Monocular Image 3D Human Pose Estimation under Self-Occlusion

Ibrahim Radwan[1] Abhinav Dhall[2] Roland Goecke[1,2]
[1]Vision & Sensing Group, HCC Lab, ESTeM, University of Canberra, Australia
[2]IHCC Group, RSCS, Australian National University, Australia
ibrahim.radwan@Canberra.edu.au, abhinav.dhall@anu.edu.au, roland.goecke@ieee.org

Abstract

In this paper, an automatic approach for 3D pose reconstruction from a single image is proposed. The presence of human body articulation, hallucinated parts and cluttered background leads to ambiguity during the pose inference, which makes the problem non-trivial. Researchers have explored various methods based on motion and shading in order to reduce the ambiguity and reconstruct the 3D pose. The key idea of our algorithm is to impose both kinematic and orientation constraints. The former is imposed by projecting a 3D model onto the input image and pruning the parts, which are incompatible with the anthropomorphism. The latter is applied by creating synthetic views via regressing the input view to multiple oriented views. After applying the constraints, the 3D model is projected onto the initial and synthetic views, which further reduces the ambiguity. Finally, we borrow the direction of the unambiguous parts from the synthetic views to the initial one, which results in the 3D pose. Quantitative experiments are performed on the HumanEva-I dataset and qualitatively on unconstrained images from the Image Parse dataset. The results show the robustness of the proposed approach to accurately reconstruct the 3D pose form a single image.

1. Introduction

The automatic recovery of 3D human pose from a single, monocular image is a very challenging problem in computer vision due to the strong ambiguities of estimating human body articulations from a single image caused by the deformation of an articulated body, self-occlusion, large degrees of freedom and different poses for the same person performing actions under different environmental constraints. A solution to this problem may lead to applications in pedestrian detection and tracking, automotive safety, video annotation, human action recognition and graphic aspects.

Recent work in 3D pose reconstruction from 2D images can be categorised into (1) data-driven and (2) structure from motion based techniques. Data-driven methods predict the 3D poses via mapping 3D joints from the image observations or the 2D joint locations [1, 5, 6]. In contrast, structure from motion methods extract the 3D points from the corresponding 2D points in different images for the same subject [18, 19] through estimating the camera parameters, bone lengths and parts directions. Here, we combine these two techniques to benefit from the advantages of both and obviating their disadvantages.

Given an input image, we start with an off-the-shelf 2D body part detector (e.g. Yang and Ramanan [20]) to estimate the 2D joint locations. Due to its limitations in the presence of self-occlusion, we add an inference step handling self-occlusion, improving the initial input to the 3D pose estimation. Subsequently, we project a 3D model onto the 2D joints, which results in a very ambiguous 3D pose. By enforcing kinematic and geometric constraints, we reduce this ambiguity. To solve for any remaining ambiguity, we use the Twin-GP regression method [5] to predict novel views from the initial one and project the 3D model onto the initial and synthetic views to estimate the relative depth of the parts. Finally, to solve the problem of the part directions, we 'borrow' the unambiguous parts of the synthetic views to correct ambiguous parts of the initial view.

The **key contributions** of this paper are:

- A framework for automatic 3D human pose reconstruction from a single 2D image, evaluated on difficult human pose scenarios.
- A self-occlusion reasoning method to improve the initialisation step and to increase the accuracy of state-of-the-art 2D pose estimation, evaluated on a publicly available dataset.
- A method to automatically solve for the ambiguity of the parts' direction instead of having to rely on user input as in [18].

2. Background

While there is a plethora of literature on 3D human pose reconstruction from 2D images, we focus our attention on research to predict the 3D pose using data-driven or structure from motion approaches.

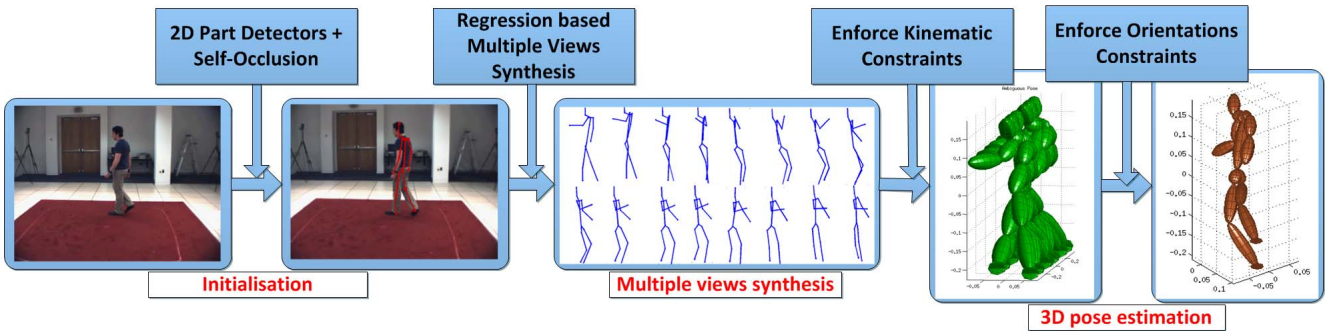


Figure 1: **Outline of our processing pipeline**: (From the left:) Starting with an input image, 2D part detectors and self-occlusion reasoning are applied. Next, multiple synthetic views are generated from the initial view. Then, structure from motion is used to enforce kinematic constraints and reduce the ambiguity. Finally, orientation constraints are enforced from the synthetic views onto the initial input in order to generate the 3D pose.

