Unconstrained Monocular 3D Human Pose Estimation by Action Detection and Cross-modality Regression Forest

Tsz-Ho Yu
University of Cambridge
Cambridge, UK
thy23@cam.ac.uk

Tae-Kyun Kim
Imperial College London
London, UK
tk.kim@imperial.ac.uk

Roberto Cipolla
University of Cambridge
Cambridge, UK
cipolla@eng.cam.ac.uk

Abstract

This work addresses the challenging problem of unconstrained 3D human pose estimation (HPE) *from a novel perspective. Existing approaches struggle to operate in realistic applications, mainly due to their scene-dependent priors, such as background segmentation and multi-camera network, which restrict their use in unconstrained environments. We therfore present a framework which applies* action detection *and* 2D pose estimation *techniques to infer 3D poses in an unconstrained video. Action detection offers spatiotemporal priors to 3D human pose estimation by both recognising and localising actions in space-time. Instead of holistic features,* e.g. *silhouettes, we leverage the flexibility of deformable part model to detect 2D body parts as a feature to estimate 3D poses. A new unconstrained pose dataset has been collected to justify the feasibility of our method, which demonstrated promising results, significantly outperforming the relevant state-of-the-arts.*

1. Introduction

3D human pose estimation (HPE) has been a longstanding challenge in computer vision. 3D HPE aims to infer a human pose, represented by joint positions or angles, from input images or videos. Contemporary methods commonly approach 3D pose estimation as a regression or a manifold learning scenario, features are embedded to a parametrised 3D pose space. However, learning mappings between high-dimensional spaces is an essentially ill-posed problem [10]. Additional priors are needed to optimise a correct pose from multiple hypotheses. These priors are crucial to pose estimation, however, they also require a much controlled environment to capture compatible data: clean background segmentation, a calibrated multi-camera network, and depth sensor. Furthermore, if there are changes to the imaging environment, the whole pose estimator has to be retrained, making the pose estimation algorithm not scalable.

In this paper, we present a new method that incorporates action detection and 2D part-based pose estimation techniques for realistic, video-based 3D pose estimation. Our contributions are three-folds:

Action detection. Firstly, we combine action detection with 3D pose estimation to utilise the strong spatiotemporal structures of actions. The analysis of human pose and action are two closely interrelated areas in computer vision. Although there exist initial studies in using human poses for recognising actions *e.g.* [34, 31], the opposite direction, *i.e.* using actions to help pose analysis, is still an aspect that many methods have overlooked. An atomic action is considered as a time series of poses, with particular starting/ending poses and their transitions in-between. By detecting an atomic action in video, a strong 3D pose prior per each frame is obtained. In addition to kinematic constraints, action determines the temporal structure of a series of poses. For instance, in figure 1, action detection simultaneously estimates the action category and the space-time location of an action, supporting pose estimation within the action's time span.

2D part detection. Secondly, we apply 2D part-based pose estimation techniques to infer 3D articulated poses. Holistic shape features, such as silhouettes which rely on background segmentation, are replaced by a deformable part model (DPM), *e.g.* [33], to maximise flexibility. Our method is therefore knowledge transferable, learned models can be reused in unseen environments without retraining (see figure 2).

Cross-modality regression forest. Finally, we refine 3D human poses from 2D part detections using cross-modality regression forests. To the best of our knowledge, this is the first application of regression forest [8] across modalities to 3D HPE problem. Estimating 3D human pose is essentially a cross-modality regression problem: *3D structure* of a human pose is inferred using features extracted from its *2D appearance* Since the relationship between the two spaces are implicit, learning a robust regression model

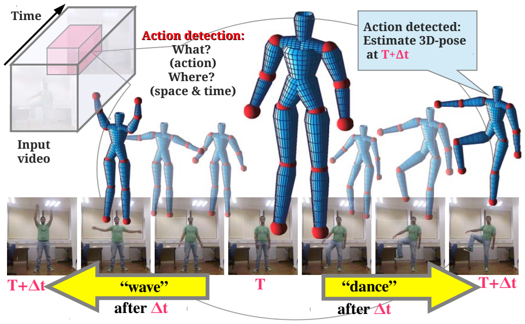


Figure 1: Action detection helps 3D pose estimation by providing the spatiotemporal structure of actions.

