Entropy Driven Hierarchical Search for 3D Human Pose Estimation

Ben Daubney
B.Daubney@Swansea.ac.uk
Xianghua Xie
X.Xie@Swansea.ac.uk

Department of Computer Science
Swansea University
UK, SA2 8PP

Abstract

In this work a hierarchical approach is presented to efficiently estimate 3D pose from single images. To achieve this the body is represented as a graphical model and optimized stochastically. The use of a graphical representation allows message passing to ensure individual parts are not optimized using only local image information, but from information gathered across the entire model. In contrast to existing methods the posterior distribution is represented parametrically. A different model is used to approximate the conditional distribution between each connected part. This permits measurements of the Entropy, which allows an adaptive sampling scheme to be employed that ensures that parts with the largest uncertainty are allocated a greater proportion of the available resources. At each iteration the estimated pose is updated dependent on the Kullback Leibler (KL) divergence measured between the posterior and the set of samples used to approximate it. This is shown to improve performance and prevent over fitting when small numbers of particles are being used. A quantitative comparison is made using the HumanEva dataset that demonstrates the efficacy of the presented method.

1 Introduction

3D Human pose estimation from monocular images or video is an extremely difficult problem. Currently there are two main approaches to solving this problem, the first is to learn a direct mapping from image features to 3D pose [1, 19], the second is to first extract 2D pose as an intermediate stage and then 'lift' this to a 3D pose [2]. The limitation with both of these approaches is that they are only applicable to poses that are similar to those represented in the original training set, e.g. walking. It is unlikely they will scale to extract arbitrary 3D poses. Contrary to this, in the domain of 2D pose estimation current state-of -the-art methods have been shown capable of detecting poses that are much more varied [3, 8, 14]. This has been achieved using generative models built around the Pictorial Structures representation [10]. Recent work in 2D pose estimation has shown that by first clustering 2D poses detection rates can be significantly improved [14, 15]. Often these clusters have a semantic meaning, for example different orientations of the person being detected [14]. We suggest that rather than simply clustering 2D poses a much more direct approach is to learn a generative 3D model that can be orientated and projected at arbitrary scales and orientations, allowing 3D pose to be directly extracted from single images.

BMVC 2011 http://dx.doi.org/10.5244/C.25.31

The main challenges are how to represent this 3D generative model and how to efficiently search the high dimensional pose space. Current 2D approaches that uniformly discretize the search space will be unsuitable for 3D pose estimation as the search space is significantly larger and unconstrained, so a stochastic approach will be required. When searching for 3D pose, probable limb configurations are completely dependent on the state of central part, i.e. the global orientation and position of the torso, which we define as the root node of the model. This is as neighbouring parts are connected at fixed locations and have fixed lengths. This leads us to suggest a hierarchical approach to 3D pose estimation starting from a hypothesized root node state.

The obvious limitation with this approach is that if the hypothesized state of the root node is incorrect, the entire approach will fail. However, if we consider the original search space proposed in the work of Felzenszwalb and Huttenlocher [8] for 2D pose estimation, the search space was divided into a grid over image positions, scales and orientations. In total 1.5×10^7 image likelihood evaluations were performed in a single frame, where a single evaluation is performed for each hypothesized part location. In this work we show that for a single hypothesized root node the pose of the remaining parts can be adequately searched in ≈ 5000 image likelihood evaluations, this would allow 3000 hypothesized root node states to be evaluated without increasing the computation compared with that of [8] (assuming image evaluation is the dominant computational expense).

In this work we address how to efficiently search and represent the 3D pose space given a hypothesized root node state. For evaluation we assume that the root node of the solution is known. Contrary to many stochastic approaches, where the posterior is represented as a set of samples [17, 21], ours is always represented parametrically. Samples are drawn from it to permit the update and evaluation of the posterior. A parametric model has several useful benefits. Firstly, it allows the posterior to be represented in a compact form. Secondly, it will be of benefit if applied to a tracking by detection approach [2, 9]. Given independent pose estimates in each frame, methods such as Virterbi searching or Continuous Belief Propagation could be used to link independent detections together. Furthermore, as the proposed solution is iterative an approach could be envisaged where more complex part detectors are employed after each iteration as further knowledge is extracted and the search space is reduced.

