manner to the regular TMG model.

$$\tilde{\boldsymbol{\mu}}_{n_{ij}} = \frac{\langle p(c_{ij} = n | \mathbf{X}) E[\mathbf{z}_{ij} | c_{ij} = n, \mathbf{X}_k] \rangle}{\langle p(c_{ij} = n | \mathbf{X}) \rangle}, \quad (17)$$

$$\tilde{\boldsymbol{\Phi}}_{n_{ij}} = \text{diag} \left[\frac{\langle p(c_{ij} = n | \mathbf{X}_t) E[(\mathbf{z}_{ij} - \boldsymbol{\mu}_{c_{ij}}) \circ (\mathbf{z}_{ij} - \boldsymbol{\mu}_{c_{ij}}) | c_{ij} = n, \mathbf{X}_k] \rangle}{\langle p(c_{ij} = n | \mathbf{X}_k) \rangle} \right], \quad (18)$$

$$\boldsymbol{\Psi}_{ij} = \text{diag}(E[(\mathbf{x}_{k_{ij}} - \boldsymbol{\Gamma}_{l_{ij}} \mathbf{z}_{ij}) \circ (\mathbf{x}_{k_{ij}} - \boldsymbol{\Gamma}_{l_{ij}} \mathbf{z}_{ij}) | \mathbf{X}_k]). \quad (19)$$

To avoid overfitting it is sometimes useful to let entries in $\boldsymbol{\Psi}_{ij}$ and $\boldsymbol{\Phi}_{ij}$ that fall below some ϵ be equal to ϵ . Similarly, one can also let entries in the conditional probability tables that fall below some ϵ' be equal to ϵ' and then re-normalize the conditional distribution.

6 Results and Analysis

In this section we examine the behavior of a tree structured montage trained using EM as described in Section 5. We illustrate MAP reconstruction images under various conditions so as to inspect the quality of the representations learned by the model. We compare MAP reconstruction images for the tree montage with MAP reconstruction images obtained from a transformed mixture of Gaussians model applied to the entire image. We then examine the MAP reconstruction images for the montage applied to test data of the same person walking with differing articulation of the body parts, slightly different lighting and coarse alignment of the underlying figure. We present results for a simple pose recognition task and illustrate the use of the model for tracking people within crowds possessing significant background activity. We show how the approach could be used to automate the control of an active security camera.

6.1 Comparing Montages with Large Transformed Mixture of Gaussians

In Figure 6 we compare our tree montage using four latent images per patch with one large TMG trained on the same data using six, global latent classes. The top row of Figure 6 shows a sequence of 5 images of walking motion. The middle image shows the MAP TMG approximation. The bottom row shows the MAP TMG montage with a tree structured prior. For this segment of the sequence the single large TMG models the images with two of the six latent classes, coarsely approximating the underlying change in shape. Figure 6 also illustrates how the classes found by the single large TMG suffer for combinatorial problems when modelling the complex motions of the sub-parts of an object.



Fig. 6. (top row) Five images from an image sequence. (middle row) The single, large, six class TMG MAP approximation. (bottom row) The four class per grid cell TMG tree montage, MAP approximation.

6.2 Behavior of the Model Using Test Data

The top row of Figure 7 illustrates images from the test sequence of a figure walking to the right. The articulation of the book and cup are different from the training data. The global alignment of the figure in the image varies differently from the training data. Further, the lighting conditions also have been varied slightly. The middle row consists of the MAP reconstruction of the testing data using the tree montage with parameters fit using the differing training data of figures walking to the right. The bottom row illustrates the MAP reconstruction

images of the figure walking to the right illustrated in the top row. However, in this bottom row the tree montage was trained on images of figures walking to the left. This row thus illustrates the way in which the model attempts to "explain" the right walking pose using features that were learned from the left walking pose. The fact that these MAP reconstruction images do not fit the data well and indeed look more like pieces of a left walking figure is a positive sign that the model has not over-generalized. Notice that the MAP images in the



Fig. 7. (top row) Examples of right facing images used to test the model. (middle row) the MAP image for the TMG tree montage model trained on *different* right facing images. (bottom row) The MAP image for the TMG tree montage model trained on left facing images.

bottom row are, in almost all cases missing a head, with darker portions of the image approximated by small feet templates pointing in the opposite direction. Not surprisingly, the MAP reconstruction images for unseen data significantly different from the original training data are reminiscent of photomosaics [15, 16]. Photomosaics are produced by combining many small photographs to form a single larger image that becomes more recognizable when viewed from a distance. As such, these smaller images act as a form of half-toning [17]. More importantly, the MAP reconstruction images in Figure 7 provide insight. We have used a shallow tree for this experiment. Additional sub-structure in the tree as illustrated in Figure 5 would constrain the model so as to further reduce the probability of physically unrealistic configurations under the model.