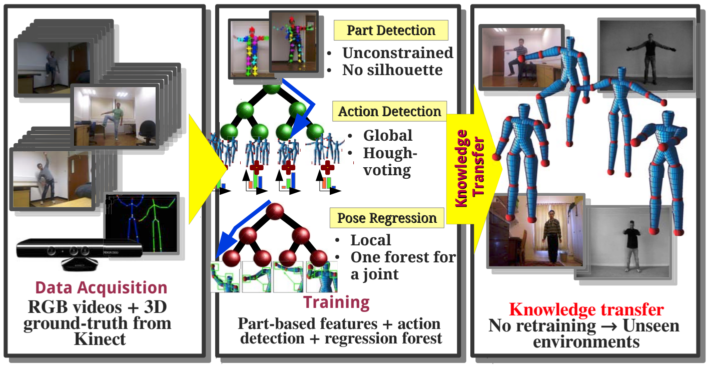


Figure 2: Knowledge transfer capability of the proposed method

across two modalities becomes a challenging issue. The regression forests estimate joint positions in 3D from detected 2D parts, which are combined with action detection to optimise 3D pose estimation. The outputs of both forests, and their combined results, are formulated in a probabilistic framework. While some methods yield a point estimate in the pose space, our approach outputs joint positions as probability distributions in 3D space.

2. Literature Review

Early methods. Part-based 2D pose analysis has been studied for decades, examples of early approaches include *e.g.* pictorial structure [13] and template matching [16]. However, these methods are limited due to the lack of an automatic part detector, which implies manual labelling required for both training and testing data. 3D human pose estimation is more complicated than its 2D counterpart due to occlusions and high dimensionality. To estimate 3D pose from video data, various techniques have been proposed, *e.g.* image edges [15] and silhouettes [19]. Approaches for traditional 3D human pose estimation are discussed in [21]. Cross-modality approach had not been explored until [18], where 2D face and hand detectors were used to infer a simple 3D pose of upper body.

More recent techniques of human pose estimation are discussed below according to the techniques or representations used.

Silhouette or depth image. Holistic shapes, silhouettes in particular, are common features for 3D pose estimation. Current approaches achieve state-of-the-art performance by combining silhouettes with new features or constraints, including motion templates [24], pedestrian detectors [5], shape-contents [3] and user interaction [14]. Thanks to the introduction of the Kinect sensor, using depth images emerges as a new direction for 3D HPE, as it provides inherent depth information and object segmentation. Several methods recognise 3D human poses from depth images, using techniques such as point cloud matching [6, 35] and random forest [30, 17]. However, many of the above methods require an accurate image segmentation to extract shape features, thus special hardware (*e.g.* Kinect sensor) or controlled environments are often necessary to acquire compatible training or testing data.

Multiple-cameras. A common approach to resolve the pose ambiguity is to maximise the field of view by capturing multiple images simultaneously using calibrated cameras, *e.g.* [20, 27, 34]. Although they achieve excellent accuracy, potential applications are restricted to a fixed, calibrated multi-camera system.

Action for pose estimation. As mentioned previously, the integration of action and pose, in particular action for pose, is still a largely unexplored area. We seek to investigate the feasibility of using *action detection* to facilitate 3D human pose estimation in uncontrolled and monocular videos. Early approaches that combines action and pose constrains include [36] and [22]. The closest work to this idea is [34] that uses action recognition to assist a multi-view 3D HPE algorithm. Separate regression models are trained; After an action is recognised, poses are inferred using the model of the action class estimated. Whilst action recognition is applied, the inputs are still images captured from a controlled, multi-camera environment. Instead, we perform action detection in video, by which we exploit the spatiotemporal structure of actions in addition to action class labels, to infer 3D poses.