Typical hierarchical approaches assume that a part of the body can first be accurately located, based on which the remaining parts can be located sequentially, moving outwards from the centrally located part to the model extremities. The problem with this approach is that it results in parts that are only locally optimized, thus often recovering the incorrect pose across the entire body. In this work we reformulate the hierarchical search as that of Bayesian inference performed over a graphical model with a single fixed node. Initial outward optimization can be viewed as message passing between connected nodes. The benefit of this methodology, differing from typical approaches, is that information is also back propagated to parts higher up the model hierarchy before optimization is performed, ensuring a correct global solution is more likely to be recovered.

The use of graphical models to estimate pose has gained much popularity. Parts of the body are represented by the graph nodes and the conditional dependencies between these parts are represented by the edges. To estimate 2D pose the search space can be represented by pixel locations in the image and pose can be estimated using methods such as Dynamic Programming [8], Belief Propagation [11] or Loopy Belief Propagation [20]. For 3D pose estimation the search space is too large to be uniformly discretized and stochastic methods such as Non-Parametric Belief Propagation [21], Markov-Chain Monte Carlo [17], Variational MAP [13] or Partitioned Sampling [18] can be employed. A problem with these

approaches is that they often rely on weakly constrained parts where joint locations are not forced to coincide [13, 21], this was recently shown to be too unconstrained for accurate 3D pose estimation [6]. In this work we examine how a graphical model can be utilized given a part's state is known *a priori*, effectively fixing a node in the graph. To the best of our knowledge this problem has not previously been investigated for 3D pose estimation using this framework and provides insight into existing hierarchical approaches, in particular the benefit gained by passing information between parts.

Most stochastic approaches are iterative, this allows a fixed number of particles to efficiently search the solution space by focusing the particles into progressively smaller regions with high likelihoods [7, 13, 21]. A problem with these techniques is how to diffuse the particles between iterations. Often a single covariance is used which is shrunk by a predetermined fixed amount across iterations [7, 13]. In this work the posterior distribution is represented as a set of Gaussian Mixture Models (GMM). Samples are then drawn from these models, weighted by their beliefs and the models updated given observations obtained from the image. Over a number of iterations the models converge to a single maximum without the need to artificially shrink any covariances.

In this paper the focus is on formulating standard hierarchical methods as Bayesian inference over a graph where a node is fixed and on increasing the efficiency of the method so less samples are required to search the pose space. Firstly, outward messages are calculated via importance sampling making the search strategy more efficient. Secondly, calculating the Entropy of each model a method is provided to adaptively create a sampling scheme to ensure that a higher proportion of resources are used to search for parts that have greater uncertainty. Finally, a method using the KL divergence between the model and the set of samples used to approximate it is presented to prevent over fitting when low numbers of samples are being used. This is shown to substantially improve the accuracy of the method. Quantitative results are provided using the HumanEva dataset.

2 Model Representation

The body is represented by a set of ten parts, each part has a fixed length and connected parts are forced to join at fixed locations. The conditional distribution between two connected parts, which describes the likely orientation of a part given that of the part to which it is connected, is modeled by first learning a joint distribution using a Gaussian Mixture Model (GMM) $p(\mathbf{x}_i, \mathbf{x}_j|\theta_{ij})$, where $\mathbf{x}_i$ and $\mathbf{x}_j$ is the state of the ith and jth part respectively and θ_{ij} is the set of model parameters. As each model is represented using a GMM the model parameters are defined as $\theta_{ij} = \{\lambda_{ij}^k, \mu_{ij}^k, \Sigma_{ij}^k\}_{k=1}^K$, where K is the number of components in the model and $\lambda_{ij}^k, \mu_{ij}^k, \Sigma_{ij}^k$ represent the kth component's weight, mean and covariance respectively. For efficiency all covariances used to represent limb conditionals are diagonal and can be partitioned such that $\Sigma_{ij}^k = diag(\Lambda_{ii}^k, \Lambda_{jj}^k)$ and likewise $\mu_{ij}^k = \left(\mu_i^k, \mu_j^k\right)$.