6.3 Learning Poses from Data

For the task of learning representations for different activities of people from full body images of people walking, the previous section illustrated how our model is able to capture features for arms, legs, feet, hands and the head. Our model also learns likely sub-poses of the underlying object. This information is embedded within the highest level discrete variable of the tree model. The



Fig. 8. (left) A sample from the first sub-type of the TMG tree montage model. (right) A sample from the second sub-type of the TMG tree montage model.

left and right figures of Figure 8 illustrate two "sub-poses" that were learned by our model when 193 training cases from the video sequence of the figure walking to the right were used to train the model. The images were of size 80x90 pixels. Four classes were used at the highest level in the tree and EM converged to two degenerate poses. 16x16 pixel latent images were used with 10x10 pixel images cropped out of the latent image to compose the montage. The images were obtained by sampling from the model given the highest level classes. Here again, a tree with greater depth would likely produce higher quality samples.

6.4 Recognizing Poses

Here we present results on a test image sequences also consisting of human figures walking while simultaneously manipulating hand held objects. The top row of Figure 7 illustrates samples of the test data for the figure walking to the right while Figure 9 illustrates samples of the test data for the figure walking to the left. As discussed previously, these test images differed from the training images with respect to the relative articulation of the figure, the coarse alignment of the figure in the image and the lighting conditions are slightly different.

Using the test sets of 193 images of walking to the left and 193 images of walking to the right the models trained on the left and right walking images were used to compute the marginal likelihood for each of test images. Table. 1 shows a confusion matrix for pose recognition on the test data. Importantly, when the errors were analyzed further, it was found that most of the misclassifications occurred at the start and end of the image sequence. This effect is likely due in part to the higher density of example data for direct side sub-poses versus example data for sub-poses of the figure facing slightly away from the camera or slightly toward the camera. In contrast, for the nearest neighbor classifier

the errors were the most dense in the second half of the sequence walking left sequence. This can be explained by the fact that the coarse centering of the data was skewed to the right in the example data, while skewed to the left in the testing data. This misalignment was less of an issue for the TMG montage model.



Fig. 9. Images of a figure walking left used to test the model.

Table 1. Confusion matrix for pose classification of the test data using the TMG tree montage and using nearest neighbors (*italics*). The vertical column is the actual pose and the horizontal column is the classified pose.

actual / classified	right	left
right	.83 (<i>.92</i>)	.17 (<i>.08</i>)
left	.03 (<i>.25</i>)	.97 (<i>.75</i>)

6.5 Tracking and the Control of an Active Camera

Consider the task of automating the control of a movable security camera. In this task the operator of the camera wishes to briefly use the camera to follow a particular person and then allow an algorithm to take over control of the camera and continue to track the person. For this task, the subject of the security camera might be found within a crowd of people who are also in motion. We have gathered image data from various realistic locations (eg. shopping malls, busy street corners and train stations) where complex dynamic background are common.

We have used high resolution wide angle views to allow us to simulate the movement of a smaller field of view of an active security camera. Figure 10 (left) illustrates one frame from a wide angle view of a high traffic area outside a subway station. The subject that we wish to track has been indicated by hand over the first 60 frames leading up to the occlusion of the subject by another figure. The subject is indicated using a white rectangle. The four images on the right half of Figure 10 illustrate the results of a simple greedy tracking algorithm



Fig. 10. (left) A wide angle view of a surveillance area. (right) Tracking a subject despite substantial background motion and occlusion.

that utilizes a tree montage trained on the initial 60 frames prior to the occlusion. The model has the same structure as in the previous tests. We use the marginal likelihood under the model to select one of 24 possible discrete global translations in x and y of the complete tree model. Our implementation of the same greedy tracking algorithm using various metrics of simple template based similarity all have difficulty tracking objects in such scenes.