Part-based pose estimation. Various methods have been introduced built upon the original seminal model of pictorial structures [12, 11]. Recent extensions include new motion features [4], improved part-based model learning and inference [33, 29], and pre-processing techniques (*e.g.* face detector) [9].

Recent advances in 2D human pose estimation methods, particularly in uncontrolled environments, have inspired a resurgence of part-based approaches for 3D pose estimation. While some approaches use manually labelled 2D parts to estimate 3D poses, *e.g.* [32, 23], 2D deformable part

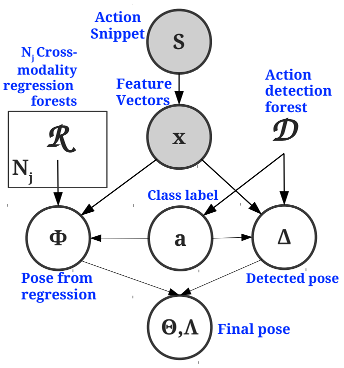


Figure 3: Graphical representation of the proposed model

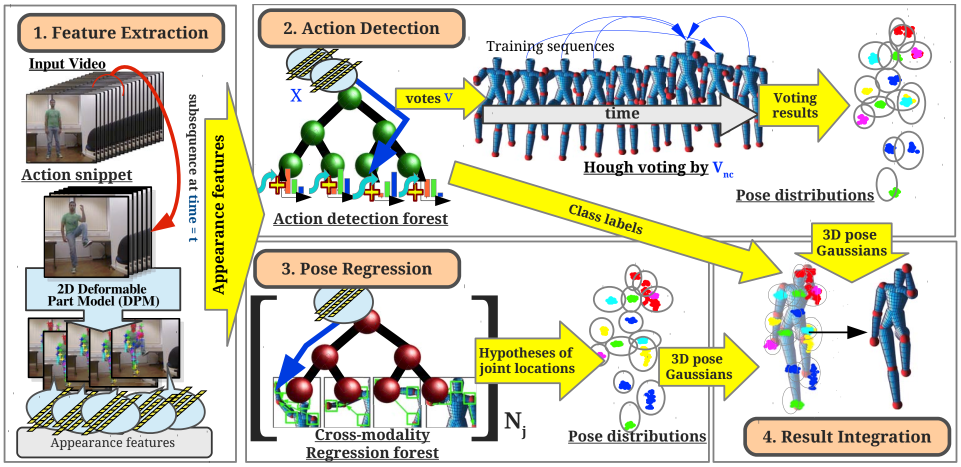


Figure 4: Overview of the proposed framework

models are also investigated in 3D human pose estimation, for instance, [5] uses a pedestrian detector with deformable body parts to estimate rough 3D poses in street scenes, [28] optimises multiple pose hypotheses from 2D DPM using inverse kinematics to estimate 3D pose in a single image. In this work, we further investigate the use of 2D DPM in a multi-action 3D HPE scenario.

3. Method

Figure 3 describes the graphical model of our 3D human pose estimation framework: action detection is performed to yield the rough 3D pose estimates, then cross-modality regression forests with the estimated action classes are applied to refine the 3D pose estimations. While full poses, *i.e.* 3D coordinates of all $\mathrm{N_J}$ joints, are learnt in the action detection forest, one joint location is estimated per cross-modality regression forest. $\mathrm{N_J}$ regression forests are hence trained separately in the model. The conceptual data-flow of the proposed framework is illustrated in figure 4.

3.1. Part-based Feature Extraction

In order to perform 3D pose estimation in different backgrounds and scenarios, input features are extracted using a 2D part-based model. Firstly, a deformable part model (DPM) [33] is employed as an off-the-shelf method to extract body parts from input frames. Unlike the traditional holistic silhouette-based methods, where background subtraction is preformed, our method based on body parts works at unseen and dynamic backgrounds. The DPM fires $\mathrm{N_h}$ body part configurations per frame, each hypothesis contains 26 locations of detected body parts as in [33]. Subsequently, the detected configurations are normalised with respect to the distance between the head and the waist part. Finally, a feature vector is computed per configuration by the pairwise distances among the normalised parts, hence the feature vector takes $325 (= 26 \times 25/2)$ dimensions.