Given a value for $\mathbf{x}_j$ (e.g. a sample) the conditional distribution $p(\mathbf{x}_i|\mathbf{x}_j, \theta_{ij}^k)$ is calculated from the joint distribution $p(\mathbf{x}_i, \mathbf{x}_j|\theta_{ij})$. This is also a GMM with model parameters $\{\lambda_i^k, \mu_i^k, \Lambda_{ii}^k\}_{k=1}^K$. The component weights are proportional to the marginal distribution $\lambda_i^k \propto p(\mathbf{x}_j|\theta_{ij}^k)$, which is calculated from the normal distribution $p(\mathbf{x}_j|\theta_{ij}^k) = \lambda_{ij}^k \mathcal{N}(\mathbf{x}_j; \mu_j^k, \Lambda_{jj}^k)$. Note this conditional model is different to typical approximations used, when the conditional model is approximated by $p(\mathbf{x}_{ij}|\theta_{ij})$, where $\mathbf{x}_{ij}$ is the value of $\mathbf{x}_i$ in the ***local frame*** of reference of $\mathbf{x}_j$ [8, 21].

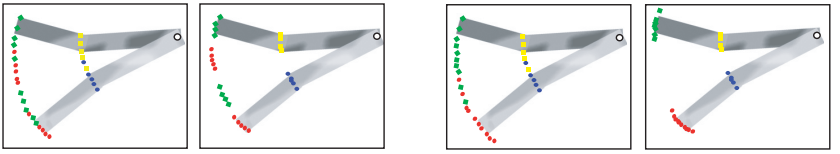

Figure 1: Hypothetical two part example highlighting the difference in modeling different parts independently (a) - (b) and using conditional models (c) - (d). (a) and (c) show the prior model and (b) and (d) the model after a number of iterations. Both limb conditionals are represented by a two component mixture model where each component is represented by different colors. Whilst the conditional model can represent each observational mode by a single mixture component (d), the independent (unconditional) model can not and as such 'phantom' modes appear (b).

The state of each part $\mathbf{x}_i$ represents a quaternion rotation that defines its orientation in the global frame of reference, which here is defined to be that of the torso. The location of a part is dependent on the state of the part to which it is attached.This is as parts are forced to be connected at fixed joint locations.

The benefit of learning a full conditional model between neighbouring parts is two fold. Firstly, consider an approach where two independent particle filters are used to locate the upper and lower arm respectively and suppose that each distribution has two modes. As the two particles filters are modeled independently how will the samples drawn from the second filter know to which mode in the first particle filter they are correlated with? This is illustrated in Fig. 1 where we see in (a) and (b) the effect of using an approximate limb conditional, $p(\mathbf{x}_{ij}|\theta_{ij})$, and in (c) and (d) where a full conditional is learnt, $p(\mathbf{x}_i|\mathbf{x}_j,\theta_{ij})$, and the modes are correlated. In (a) and (b) as the conditional distribution between the two connected parts is not modeled extra modes appear in the posterior. This is as the distribution, $p(\mathbf{x}_{ij}|\theta_{ij})$, is modeled in only the local frame of reference of the upper part.

The second benefit of using this method to model conditional distributions is that different GMM components learnt in quaternion space correspond to different spatial locations in $\mathbb{R}^3$. This is illustrated in Fig. 2 where samples for each part have been generated using the proposed distributions. A covariance has then been fitted in $\mathbb{R}^3$ to the set of samples generated by each GMM component in quaternion space. As can be seen, in general, a single component modeled in quaternion space corresponds to a separate location in euclidian space. It also demonstrates that this representation can clearly capture multiple modes. Details of how to sample from the limb conditionals are provided in the following section.

3 3D Pose Estimation

The human body is defined by a graph, which is assumed to be a tree, where the set of n nodes $v_i \in \mathcal{V}$ represents the set of parts used to ensemble the object and $\{v_i, v_j\} \in \mathcal{E}$ represent the edges that connect the nodes of the graph together. Given proposal values for each node $X = \{\mathbf{x}_1, .., \mathbf{x}_n\}$ and corresponding observations $Z = \{z_1, .., z_n\}$, the posterior can be calculated as

$$p(X|Z,\theta) = \prod_{\{i,j\}\in\mathcal{E}} p(\mathbf{x}_i|\mathbf{x}_j,\theta_{ij}) \prod_{i\in\mathcal{V}} p(z_i|\mathbf{x}_i), \tag{1}$$

where $p(\mathbf{x}_i|\mathbf{x}_j,\theta_{ij})$ are the limb conditionals which represent the model prior and were described in the previous section, and $p(z_i|\mathbf{x}_i)$ are observational likelihoods that describe how well the state of a given part explains the observed image. The observational likelihoods use