The key components of data-driven methods are the choice of image descriptor, the shape of output and the prediction phase. Generally, the steps are: (1) extract features from a 2D image and then (2) infer the 3D pose by using the predefined predictors. Predictors based on sparse regression, nearest neighbours and feature descriptors such as SIFT have been employed to allow an automatic recovery of 3D poses from 2D images. Agarwal *et al.* [1, 3] used silhouettes as an image descriptor followed by the relevance of a sparse regression method to map the extracted silhouettes to 3D pose and applied it to human tracking [2]. Bo *et al.* [6] utilised different robust image descriptors (*e.g.* multi-level block of SIFT feature descriptor) and predict the 3D pose in a Bayesian framework. They employed conditional Bayesian mixtures of experts to map from the image observations to the corresponding 3D joint locations directly.

Recently, Bo *et al.* [5] proposed a twin Gaussian process regression method to estimate the 3D pose from Histogram of Oriented Gradients (HOG) and HMAX feature descriptors. A limitation of these methods is their need for huge amounts of training data to model the predictors and represent the variability of appearance of different people and viewpoints. Experiments based on these methods have typically only been performed on lab-controlled data. In this paper, we propose to reconstruct the 3D pose of a human body in images / frames in an *uncontrolled environment*. In addition, the spatial information is not guaranteed to be empirically captured using image descriptors in methods such as [5]. These limitations are overcome by our method as the part localisation for real-world images is based on Pictorial Structures (e.g. [20]), which explicitly applies shape constraints. Moreover, our method still only needs a single input image as the previous techniques. However, the earlier methods' focus (e.g. [5]) on mapping from image observation to 3D reduces the robustness and generalisation. It suffers in cases of dynamic backgrounds and images with hallucinated and occluded parts. In contrast, our method is accurately reconstructing 3D poses for scenes with cluttered, changing background and uncontrolled body parts.

Structure from motion based methods have gained much popularity. The 3D pose is estimated from the 2D correspondences through a set of images / frames via applying a factorisation method, which was firstly introduced in [17] for reconstructing the 3D pose of a rigid structure. [8] proposed a factorisation method for non-rigid structures by imposing constraints on the object being reconstructed. In an interesting work by Wei *et al.* [19], the 3D pose was recovered for articulated object from multiple images of the same subject in different poses by imposing constraints on the rigid and non-rigid structure to reduce the ambiguity. They combine the rigid and non-rigid structure in a non-linear optimisation framework to estimate the camera parameters and bone lengths. Their method has been extended by Valmadre *et al.* [18] through basic factorisation methods and a linear least squares solution to the parameters. A fundamental criticism of the previous structure from motion based methods is their requirement of multiple images. Further, for finding a solution to the direction of hallucinated and hidden parts, they require manual input from the user. We provide a solution to decode the direction of the ambiguous parts automatically. The positive effect of this is evident from the performance of our method in the experiments.

Estimating 3D pose from 2D images has been investigated in other recent works, e.g. [4, 9], which enforce a temporal consistency to reduce the ambiguity, while we estimate the 3D pose from only a single image. Predicting the 3D pose from point correspondences in a single image has been earlier investigated in [16]. Recently, Simo-Serra *et al.* [15] utilised a similar initialisation step (starting from noisy 2D points), followed by a different inference scheme. They used covariance matrix adaptation (CMA) to sample the 3D pose space, while our proposed method enforces

both kinematic and orientation constraints. Utilising CMA may lead to local minima solutions producing inaccurate 3D hypotheses, while in all of the testing scenarios, our method provided accurate 3D poses.

3. Proposed Method

As shown in Fig. 1, our proposed algorithm can be outlined in three subsequent stages: (1) Initialisation, (2) inferring synthetic views and (3) estimating 3D pose. We adapted the state-of-the-art mixture of parts detectors [20] to initialise the pipeline of our algorithm. Although these detectors are efficient in detecting the articulated body parts, they still fail in the presence of self-occlusion. In the initialisation step, we therefore pursued a small and efficient trick to overcome the problem of self-occlusion (see Section 3.1).

Projecting the 3D model onto the initial view will result in ambiguous poses. We explicitly impose geometric and kinematic constraints to reduce the ambiguity of the 3D pose via pruning those parts that are incompatible with anthropomorphism. However, utilising these constraints only is not sufficient to completely solve the ambiguous parts, especially the direction of the limbs (towards or away from the camera). Thus, to solve the remaining ambiguity, we need more cues about the direction of the body parts. Here, we proposed a novel inference method by generating synthetic (additional) views using pose distributions learned from training data and finally adopted a structure from motion step to estimate the relative depth of different parts from the corresponding points in both initial and synthetic views. This allows solving the problem of the remaining ambiguous poses not only for simple lab-controlled cases (*e.g*. HumanEva datasets [13]), but also for very difficult hallucinated cases as in the Image Parse (IP) dataset [12].