6.6 Implementation Issues

Our current Matlab implementation of the tree montage with the parameters describe above requires approximately 45 seconds to evaluate the marginal likelihood for a given discrete global transformation. The same amount of time is required for our code to do an incremental Expectation step (E-step) for one example of training data. Our incremental M-step requires approximately 2 seconds. Our Matlab research code is inefficient and as such we perform incremental E-steps and evaluate marginal likelihoods used in our tracking implementation in parallel using a cluster of networked computers.

7 Conclusions and Discussion

In this paper we have introduced the TMG montage and presented a number of ways such a montage can be constructed. We have presented Markov Random Field and Tree structured methods of spatially coupling TMG montages. We have derived and presented the EM algorithm for a simple tree model with a fixed structure. We have presented results in which our model has clustered images of walking human subject into poses and extracted features consisting of body sub-parts in an "unsupervised" fashion. We have presented some "supervised" classification results and presented an active camera application for which our model is able to deal with significant background movement and complexity where simpler tracking approaches would clearly fail.

The TMG montage model can do a reasonable job of modelling background as observed in our MAP images. For some applications such as tracking, this property is not desirable. In cases where there is significant change in the background relative to changes in the foreground, a variance discrepancy will be produced and will be observable in the noise parameter of the latent image. The resulting discrepancy in variance can be used to produce an image "mask" indicating the boundary of the foreground object. This property allows the model to do a better job than simpler techniques for tracking figures with complex backgrounds.

However, for some types of data the background variance is intrinsically low relative to the foreground. This can also arise when little training data is available. In these cases, a pre-processing step can be employed to identify static elements of the background and the EM algorithm for training the model can be modified relatively easily to account for the boundary of the foreground figure.

To address the issues of the choice of tree structure and size, techniques such as Structural EM [18] could be used to learn trees with arbitrary size and structure. Random variables corresponding to lighting conditions, occlusions and foreground vs. background can be incorporated into such a framework in a straightforward manner.

References

1. B. J. Frey and N. Jojic, "Estimating mixture models of images and inferring spatial transformations using the em algorithm," *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, June 1999.
2. M. J. Black and Y. Yacoob, "Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motions," *Proc. International Conference on Computer Vision*, pp. 374–381, 1995.
3. M. J. Black S. Ju and Y. Yacoob, "Cardboard people: A parameterized model of articulated image motion," *Proc. International Conference on Face and Gesture Recognition*, pp. 38–44, 1996.
4. A. Blake and M. Isard, *Active Contours*, Springer-Verlag, 1998.
5. G.J. Edwards T.F.Cootes and C.J.Taylor, "Active appearance models," *Proc. European Conference on Computer Vision*, vol. 2, pp. 484–498, Springer, 1998.
6. C. Bregler, "Learning and recognizing human dynamics in video sequences," *Proc. IEEE (CVPR)*, June 1997.
7. R. Rosales, M. Siddiqui, J. Alon, and S. Sclaroff, "Estimating 3d body pose using uncalibrated cameras," *Proc. IEEE (CVPR)*, 2001.
8. C.J. Taylor, "Reconstruction of articulated objects from point correspondences in a single uncalibrated image," *Proc. Computer Vision and Image Understanding (CVIU)*, pp. 80:349–363, 2000.
9. H. Lee and Z. Chen, "Determination of 3d human body postures from a single view," *Computer Vision Graphics and Image Processing (CVGIP)*, pp. 30:148–168, 1985.
10. J. Pearl, *Probabilistic Inference in Intelligent Systems*, Morgan Kaufmann, San Mateo, California, 1988.

11. Y. Bengio Y. LeCun, L. Bottou and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, November 1998.
12. K. Fukushima, "Neocognitron: A hierarchical neural network capable of visual pattern recognition," *Neural Networks*, vol. 1, pp. 119–130, 1988.
13. M. Jordan, *Learning in Graphical Models*, Kluwer, Dordrecht, 1998.
14. S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-6, no. 6, November 1984.
15. A. Finkelstein and M. Range, "Image mosaics," *Proc. EP'98 and RIDT'98, St. Malo, France*, vol. 15, no. 10, pp. 1042–1052, March 1998.
16. R. Silvers and M. Hawley, *Photomosaics*, New York: Henry Holt and Company, 1997.
17. K. Knowlton and L. Harmon, "Computer-produced grey scales," *Computer Graphics and Image Processing*, vol. 1, pp. 1–20, 1972.
18. N. Friedman, "The bayesian structural em algorithm," *Fourteenth Conf. on Uncertainty in Artificial Intelligence (UAI)*, 1998.