In the training process, every feature vector is associated with a ground truth 3D pose. The Kinect sensor is used to acquire 3D poses simultaneously with the RGB video sequences. A 3D human pose is represented by the scale-normalised coordinates of the $\mathrm{N_J}$ joints detected by the Kinect sensor, an articulated model of $\mathrm{N_J} = 15$ joints is used in this work. Every feature vector in the training dataset is assigned to the 3D pose detected and one of the C action categories, according to its corresponding frame and video. The feature vectors are denoted as $\mathcal{X} = \{\mathbf{x}_{pqr}\}$, where $\mathbf{x}_{pqr}$ is the r-th configuration detected in the q-th frame of the p-th training video. The corresponding class label and corresponding 3D pose are defined as $\mathcal{A} = \{a_{pqr} | a_{pqr} \in 1, \ldots, C\}$ and $\mathcal{Y} = \{\mathbf{y}_{pqr} | \mathbf{y}_{pqr} \in \mathbb{R}^{3\mathrm{N_J}}\}$ respectively. Furthermore, 3D poses $\mathcal{Y}$ of each action category are compressed separately into low-dimensional vectors $\mathcal{U} = \{\mathbf{u}_{pqr} | \mathbf{u}_{pqr} \in \mathbb{R}^6\}$ using PCA.

3.2. Learning

Action Detection Forest. The action detection forest, $\mathcal{D}$, performs action categorisation and 3D pose clustering simultaneously. In each leaf node, the 3D pose vectors $\mathcal{U}$ are classified into the action class labels $\mathcal{A}$ and cohering 3D pose vectors are grouped together. For given data triplets $\mathcal{X}$, $\mathcal{U}$ and $\mathcal{A}$, we grow $\mathrm{N}_{\mathcal{D}}$ decision trees by recursively splitting the data into two child nodes, where candidate split functions are generated randomly, each compares a value in the feature vector $\mathbf{x}$ with a threshold. The best split is chosen among the candidates by maximising a quality measure. The splitting process is performed until the new node reaches its maximum depth or minimum number of data points. Lastly, every leave node stores the votes that are required in Hough-voting for action detection during testing.

We propose a new integrated quality measure $\mathrm{H}(\cdot)$ at the node n as:

$$\mathrm{H}(\mathcal{U}_n) = (1-\omega)\mathrm{H}_\mathrm{a}(\mathcal{U}_n) + \omega\mathrm{H}_\mathrm{p}(\mathcal{U}_n). \tag{1}$$

The first term, $\mathrm{H}_\mathrm{a}(\cdot)$, is the information gain measure used in standard classification forest [7] (here for action classification); the second term measures the improvement in 3D pose coherence when a split is performed

$$\mathrm{H}_\mathrm{p}(\mathcal{U}_n) = \sum_{c=1}^{C} \Psi(\Sigma(\mathcal{U}_{nc})) - \Psi(\Sigma(\mathcal{U}_{lc})) - \Psi(\Sigma(\mathcal{U}_{rc})), \tag{2}$$

where $\Psi(\cdot) = \log(\det(\cdot))$ and $\Sigma(\mathcal{U}_{nc})$ is the covariance matrix of the PCA-compressed 3D pose vectors $\mathcal{U}_{nc}$ of the n-th node (l, r denote the left and right split respectively) and the c-th action class. These measures are weighted by ω that describes the class purity of a node as

$$\omega = \max_c(|\mathcal{A}_{nc}|/|\mathcal{A}_n|) - \min_c(|\mathcal{A}_{nc}|/|\mathcal{A}_n|), \tag{3}$$

where $\mathcal{A}_n$ denotes the action labels of training data in node n and $\mathcal{A}_{nc}$ the action labels of node n and class c. The first quality term in (1) optimises action classification performance while the second term optimises pose clustering performance within a node. Initially, classification is preferred over clustering, $\mathrm{H}_\mathrm{a}(\cdot)$ is dominant when class labels are evenly distributed. However, ω gradually increases as the tree grows, shifting the learning focus to clustering.