3.1. Initialisation

Given the importance of the initialisation step, we first propose a novel way of dealing with self-occlusion to improve the results of the final pose estimation.

Mixture of Pictorial Structures: Yang and Ramanan [20] perform human pose estimation by representing the human body parts as a mixture of pictorial structure (MoPS) where the nodes are the parts in different orientations. Following the pattern of the notations in [20], the score of a specific pose configuration is:

$$S(I,p,t) = S(t) + \sum_{i \in V} w_i^{t_i} \cdot \phi(I,p_i) + \sum_{ij \in E} w_{ij}^{t_i,t_j} \cdot \psi(p_i - p_j) \quad (1)$$

where $\phi(I,p_i)$ is the HOG descriptor extracted from location p_i in image I, the first sum represents the scores of the image locations against the set of pre-trained appearance templates and the second sum encodes the spring relationships between adjacent parts. Inference is pursued by maximising the score over the locations p and types t.

(a) (b)

Figure 2: Sample results of applying the body part detectors (a) with [21] and (b) with self-occlusion handling.

Self-Occlusion Reasoning for MoPS: In tree structured models, the local scores of children would be correctly traversed to their parents. However in the presence of occlusion (*i.e*. partially or completely), the tree structure turns into a graph and the score may traverse to the wrong parent resulting in missing parts and inaccurate detections, as shown in Fig. 2a. In [11], we proposed a regression based occlusion rectification method. We observed that occlusion detection is more difficult than occlusion rectification. In this paper, we detect occlusion *within* the MoPS inference framework, which encodes the kinematic configurations in a tree. It implicitly assumes that non-adjacent parts are independent, which is violated under self-occlusion [14]. To make the independence assumption hold so that we can use belief propagation, we estimate the occluded parts from their scores. The score of pixel p will be down weighted to $-\infty$ if it leads part i to be detected inaccurately or even missing if that pixel is being occluded by any another part. Under self-occlusion, the score of location p is:

$$\hat{S}(I,p,t) = \begin{cases} -\infty & \text{if } p \text{ is occluded,} \\ S(I,p,t) & \text{otherwise.} \end{cases} \quad (2)$$

To find occluded pixels, we pursue the following scenario: for each part i, select k pixels with maximum scores; obtain its bounding box representing the candidate result of the part; find the maximum overlapping ratio of other parts with part i; if it exceeds a threshold σ and if the score in the location p is smaller than the score of the pixel surrounded by the overlapping region, then handle this part i as an occluded part at the pixel p. As a result, we break the spring, which might be constructed between non-adjacent

Method	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	Total
Yang [21]	90.2	85.4	68.0	47.1	77.1	75.4	67.1	72.9
Yang [21] + self-occlusion handling	**89.8**	**88.2**	**68.8**	**48.1**	**80.5**	**77.2**	**69.8**	**74.6**

Table 1: **Effect of handling self-occlusion in MoPS**: There is a small but consistent improvement in performance over the default MoPS formulation [21], using the probability of correct keypoints (PCK) as the evaluation criterion, as in [21].

parts due to self-occlusion and, thus, the local scores are independent. Then, we use the remaining belief propagation inference process of [20], resulting in more accurate detections (Fig. 2b). In the experiments, we empirically set $k = 5$ and $\sigma = 0.15$. Table 1 shows the improvement due to the self-occlusion reasoning step over the state-of-the-art results. For details of the evaluation protocols, see [20].

3.2. Multiple View Synthesis

For generating an accurate 3D pose, we use the approach of Wei and Chai [19] to project a 3D model onto the vector $\boldsymbol{x}$ of 2D joints that resulted from the previous step. [19] assumes that at least five 2D images are available and uses structure from motion to estimate camera parameters. In contrast, we use only one 2D image, which implies that the camera scale parameter will be unity. To remove the ambiguity for the depth of different parts, we propose to infer multiple synthetic views from the initial one, which enables us to impose new constraints about the space of orientation for each bone, reducing the ambiguity of the 3D poses.

3.2.1 Extracting 3D training Data

In our experiments, all of the training data were collected from the CMU Motion Capture Database[1]. The set of data for each view was collected by selecting 5 frames randomly from each video sequence. Based on the extracted 3D joints for each frame, we measured the heading angle of the human pose and then rotated that 3D pose to extract its 3D points in the 360 polar angles. Projecting the landmarks onto the 2D plane with different orientations led to the 2D points of all joints in all polar angles.