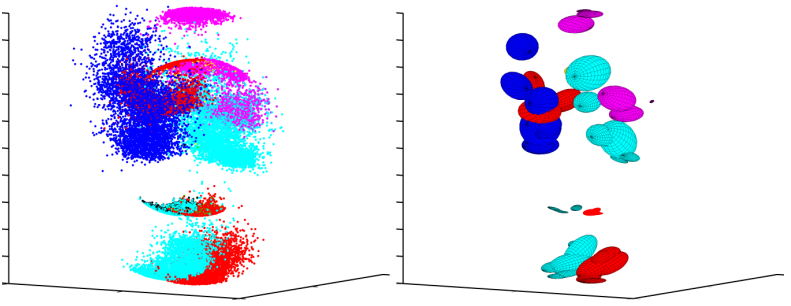

Figure 2: A visualization of the conditional distributions $p(\mathbf{x}_i|\mathbf{x}_j,\theta_{ij}^k)$ for each part. (left) individual samples for each part. (right) fitting a covariance to the samples generated from each covariance. Red - left foot, left knee and right elbow. Dark blue - right hand. Purple - left elbow and head. Light blue - right foot, right knee and right hand.

a combination of edge and color cues and are described in Section 4. The posterior for a single part, also known as the belief, can be calculated for each part using message passing

$$p(\mathbf{x}_i|Z) = p(z_i|\mathbf{x}_i)\prod_{v_j\in C_i} p(\mathbf{x}_i|Z_j), \tag{2}$$

where $v_j \in C_i$ are the set of nodes connected to the ith node and Z_j represents the observations for the subtree containing v_j, created by removing the edge $\{v_i, v_j\}$. The messages are represented by the term $p(\mathbf{x}_i|Z_j)$, which is a prediction over $\mathbf{x}_i$ given the observations Z_j.

In this work we assume the state of a single part of the model is known *a priori* which fixes a node in the graph. This node will be referred to as the root node, $\mathbf{x}_r$, and is located at the top of the tree with all branches coming from it. In this work it is defined as the torso. Whilst the choice of node picked to be the root node is arbitrary, intuitively it should be the part that has the least variation in its state.

There are two types of messages: Outwards messages propagating from the root node out to the limb extremities, called leaf nodes, and inwards messages passed from the leaf nodes to the root. The messages are represented using sample sets and outward messages are propagated using importance sampling.

To calculate an outwards message $p(\mathbf{x}_i|Z_j)$, where i is the child of j, first consider a set of samples used to represent the message from the kth node to the jth where k is the parent of j. This set of N_j uniformly weighted samples is defined as $p(\mathbf{x}_j|Z_k) \approx \left[\mathbf{x}_j^n, \pi_j^n\right]_{n=1}^{N_j}$ where π represents a sample's weight. Each sample is then weighted proportional to the observational likelihood $\pi_j^n \propto p(z_j|\mathbf{x}_j^n)$ following which a set of N_i samples can then be selected to approximate $p(\mathbf{x}_i|Z_j) \approx [\mathbf{x}_i^n, \pi_i^n]_{n=1}^{N_i}$. This is achieved by selecting a sample $\mathbf{x}_j^n$ with likelihood $\propto \pi_j^n$ and then calculating the conditional distribution $p(\mathbf{x}_i|\mathbf{x}_j^n,\theta_{ij}^k)$ as described in the previous section. A sample $\mathbf{x}_i^n$ is then generated from this GMM by first selecting a component with probability $k^* = \lambda_i^k$, following which a sample is generated from the selected component $\mathbf{x}_i^n \sim \mathcal{N}(\mathbf{x}_i; \mu_i^{k^*}, \Lambda_{ii}^{k^*})$. The set of N_i samples are assigned an equal weight.

The sample $\mathbf{x}_j^n$, from which the sample $\mathbf{x}_i^n$ is drawn conditioned on, is called its ancestor and this method of sampling is called ancestral sampling [4]. Its purpose in this work is to grow the search space out from the central part of the model, exploring regions of the pose space that have a higher likelihood. The key difference between this approach and that

of Partitioned Sampling (PS) [13] or Markov Chain Monte Carlo (MCMC) [17] is that in our approach a given sample only represents the state of the limb it is drawn for. This is in contrast to PS or MCMC where a sample must represent the state of all parts of the model, but specific components of the sample are changed depending on the part being optimized. We can achieve this as the conditional model links different modes across connected parts. This allows samples to be drawn for each part independently whilst maintaining conditional dependence between connected parts.