Once the learning is completed, the class posterior of a leaf node $\hat{n}$ is obtained by:

$$P(a = c|\hat{n}) = |\mathcal{A}_{\hat{n}c}|/|\mathcal{A}_{\hat{n}}| \tag{4}$$

The distribution of 3D poses, given a action class-label c, is modelled by a Gaussian $\mathcal{N}(\mu(\mathcal{U}_{\hat{n}c}), \Sigma(\mathcal{U}_{\hat{n}c}))$. A vote $v_{\hat{n}c}$ cast by the leaf node is a pair $v_{\hat{n}c} = (p_{\hat{n}c}, q_{\hat{n}c}), c = 1, ..., C$ such that

$$(p_{\hat{n}c}, q_{\hat{n}c}) = \arg\min_{(p,q)} ||\mathbf{u}_{pqr} - \mu(\mathcal{U}_{\hat{n}c})||_2 \tag{5}$$

$p_{\hat{n}c}$ and $q_{\hat{n}c}$ are the indices of the training sample that is the nearest neighbour of Gaussian mean $\mu(\mathcal{U}_{\hat{n}c})$. As a result, q is a temporal vote to the staring point of actions in sequence p.

Cross-modality Regression Forest. The cross-modality regression forests $\{\mathcal{R}^{(j)}|j = 1, \ldots, \mathrm{N_J}\}$, inspired by [8], are learned to refine the 3D locations of joints. Each forest $\mathcal{R}^{(j)}$ contains $\mathrm{N}_\mathcal{R}$ trees that estimate the location of the j-th joint, trained independely from the dataset $\{\mathcal{Y}^{(j)}, \mathcal{X}, \mathcal{A}\}$, where $\mathcal{Y}^{(j)}$ is the j-th joint's 3D coordinates in $\mathcal{Y}$. Split function candidates are generated in the same way as action detection using the feature vector $\mathbf{x}$. Although action class posteriors can be computed in the terminal nodes, it is not modelled in this method as the action detection forest $\mathcal{D}$ provides a better action recognition rate that helps the localisation accuracies of $\mathcal{R}^{(j)}$. Thus, $\mathrm{H}_\mathrm{p}(\mathcal{Y}^{(j)}_n)$ is used instead to optimise the split functions for the localisation accuracy of the j-th joint.

Tree growing is stopped when the current node is smaller than a certain size. Upon completion of $\mathcal{R}^{(j)}$, the output of its leaf nodes $\hat{n}$ are described by the mean joint coordinates with respect to class label, given by $\mu(\mathcal{Y}^{(j)}_{\hat{n}c})$.

3.3. Testing

Video snippet, the basic unit required for action detection [25], is a short sequence excerpted from the testing video, centered at time t. A snippet $\mathcal{S}_t$ contains l frames such that $\mathcal{S}_t = \{I_{t-l/2}, \ldots, I_{t+l/2-1}\}$. The testing process starts with extracting features from $\mathcal{S}_t$, such that $\mathcal{X}_{\mathcal{S}_t} = \{\mathbf{x}_{ij}\}$, where $i = t - l/2, ..., t + l/2 - 1$ and $j = 1, ..., \mathrm{N_h}$ with $\mathrm{N_h}$ denotes the number of part configurations of the frame.