Normalised Skeleton: The usage of the world coordinates in regression often results in bad predictions due to the large variance in the translation and scaling of the different human skeletons pursuing different actions. To achieve a certain level of invariance to the translation and scaling, we carry out the normalisation with one template for each view. The 2D input skeleton is a tree with the c_Hip point as a root, joints represent the nodes and each edge between a parent and its child nodes represent a bone. Mathematically, given $S = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n)$ where $\boldsymbol{x}_i \in R^d$ is an input skeleton with d joints, the normalisation is done by: Firstly, translating each x_i to the origin with the c_Hip joint as a reference point. Secondly, transferring the resulting joints from Cartesian coordinates to polar format such that $X_i = (l_i^{p,c}, \theta_i^{p,c}; i = 1, \cdots, n)$, where $l_i^{p,c} = \parallel x_p - x_c \parallel_2$ is the absolute length of the bone, residing between the parent p and child c pair of nodes, and $\theta_i^{p,c} = \tan^{-1} \frac{x_{p_y} - x_{c_y}}{x_{p_x} - x_{c_x}}$ is the orientation of the bone relative to the horizontal axis. Thirdly, scaling the bone lengths of each skeleton l_i w.r.t. a predefined base skeleton $\boldsymbol{x}_0$ selected for each view. The great benefit of the normalisation step, along with a mitigation of the large variation in scaling and translation, is to fit the input data in a Gaussian distribution.

3.2.2 Multi-view Extension

The normalisation step is applied to all instances, resulting in N samples for each view. Subsequently, in this section, we will construct a specific model to regress from view i to view j. In our experiments, we collect data from the CMU Mocap dataset for 16 views (from 0° to 360° in 22.5° steps). The key idea here is to produce new skeletons from the input instance by means of regression. For this task, we employ the Twin Gaussian Process Regression (Twin-GPR) [5] in a cascaded manner. Finally, we use the constructed models to infer virtual poses from a certain pose.

Recently, Twin-GPR has been used instead of classic regression methods, such as Gaussian process regression and ridge regression, in structured prediction of the 3D pose from image observations. Twin-GPR is a multivariate regression method, which encodes the correlation between both the inputs and outputs. Following [5], we build regression models to generate novel views from the input one.

Given $Z^i = (\boldsymbol{z}_1^i, \cdots, \boldsymbol{z}_n^i)$ and $Z^j = (\boldsymbol{z}_1^j, ..., \boldsymbol{z}_n^j)$ are the normalised instances for two consecutive views i and j (*i.e.* $i = 0$, $j = 22.5$) for n instances, the objective of the regression is to estimate the predictive distribution of an unobserved vector $\tilde{\boldsymbol{z}}^j$ over the observed Z^j data given the input vectors Z^i such that the predictive Gaussian of a test vector will be measured by minimising the divergence between the distribution of the inputs [5]:

$$p(\tilde{\boldsymbol{z}}|Z^j, Z^i = \boldsymbol{z}) \sim \mathcal{N}(\boldsymbol{\mu}^I, C^I) \tag{3}$$

and the distributions of the outputs

$$p(\tilde{\boldsymbol{z}}|Z^j, Z^i = \boldsymbol{z}) \sim \mathcal{N}(\boldsymbol{\mu}^J, C^J) \tag{4}$$

[1] http://mocap.cs.cmu.edu

where $\tilde{\boldsymbol{z}}$ is the normalised vector of the estimated target pose for testing input vector $\boldsymbol{z}$, $\boldsymbol{\mu}^I$ and $\boldsymbol{\mu}^J$ are the mean vectors of the training poses of views I and J, resp., and

$$C^I = \begin{bmatrix} K_I & K_I{}^{\boldsymbol{z}} \\ (K_I{}^{\boldsymbol{z}})^T & K(\boldsymbol{z},\boldsymbol{z}) \end{bmatrix}$$

$$C^J = \begin{bmatrix} K_J & K_J{}^{\tilde{\boldsymbol{z}}} \\ (K_I{}^{\tilde{\boldsymbol{z}}})^T & K(\tilde{\boldsymbol{z}},\tilde{\boldsymbol{z}}) \end{bmatrix}$$

are the positive semi-definite covariance functions, which encode the correlations between training input vectors I and a testing vector $\boldsymbol{z}$, and the correlations between training target vectors J and the estimated target vector $\tilde{z}$ where, K is $N \times N$ matrix for either the input I or the target J with $K_{ab} = K(a,b)$ and each of $K^{\boldsymbol{z}}$ and $K^{\tilde{\boldsymbol{z}}}$ is $N \times 1$ vector for the correlation between a vector $\boldsymbol{z}$, $\tilde{\boldsymbol{z}}$ and the matrix I or J, respectively. The question now is how to compute the distribution in Eq. 4 without obtaining the estimated value for $\tilde{\boldsymbol{z}}$. To this end, we employ the Kullback-Leibler divergence between the two distributions in Eqs. 3 and 4, $D_{KL}(P^I \| P^J)$. Then, BFGS quasi-Newton optimisation is used to minimise the divergence through an iterative process, initialising $\tilde{\boldsymbol{z}}$ with the response of the ridge regressor, trained independently for each of the output vectors.