Inwards messages are computed from the leaf nodes towards the root. Each sample passes information back to its ancestors which is computed as

$$p(\mathbf{x}_i^n|Z_j) = \frac{1}{|N_j(\mathbf{x}_i^n)|} \sum_{m \in N_j(\mathbf{x}_i^n)} p(z_j|\mathbf{x}_j^m) \prod_{v_c \in C_j} p(\mathbf{x}_j^m|Z_c), \tag{3}$$

where j is the child of i, $m \in N_j(\mathbf{x}_i^n)$ is the set of samples that are ancestors of $\mathbf{x}_i^n$, $|N_j(\mathbf{x}_i^n)|$ is the number of ancestors and $v_c \in C_j$ is the set of child nodes of v_j. The marginal or belief can then be calculated by combining the incoming messages with the local observation likelihood as described by Eqn. 2

$$p(\mathbf{x}_i^n|Z) = p(z_i|\mathbf{x}_i^n) \prod_{v_j \in C_i} p(\mathbf{x}_i^n|Z_j). \tag{4}$$

The distribution of the samples used to represent the belief at each node is provided by the outwards messages and their corresponding weights are provided by the inwards messages combined with local observational likelihoods. The expectation value for each part can then be calculated as

$$E[\mathbf{x}_i] = \sum_{n=1}^{N} p(\mathbf{x}_i^n|Z). \tag{5}$$

Given a fixed set of particles, T, a sampling scheme is employed to distribute the particles across the different parts of the model. Intuitively parts that have greater uncertainty require more particles since a larger space must be searched. A measure of the uncertainty is provided by the Entropy of the distribution $h(\theta_{ij}) = \int p(\mathbf{x}_i|\theta_{ij}) \ln p(\mathbf{x}_i|\theta_{ij}) d\mathbf{x}_i$. Whilst there is not an exact analytical expression to calculate the Entropy of a GMM, an approximation can be used to express the GMM as a single Gaussian [12]

$$\hat{\mu}_i = \sum_{k=1}^{K} \lambda_{ij}^k \mu_i^k, \tag{6}$$

$$\hat{\Lambda}_{ii} = \sum_{k=1}^{K} \lambda_{ij}^k \left(\Lambda_{ii}^k + (\mu_i^k - \hat{\mu}_i)(\mu_i^k - \hat{\mu}_i)^T \right). \tag{7}$$

The Entropy can then be calculated as $h(\theta_{ij}) = \ln \sqrt{(2\pi e)^d |\hat{\Lambda}_{ii}|}$, where d is the dimension of $\mathbf{x}_i$. The number of samples assigned to each part N_i is then given by $N_i \propto e^{h(\theta_{ij})}$, which is equivalent to distributing the particles to each part proportional to the hypervolume encompassed by the covariance. The minimum number of samples assigned to a single part is set as 1% of the total number of samples, to ensure no parts are assigned zero samples.

Once the beliefs have been calculated for each node, represented as a set of weighted samples, the model is updated. The parameters of each mixture component are reestimated using the weighted samples that are drawn from it. Before each model is updated, simulated annealing is applied to the particles' weights so that $\approx 60\%$ of the samples would be discarded if resampling has taken place [7]. A new sampling strategy is then estimated and a new set of samples is drawn from the updated model. The beliefs for the new samples are

calculated and the model again updated. This process is iterated a set number of times. If the normalized weight of a component falls below a predetermined threshold, the component is pruned from the mixture model. This is to allow the samples to be focused towards areas of the posterior with a higher likelihood and will eventually converge to a single solution.

However, model over fitting may occur if very few samples are drawn from the components. This is because with too few samples it poorly represents the underlying distribution. This problem however can be largely alleviated by checking the model fitting, for instance, using the Kullback Leibler (KL) divergence to measure how well a sample set represents a mixture component. The model could then be updated depending on this measure, based on the intuition that if the sample set poorly represents the mixture component the model can not be confidently updated using this sample set.