Action detection (classification). Action detection forest $\mathcal{D}$ performs action classification on $\mathcal{S}_t$. Let $\hat{n}_k[\mathbf{x}_{ij}]$ be the leaf node reached by a feature $\mathbf{x}_{ij}$ in the k-th tree of $\mathcal{D}$. The posterior of snippet action class at time t is defined as

$$P(a = c|\mathcal{S}_t, \mathcal{D}) = \sum_{k,i,j=1}^{\mathrm{N}_\mathcal{D}, l, \mathrm{N_h}} \frac{P(a = c|\hat{n}_k[\mathbf{x}_{ij}])}{\mathrm{N}_\mathcal{D} l \mathrm{N_h}}. \tag{6}$$

Action detection (pose voting). A Hough-based voting scheme is designed for action detection. As mentioned in section 3.2, the vote (p, q) stored in $\mathcal{D}$ are temporally associated with their corresponding action sequence and time frame in the training data set. Note each training frame is paired with a ground truth 3D pose. Hence, all frames I can vote for a 3D pose at time t, by applying temporal offsets δ to the votes obtained from $\mathcal{X}_{\mathcal{S}_t}$. We define a function $\Delta(\mathcal{S}_t, c)$ that returns a set of 3D pose estimates in $\mathcal{Y}$ for action label c from Hough-voting during action detection:

$$\Delta(\mathcal{S}_t, c) = \mathcal{Y}^{\Delta}_{\hat{n}_k[\mathbf{x}_{(t+\delta)j}]c} \tag{7}$$

where $k = 1, ..., \mathrm{N}_\mathcal{D}$, $\delta = -l/2, ..., l/2 - 1$, $j = 1, ..., \mathrm{N_h}$. The set $\mathcal{Y}^{\Delta}_{\hat{n}_k[\mathbf{x}_{(t+\delta)j}]c}$ denotes the δ-voted (offset-ed) pose from the δ-th frame in $\mathcal{S}_t$, *i.e.* $(t+\delta)$-th frame in input video, by passing down the k-th tree in $\mathcal{D}$

$$\begin{gathered} \mathcal{Y}^{\Delta}_{\hat{n}_k[\mathbf{x}_{(t+\delta)j}]c} = \{\mathbf{y}_{p(q-\delta)r}\} \\ \text{s.t. } (p, q) = v_{\hat{n}_k[\mathbf{x}_{(t+\delta)j}]c} \text{ and } a_{pqr} = c \end{gathered} \tag{8}$$

Elements returned from $\Delta(\mathcal{S}_t, c)$ represent 3D pose estimations at time t. A 3D pose α_t is hence modelled by $\mathrm{N_J}$ independent Gaussians with respect to its joints.

$$\begin{aligned} &P(\alpha^{(j)}_t|\mathcal{S}_t, a = c, \mathcal{D}) \\ &= \mathcal{N}(\alpha^{(j)}_t; \mu(\Delta^{(j)}(\mathcal{S}_t, c)), \Sigma(\Delta^{(j)}(\mathcal{S}_t, c)) \end{aligned} \tag{9}$$

where $\alpha^{(j)}_t \in \mathbb{R}^3$ is the j-th joint in $\alpha_t \in \mathbb{R}^{3\mathrm{N_J}}$.

Cross-modality Regression. Estimation of current 3D pose by the regression forest, β_t, is performed on per-frame basis. Passing down features of current tth frame $\{\mathbf{x}_{ti}|i = 1,...,\mathrm{N_h}\}$, the set of pose estimates for class c is returned by the function $\Phi(\mathcal{S}_t, c)$

$$\Phi^{(j)}(\mathcal{S}_t, c) = \mathcal{Y}^{(j)}_{\hat{n}_k[\mathbf{x}_{ti}]c}, i=1,...,\mathrm{N_h}, k=1,...,\mathrm{N}_{\mathcal{R}} \quad (10)$$

The distribution of votes for the j-th joint is described by an Gaussian:

$$\begin{aligned} &P(\beta_t^{(j)}|\mathcal{S}_t, a = c, \mathcal{R}) \\ &= \mathcal{N}(\beta_t^{(j)}; \mu(\Phi^{(j)}(\mathcal{S}_t, c)), \Sigma(\Phi^{(j)}(\mathcal{S}_t, c)) \end{aligned} \quad (11)$$