Cascaded Twin-GPR: Dollar *et al.* [10] proposed an interesting regression method, which gradually reaches the ground truth in a cascaded fashion. In our framework, we regress from an input view to other multiple views. A simple method is to learn the mapping from one view to all other views. However, this increases the complexity of the system as the number of models to be learnt is very large. Inspired by [10], we pose the problem of learning view-specific regression models as a cascaded Twin-GPR problem. Let $Reg(\theta_i, \boldsymbol{z}^i)$ be a function based on Twin-GPR, which maps $\boldsymbol{z}^i \rightarrow \boldsymbol{z}^j$ where $\boldsymbol{z}^i$ is the normalised vector of an input pose, $\boldsymbol{z}^j$ is the vector of the novel view and θ_i is the view of $\boldsymbol{z}^i$. The output of Reg becomes the input of the next iteration and $\theta_i^{'} = \theta_i + \delta$. At every step, δ is added to the view and a pose-specific model is used for regression. Algorithm 1, which is computed N times, outlines the steps for generating novel views from the input one.

Algorithm 1: Cascaded Twin-GPR based synthetic view generation

Require: Input pose $\boldsymbol{z}_i$, view θ_i, step size δ.
Iterations N = $(\theta_j - \theta_i)/\delta$
for view $i \in N$ **do**
  Regression: $\boldsymbol{z}_j = Reg(\theta_i, \boldsymbol{z}_i)$
  Update $\theta_i = \theta_i + \delta$
  Update $\boldsymbol{z}_i = \boldsymbol{z}_j$
end for

3.2.3 Initial View Estimation

To initialise the cascaded regression process (Alg. 1), we estimate the orientation of the initial view. Knowing the initial view of the human pose significantly reduces the ambiguity of the 3D pose reconstruction [4]. A Gaussian Mixture Model (GMM) has been adapted to infer the initial view [7]. The GMM is utilised in a Bayesian framework with maximum likelihood. The data, which has been used to learn the regression models, also have been utilised to train the GMM. Now, we have 16 views (= 16 classes). We partition each class' members into a number of mixtures (empirically, we used 50 in our experiments). Given the input image, in the inference, the orientation of the initial view is determined by the class with the maximum likelihood.

3.3. Estimating 3D Pose

3.3.1 Propagating Ambiguous 3D Poses

To estimate the 3D pose, we start with the 2D joints of the initial view and elevate to 3D pose. The 3D pose is parametrised as a vector $\boldsymbol{v} = [\boldsymbol{v}_1^T, \cdots, \boldsymbol{v}_n^T]$ of n 3D points corresponding to 2D input points $\boldsymbol{u} = [\boldsymbol{u}_1^T, \cdots, \boldsymbol{u}_n^T]$. The 3D pose retrieval can be seen as a solution of a linear system, if multiple input images are available. In contrast, we use only one image and a set of 2D points. We assume the internal camera parameters $\boldsymbol{A}$ to be known. The projection of a point $\boldsymbol{v}_i$ onto $\boldsymbol{u}_i$ may be written as $w_i[\boldsymbol{u}_i^T 1]^T = \boldsymbol{A}\boldsymbol{v}_i$ where w_i is a projection scalar [15]. From the known values of $\boldsymbol{A}$ and $\boldsymbol{u}_i$, we can obtain the projection matrix $\boldsymbol{M}$ of size $2n \times 3n$ that relates 3D points (in a camera coordinate system) to 2D locations. We can then express this matrix for all joints as $\boldsymbol{M}\boldsymbol{v} = 0$. Solving this equation requires more constraints. The kinematic constraints have been enforced via learning the upper and lower bounds of bone angles from the training data as in [19]. This results in an ambiguous 3D pose, such as the one in Fig. 1.

3.3.2 Inferring Disambiguated 3D Pose

To solve the ambiguity and obtain an accurate 3D pose, we followed two subsequent steps. As mentioned before, structure from motion based methods reconstruct the 3D pose via estimating the camera scale, bone length and depth by projecting the 3D model onto the 2D point correspondences in different images. Having only one 2D image implies that the camera scale parameter is 1. Firstly, we remove the ambiguity of the depth for different parts with the help of the synthetic views. Given point correspondences for the input and synthetic views, our aim is to estimate the bone lengths and depths of different parts. The regression step to create multiple synthetic views can result in different bone scales.

To overcome this problem and given that we work with just one image (showing one human body), we can safely constrain the problem by fixing the corresponding bone lengths in all views to be the same as in the initial input image.

Secondly, we need to estimate the relative depth of each part. Valmadre and Lucey [18] compute the magnitude of the depth of each part via a factorisation method starting from a weak perspective projection between the 2D correspondences of different images and then deriving the required parameters by minimising the reconstruction error. Inspired by [18], we utilise the same factorisation approach on the correspondences from the initial view and some of the synthetic views inferring the relative depth of each part.