Let $[\mathbf{x}_i^n]_{n\in N(\theta_{ij}^k)}$ denote the set of samples drawn from the kth component of the mixture model θ_{ij}. The initial model update is calculated using the samples and their weights $\{\theta_{ij}^k\}_{est} = \{\lambda_{ij}^k, \mu_i^k, \Lambda_{ii}^k\}_{est}$ as described previously. However, using the same samples, the mean and covariance are recomputed assuming each sample has a uniform weight $\{\theta_{ij}^k\}_{uni} = \{\mu_i^k, \Lambda_{ii}^k\}_{uni}$. The KL divergence is then calculated between this distribution and the component that the samples were drawn from ($\{\theta_{ij}^k\}_t = \{\lambda_{ij}^k, \mu_i^k, \Lambda_{ii}^k\}_t$) denoted by $KL\left(\{\theta_{ij}^k\}_t | \{\theta_{ij}^k\}_{uni}\right)$. The component parameters are then updated as

$$\begin{aligned}
\{\mu_i^k\}_{t+1} &= (1-w)\{\mu_i^k\}_t + w\{\mu_i^k\}_{est} \\
\{\Lambda_{ii}^k\}_{t+1} &= (1-w)\{\Lambda_{ii}^k\}_t + w\{\Lambda_{ii}^k\}_{est} \\
\{\lambda_{ij}^k\}_{t+1} &= (1-w)\{\lambda_{ij}^k\}_t + w\{\lambda_{ij}^k\}_{est}
\end{aligned} \tag{8}$$

where $w = e^{-KL(f|g)}$ (details of how to compute the KL divergence between two Gaussian distributions are provided in [12]). If the sample set accurately represents the component, the KL divergence will be zero and the model will be updated to the new estimated model parameters. However, if it poorly represents the prior, the KL divergence will be very large and the model will not be updated. Note that the components' weight must be renormalized once this update has been carried out for all components in the mixture model.

4 Observational Likelihoods

In this section we describe how the observational likelihoods $p(\mathbf{z}_j|\mathbf{x}_j)$ are calculated. A part is represented by a rectangular patch with two image cues exploited, edges and color. Edge cues are extracted using a set of M overlapping HOG features [8] placed along the edges of the part. Each feature is represented as a single normalized histogram of the local image gradients at that location and they are combined such that $p(\mathbf{z}_j|\mathbf{x}_j)_{edge} = \frac{1}{M}\prod_{m=1}^{M} H(\theta_\perp)$, where $H(\theta_\perp)$ returns the value in the histogram bin that is perpendicular to the edge of the part. Color is exploited by placing a bounding box at the location of the root node and then learning a foreground model using the pixel values within the box and a model for the background using pixels outside the box. The models are learnt using a GMM. This creates a very crude and noisy foreground probability map. The likelihood is then calculated as the average foreground probability value encompassed by the part. The individual likelihoods for each cue are then combined as $p(\mathbf{z}_j|\mathbf{x}_j) = p(\mathbf{z}_j|\mathbf{x}_j)_{edge} p(\mathbf{z}_j|\mathbf{x}_j)_{col}$.

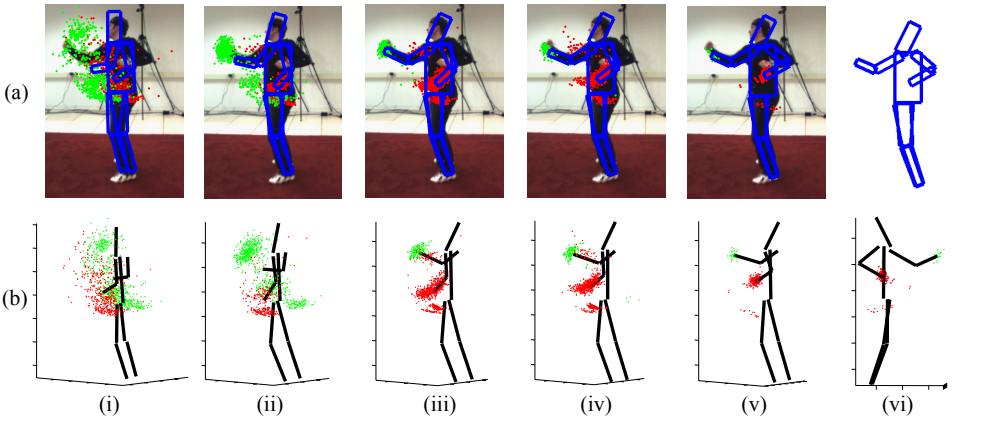

Figure 3: Example frames showing the expected pose after iterations 1 (i), 3 (ii), 5 (iii) , 7 (iv) and 10 (v) in 3D (b) and projected onto the image (a). The samples are also shown for the left wrist (red) and right wrist (green). (b)(vi) shows the final 3D pose viewed from a different orientation.