Combined Pose Estimation. Three-dimensional human poses are estimated globally via action detection, and locally by the joint regression forests. Assuming β_t and α_t are independent, the probability of both observations coincide at γ is

$$\begin{aligned} &P(\alpha_t = \beta_t = \gamma, a = c|\mathcal{S}_t, \mathcal{R}, \mathcal{D}) \\ &= P(a = c|\mathcal{S}_t, \mathcal{D}) \prod_{j=1}^{\mathrm{N_J}} P(\gamma^{(j)}|\mathcal{S}_t, a = c, \mathcal{R}) \\ &\qquad P(\gamma^{(j)}|\mathcal{S}_t, a = c, \mathcal{D}) \\ &= P(a = c|\mathcal{S}_t, \mathcal{D}) \prod_{j=1}^{\mathrm{N_J}} \mathcal{N}(\gamma^{(j)}; \Theta^{(j)}_{tc}, \Lambda^{(j)}_{tc}) \end{aligned} \quad (12)$$

Since the product of Gaussians is also a Gaussian, such that $\Theta^{(j)}_{tc}$ and $\Lambda^{(j)}_{tc}$ are

$$\begin{aligned} \Theta^{(j)}_{tc} &= \Lambda^{(j)}_{tc}[\Sigma(\Phi^{(j)}(\mathcal{S}_t, c))^{-1}\mu(\Delta^{(j)}(\mathcal{S}_t, c)) + \\ &\quad \Sigma(\Delta^{(j)}(\mathcal{S}_t, c))^{-1}\mu(\Phi^{(j)}(\mathcal{S}_t, c))] \\ \Lambda^{(j)}_{tc} &= [\Sigma(\Phi^{(j)}(\mathcal{S}_t, c))^{-1} + \Sigma(\Delta^{(j)}(\mathcal{S}_t, c))^{-1}]^{-1} \end{aligned} \quad (13)$$

Consider the probability distribution in equation (12), the final 3D pose estimation is described by the mean joint location $\Theta^{(j)}_{t\hat{c}}$ of the most probable action category $\hat{c} = \arg\max_c P(a = c|\mathcal{S}_t, \mathcal{D})$, with the confidence region indicated by the covariance $\Lambda^{(j)}_{t\hat{c}}$, where $j = 1, ..., \mathrm{N_J}$.

4. Evaluation

4.1. The APE Evaluation Dataset

Experiments were performed to investigate the feasibility of the proposed approach. Existing public 3D pose datasets are inadequate to justify the main objectives of the proposed approach. Whilst benchmarks such as [26, 34] model joint angles rather than joint positions using sophisticated techniques, e.g. camera networks, from a static area, our framework focuses on flexible, multi-action 3D HPE from monocular videos without using background statistics.

To this end, we collected the *action-pose-estimation (APE)* dataset for both quantitative and qualitative evaluations. The APE dataset contains 245 sequences from 7 subjects performing 7 categories of actions. Videos of each subject were recorded in different environments, changing camera poses and moving background objects. The APE dataset will be made publicly available.

The setting of APE dataset is considered challenging for traditional 3D HPE because: (1) no scene-dependent cues, *e.g.* foreground segmentation, can be applied, (2) testing is done in unseen environments. Experiments are divided into two parts. In the first part, pose estimation accuracy was evaluated quantitatively with ground truth and current state-of-the-arts, in 3D and 2D respectively. In the second part, we demonstrated the knowledge transfer ability qualitatively, by testing with other videos and datasets.