However, in many cases the ambiguity around the sign of the joint angles still remains. The approach of Valmadre and Lucey [18] failed to solve the ambiguity for many poses with hallucinated parts and, hence, the user was asked to manually determine the direction (*i.e.* either front or back) of the ambiguous parts. In our proposed framework, we developed an efficient solution to this problem. A perspective projection is applied on the basic view of the image. Then, we determine the remaining ambiguous parts $G = (g_1, ...g_l)$, which still may be in either front or back direction. We repeat the previous two steps on all of the synthetic views, where we project the 3D model onto each synthetic view, which results in a 3D model for each view with some parts being ambiguous and others not. We search over all unambiguous parts in the 3D poses, obtained from the synthetic views, which are corresponding to the ambiguous parts G. This enforces the orientation constraints. Then, we iteratively borrow the direction to the 3D pose of the input image until all ambiguities are removed.

In this step, some images require just 2 or 3 instances of synthetic views, while others need all n views. That is why we add one view at a time and stop when all ambiguous parts are removed. The part is still ambiguous if it has two or more possible directions. The big advantage of using structure from motion after regressing multiple views is to prune the noisy predictions introduced by the regression process and to improve the result of the final 3D pose.

4. Experiments

We evaluate the performance of our method in recovering the 3D pose from a single image in different experiments in both quantitative and qualitative ways.

4.1. Data

All data used in training both the cascaded Twin-GPR and the GMM estimating the view of the input pose are collected from the CMU Mocap dataset. We randomly select 5 frames from each sequence of all of the available motion sequences. This results in 14229 frames in total. For each of them, we extract 16 views by rotating the 3D skeleton. We test our approach on different datasets: the HumanEva-I dataset [13] for quantitative evaluations and images from Image Parse dataset [12] for qualitative evaluations.

4.2. Quantitative Evaluation on HumanEva Dataset

The performance of our algorithm is evaluated on the *walking* and *jogging* actions of the HumanEva-I dataset [13]. By using the validation sequences for testing, we show the robustness of our method for recovering the 3D pose. The sequences for training the regression models are extracted from the CMU Mocap dataset, which demonstrates the generalisation capacity of our algorithm.

The numerical evaluation and comparison with state-of-the-art methods is shown in Table 2. We follow [15] and perform our experiments on the same sequences used to evaluate their method. The mean error and standard deviation are in mm. In our method, all values represent absolute errors as in [4, 9]. However, in [5, 15], the values are the relative errors. Regarding positioning our algorithm, the closest method is Simo-Serra *et al.* [15] where the two methods are initialised with noisy observations. In [4, 9], temporal consistency constraints are imposed to remove the ambiguity, requiring multiple images. In contrast, our method estimates 3D pose from a single image. Apart from [5], our method performs better than all other methods. [5] relies on a strong assumption by employing background subtraction and, thus, cannot easily deal with changing backgrounds. In contrast, we test our method on images with different and cluttered backgrounds without the need for prior background subtraction. Moreover, in [5], the training, validation and testing sequences are all from the HumanEva-I

	Walking		
	S1	S2	S3
Proposed	**75.1 (35.6)**	**99.8 (32.6)**	**93.8 (19.3)**
[15]	99.6 (42.6)	108.3 (42.3)	127.4 (24.0)
[9]	89.3	108.7	113.5
[4]	-	107 (15)	-
[5]	38.2 (21.4)	32.8 (23.1)	40.2 (23.2)
	Jog		
	S1	S2	S3
Proposed	**79.2 (26.4)**	**89.8 (34.2)**	**99.4 (35.1)**
[15]	109.2 (41.5)	93.1 (41.1)	115.8 (40.6)
[5]	42.0 (12.9)	34.7 (16.6)	46.4 (28.9)

Table 2: **Quantitative comparison** of our algorithm with state-of-the-art methods on the *walking* and *jogging* sequences from the HumanEva-I dataset. Values are in mm. Values outside the parentheses are the average mean error per joint from the ground truth. Values in parentheses show the standard deviation. [4, 9] do not provide an evaluation for *jogging*. [5] assumes prior background subtraction.

dataset. In our method, we show its good generalisation capability by training the regression models on frames from CMU Mocap and testing on sequences from HumanEva-I.

In the initialisation step, we propose a solution to the problem of overlapping and missing parts due to self-occlusion by breaking the springs between non-adjacent nodes. However, it is clear that the problem partially still exists and needs a more robust technique to reduce the noisy observations. Inspired by [15], a rigid alignment between the produced shapes and ground truth values is computed, which reduces the reconstruction error further. In our experiments, the average of the reconstruction error is around 200mm and the aligned error is 90mm on average. Note that most of the errors are due to the offset in the 2D points resulting from the output of the initialisation step.

W.r.t. the computational time, estimating the 3D pose takes around 1min for each input image including the time required to get the initial 2D view.