5 Experiments and Results

The model for each part was learnt using the Train partition of the HumanEva dataset using $\approx$ 4500 frames of data taken across all 3 subjects and all 5 different actions (Box, Gesture, ThrowCatch, Walk, Jog) contained in the set. Hence the solution space for correct pose is not constrained to a single action. For quantitative evaluation a test set was created from the Validation partition of the HumanEva dataset. This was composed of 100 randomly selected frames from each action category selected across all color views (C1, C2, C3), so that 500 frames were used in total. The root node and orientation was set using the pelvis marker data and the scale was set as the maximum distance between the head and the feet using the ground truth provided. This scale was often inaccurate (e.g. if the subject was squatting), however, was used so all experiments are easily reproducible. Inaccurate scale information is particularly detrimental on the 3D pose estimation error. In each frame the algorithm was iterated 10 times. It is assumed that the person is viewed under orthographic projection.

Typical 3D pose estimation results showing particle convergence are given in Figs. 3 and 4. Fig. 3 (iii) shows that the model is more than capable of supporting multiple modes as shown in the particle distribution for the left wrist. As can be seen the search space is quickly reduced and the model converges to the correct pose. A common cause of error in the reconstruction is due to depth ambiguities, for example in the lower right leg in Fig. 4 (b) (vi). These errors are most common where the prior model is less constrained. In the above example the leg is free to rotate backwards and forwards as this motion would be necessary to perform walking or jogging. However, lateral motion of the leg would be much more constrained. In the same example the right arm is correctly located in front of the person. This as the prior would not permit the forearm to be located elsewhere and still project to the same location.

Quantitative results are provided for the reprojected 2D error and the 3D pose reconstruction error using different numbers of samples in Fig. 5 using the average Euclidian distance between the markers as proposed in [22] for 3D pose and [16] for 2D errors, where the left/right limbs are switched and the smallest reprojection error used. Note that each sample represents a hypothesized state of a single part not of the entire body. For comparison we

also present results when the KL divergence technique is not used and examine the effect of message passing. This shows that when messages are not passed and each part is updated using only local information the performance deteriorates. By using the KL divergence the error is greatly reduced as the model is prevented from over fitting when less samples are used. It would be expected that the curves representing full message passing and the KL update method would converge as more samples are used, as the KL-divergence between the samples drawn and the models would approach zero.

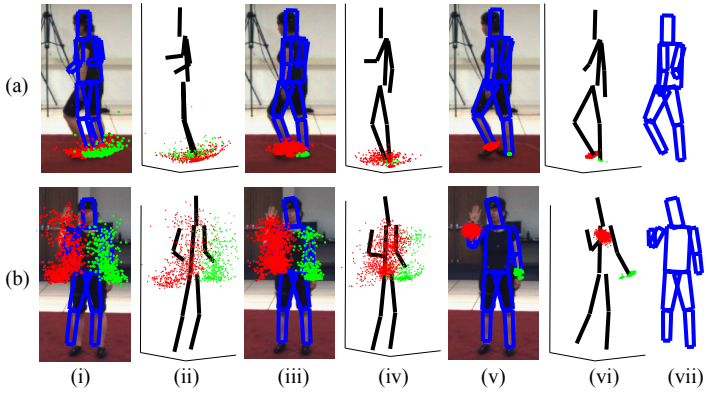

Figure 4: Example frames after iterations 1,4 and 10 showing the expected pose after each iteration in 3D and projected onto the image. Red samples show samples for the right foot (a) and right wrist (b), blue points depict the samples for the opposing parts.

The quantitative results given in Figs. 5(a) and 5(b) also show strong correlation between the reprojected 2D error and the 3D pose reconstruction error. This suggests that 3D pose estimation may be better solved by simultaneously locating individual parts of the model in the 2D image plane and extracting the underlying model in 3D space rather than treating them as independent sequential processes as is commonly adopted in the literature. The method presented in this work is a promising technique to achieve this.