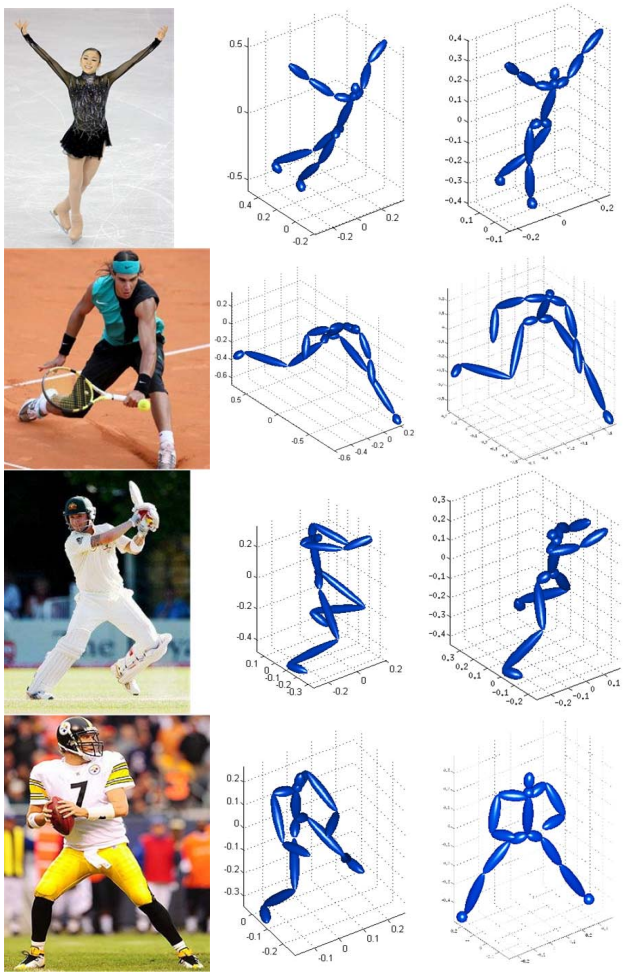
4.3. Qualitative Evaluation

To test the robustness of our algorithm for hallucinated images with a large degree of freedom and strong self-occlusion, two experiments are conducted. As the ground truth of the 3D poses for these images is not available, a qualitative visual comparison is presented.

In the first experiment (see Fig. 3), we visually compare our approach and Valmadre *et al.* [18]. For both techniques, the initialisation is performed via manually annotated 2D points. [18] uses multiple, different images to recover the 3D pose. Our approach uses only a single image. Furthermore, the method of Valmadre *et al.* fails to remove all ambiguities, in particular, the sign of the joint angles. It requires the user to specify the direction (positive or negative). In our method, the algorithm succeeds in the vast majority of cases to remove this type of ambiguity by sharing the sign of the unambiguous parts in the various synthetic views.

Fig. 3 (b) and (c) represent the 3D output for the method in [18] and our algorithm, respectively. Specifically, the motivation behind this comparison is to show the advantage of employing structure from motion *after* regressing multiple views from the initial one. Noise that results from the regression predictions is filtered out afterwards in the factorisation, which reduces the ambiguity in the final stage.

In the second experiment (see Fig. 4), we evaluate the impact of the proposed self-occlusion handling (cf. Sec. 3.1). The experiments are performed on images from the IP dataset [12]. Fig. 4a shows the results of our algorithm initialised with the output of a Mixture of Pictorial Structures [20]. Fig. 4b shows the output for the same images but with the self-occlusion handling mechanism. It is visually evident that handling self-occlusion improves the initialisation accuracy and stops the error from being propagated to the synthesised views and then to the final 3D pose.

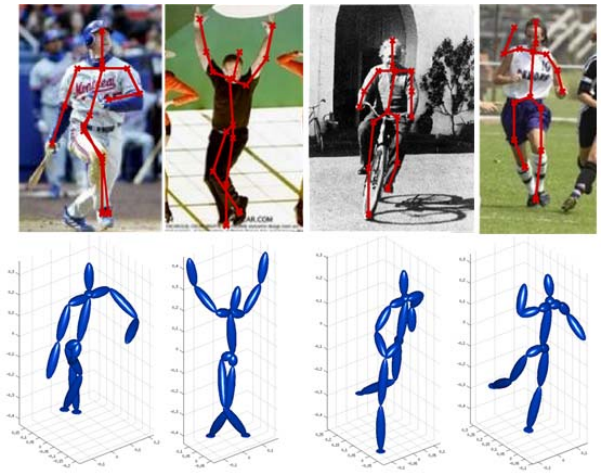


(a) (b) (c)

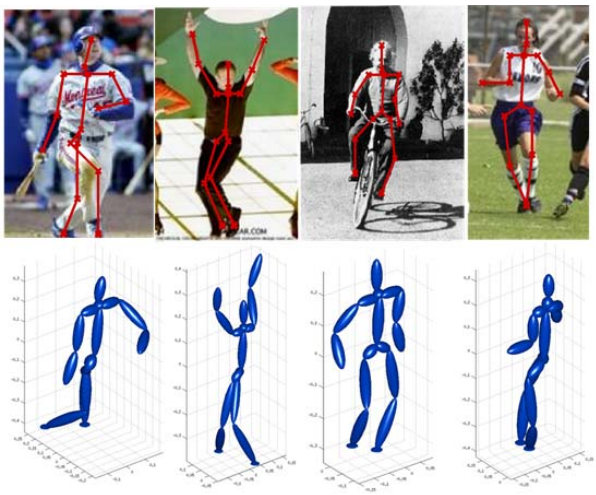
Figure 3: **Qualitative comparison:** (a) Input image. (b) Results of recovering the 3D pose for the input image by Valmadre *et al.* [18], using *multiple* images with different poses to build the 3D. (c) Results of the proposed approach, which is initialised with 2D points from a *single* image. The 3D poses are normalised and centred on the origin.