6 Conclusions

In this paper a hierarchical method, formulated as inference over a graphical model with a fixed node, has been presented to estimate 3D pose from single images. A key motivation is

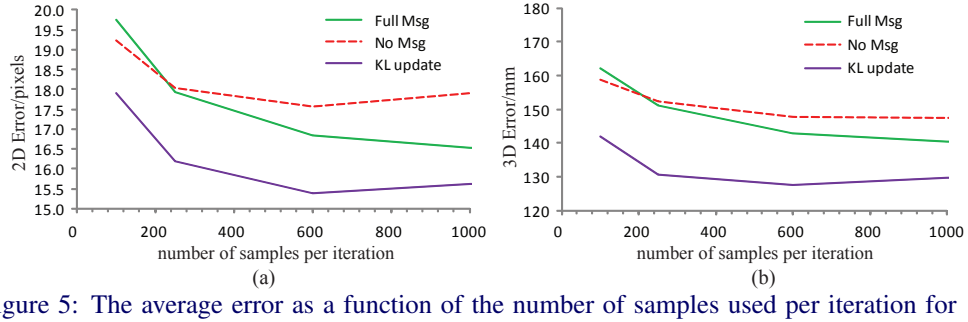

Figure 5: The average error as a function of the number of samples used per iteration for the 2D reprojection error (a) and 3D reconstruction error (b). Full Msg and No Msg show the error without the KL divergence method with and without message passing. KL update shows the error using the KL divergence update method with message passing.

that it allows a solution to be found where parts are forced to be joined at fixed locations and have fixed lengths. To demonstrate the efficacy of the approach we have applied it to single frame pose estimation. We have shown that 3D pose estimation is possible whilst employing more general model priors, compared to those where the solution space is often restricted to a single action (e.g. walking). Furthermore, this has been achieved using relatively simple appearance models. This shows that the proposed prior, whilst more general, still maintains information that is crucial in resolving image reconstruction ambiguities. An important component is the use of a parametric representation, which allows us to effectively integrate this work into a tracking by detection framework, utilizing stronger part detectors.

References

[1] Ankur Agarwal and Bill Triggs. Recovering 3d human pose from monocular images. *PAMI*, 28(1):44–58, 2006.

[2] M. Andriluka, S. Roth, and B. Schiele. Monocular 3d pose estimation and tracking by detection. In *CVPR*, 2010.

[3] Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. Pictorial structures revisited: People detection and articulated pose estimation. In *CVPR*, 2009.

[4] C. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2006. ISBN 0387310738.

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[6] B. Daubney and X. Xie. Estimating 3d pose via stochastic search and expectation maximization. In *AMDO*, 2010.

[7] J. Deutscher and I. Reid. Articulated body motion capture by stochastic search. *IJCV*, 61:185–205, 2005.

[8] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 61:55–79, 2005.

[9] V. Ferrari, M. Marin-Jimenez, and A. Zisserman. Progressive search space reduction for human pose estimation. In *CVPR*, 2008.

[10] M. A. Fischler and R. A. Elsclager. The representation and matching of pictorial structures. In *IEEE Transactions on Computer, 22(1)*, pages 67–92, 1973.

[11] J. Gao and J. Shi. Multiple frame motion inference using belief propagation. In *IEEE International Conference on Automatic Face and Gesture Recognition*, 2004.

[12] J. R. Hershey and P. A. Olsen. Approximating the kullback leibler divergence between gaussian mixture models. In *ICASSP*, pages 317–320, 2007.

[13] G. Hua and Y. Wu. Variational maximum a posteriori by annealed mean field analysis. *PAMI*, 27(11):1747–1761, 2005.

[14] S. Johnson and M. Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *BMVC*, 2010.

[15] Xiangyang Lan and Daniel P. Huttenlocher. A unified spatio-temporal articulated model for tracking. In *CVPR*, pages 722–729, 2004.

[16] Xiangyang Lan and Daniel P. Huttenlocher. Beyond trees: Common-factor models for 2d human pose recovery. In *ICCV*, pages 470–477, 2005.

[17] Mun Wai Lee, Student Member, and Isaac Cohen. A model-based approach for estimating human 3d poses in static images. *PAMI*, 28:905–916, 2006.

[18] John Maccormick and Michael Isard. Partitioned sampling, articulated objects, and interface-quality hand tracking. In *ECCV*, 2000.

[19] Ryuzo Okada and Stefano Soatto. Relevant feature selection for human pose estimation and localization in cluttered images. In *ECCV*, 2008.

[20] D. Ramanan and D. Forsyth. Finding and tracking people from the bottom up. In *CVPR*, 2003.

[21] L. Sigal, S. Bhatia, S. Roth, M.J. Black, and M. Isard. Tracking loose-limbed people. In *CVPR*, 2004.

[22] L. Sigal, A. Balan, and M. Black. Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *IJCV*, 87: 4–27, 2009.