5. Conclusions

We propose a 3D pose reconstruction algorithm from a single 2D image. In the initialisation step, we utilise a well-known 2D part detectors to produce the 2D joints. We propose a novel way to improve the output of this step by handling self-occlusion. To enforce more constraints, we generate synthetic views by regressing the initial view to multiple oriented views. The ambiguity is reduced by imposing kinematic and orientation constraints on the 3D ambiguous pose resulting from the projection of a 3D model onto the initial pose. The experiments show promising results of the proposed algorithm. However, noisy observations can still affect the accuracy of the final 3D pose. Future work in-



(a)



(b)

Figure 4: Visual comparison of the final 3D pose estimate (a) without and (b) with self-occlusion handling. In (a), self-occlusion leads to erroneous initialisation, which propagates to the final 3D pose. In (b), the initialisation is accurate, leading to an accurate 3D pose estimate.[2]

cludes providing a more robust handling of self-occlusion and testing on different 'in the wild' situations.

References

[1] A. Agarwal and B. Triggs. 3D Human Pose from Silhouettes by Relevance Vector Regression. In *CVPR 2004*, pages II–882 – II–888, 2004.

[2] A. Agarwal and B. Triggs. Learning to track 3D human motion from silhouettes. In *ICML '04*. ACM, 2004.

[3] A. Agarwal and B. Triggs. Recovering 3D Human Pose from Monocular Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1):44–58, 2006.

[4] M. Andriluka, S. Roth, and B. Schiele. Monocular 3D Pose Estimation and Tracking by Detection. In *CVPR 2010*, pages 623–630, 2010.

[5] L. Bo and C. Sminchisescu. Twin Gaussian Processes for Structured Prediction. *IJCV*, 87(1–2):28–52, 2010.

[6] L. Bo, C. Sminchisescu, A. Kanaujia, and D. N. Metaxas. Fast Algorithms for Large Scale Conditional 3D Prediction. In *CVPR 2008*, 2008.

[7] C. A. Bouman. CLUSTER: an unsupervised algorithm for modeling Gaussian mixtures, 2005. `http://cobweb.ecn.purdue.edu/~bouman/software/cluster/`.

[8] C. Bregler, A. Hertzmann, and H. Biermann. Recovering Non-Rigid 3D Shape from Image Streams. In *CVPR 2000*, pages 690–696, 2000.

[9] B. Daubney and X. Xie. Tracking 3D Human Pose with Large Root Node Uncertainty. In *CVPR 2011*, pages 1321–1328, 2011.

[10] P. Dollár, P. Welinder, and P. Perona. Cascaded pose regression. In *CVPR 2010*, pages 1078–1085, 2010.

[11] I. Radwan, A. Dhall, J. Joshi, and R. Goecke. Regression Based Pose Estimation with Automatic Occlusion Detection and Rectification. In *ICME 2012*, pages 121–127, 2012.

[12] D. Ramanan. Learning to Parse Images of Articulated Bodies. In *NIPS*, 2006.

[13] 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(1–2):4–27, 2010.

[14] L. Sigal and M. J. Black. Measure Locally, Reason Globally: Occlusion-sensitive Articulated Pose Estimation, booktitle = CVPR 2006, pages = 2041–2048, year = 2006.

[15] E. Simo-Serra, A. Ramisa, G. Alenyà, C. Torras, and F. Moreno-Noguer. Single Image 3D Human Pose Estimation from Noisy Observations. In *CVPR 2012*, pages 2673–2680, 2012.

[16] C. J. Taylor. Reconstruction of Articulated Objects from Point Correspondences in a Single Uncalibrated Image. In *CVPR 2000*, pages 677–684, 2000.

[17] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *IJCV*, 9(2):137–154, 1992.

[18] J. Valmadre and S. Lucey. Deterministic 3D Human Pose Estimation Using Rigid Structure. In *ECCV 2010*, pages 467–480, 2010.

[19] X. K. Wei and J. Chai. Modeling 3D Human Poses from Uncalibrated Monocular Images. In *ICCV 2009*, pages 1873–1880, 2009.

[20] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR 2011*, pages 1385–1392, 2011.

[21] Y. Yang and D. Ramanan. Articulated Human Detection with Flexible Mixtures-of-Parts. *IEEE Transactions on PAMI*, PP(99), 2012.

[2] More qualitative results can be found at `http://staff.estem-uc.edu.au/ibrahim/3dmodel`.