4.2. Experimental Results

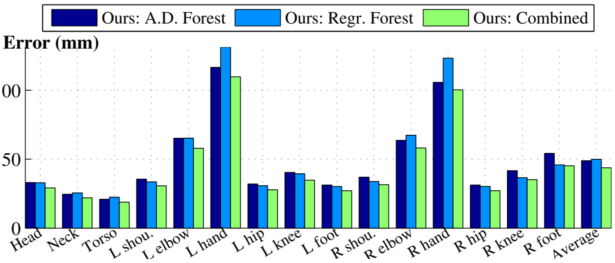
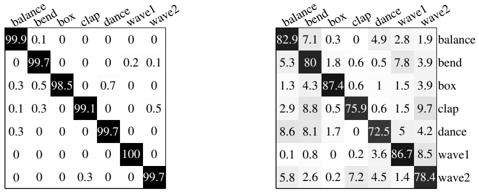


Figure 6: 3D joint localisation errors based on ground truth pose from Kinect sensor

Quantitative Evaluation. The proposed approach was evaluated quantitatively using the leave-one-out cross-validation strategy. A subject was taken out in turn for testing, thus the model is trained with 210 sequences and the remaining 35 sequences are evaluated. Snippets are extracted densely from training and testing data, where $l = 10$. Table 1 lists the training parameters.

Pose estimation accuracy was evaluated in both 2D and 3D. Accuracies of 3D joint coordinates were compared directly with ground truth 3D poses captured by the Kinect sensor. Accuracies in 2D were measured by back-projecting



balance	bend	box	clap	dance	wave1	wave2
99.9	0.1	0	0	0	0	0
0	99.7	0	0	0	0.2	0.1
0.3	0.5	98.5	0	0.7	0	0
0.1	0.3	0	99.1	0	0	0.5
0.3	0	0	0	99.7	0	0
0	0	0	0	0	100	0
0	0	0	0.3	0	0	99.7

balance	bend	box	clap	dance	wave1	wave2	
82.9	7.1	0.3	0	4.9	2.8	1.9	balance
5.3	80	1.8	0.6	0.5	7.8	3.9	bend
1.3	4.3	87.4	0.6	1	1.5	3.9	box
2.9	8.8	0.5	75.9	0.6	1.5	9.7	clap
8.6	8.1	1.7	0	72.5	5	4.2	dance
0.1	0.8	0	0.2	3.6	86.7	8.5	wave1
5.8	2.6	0.2	7.2	4.5	1.4	78.4	wave2

Figure 7: Confusion matrices of action classification by (left) action detection forest, and (right) cross-modality regression forest

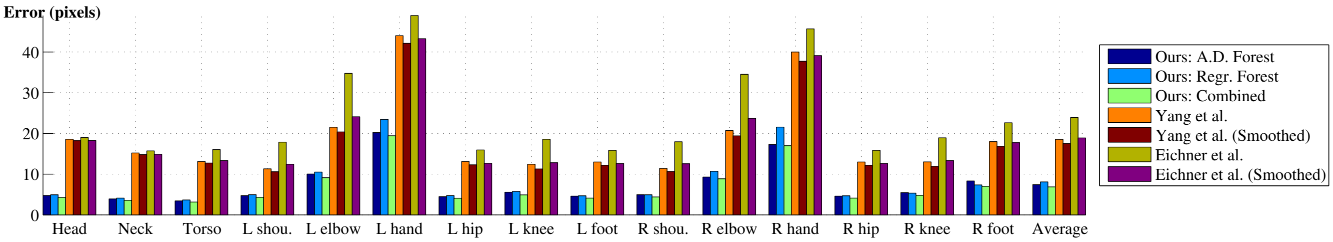


Figure 5: 2D joint localisation errors of our methods, [33] and [9] their median smoothed versions.

Forest	# tree	Max. depth	Min. node size
$\mathcal{D}$	10	16	15
$\mathcal{R}$	10	14	20

Table 1: Parameters used in training $\mathcal{D}$ and $\mathcal{R}$.

Method / Action / Error (mm)	Balance	Bend	Box	Clap	Dance	Wave 1	Wave 2
Ours:A. D. Forest	41.9	**58.6**	84.3	49.1	47.4	36.7	36.2
Ours:Regr. Forest	41.4	69.7	**60.3**	52.2	58.2	36.1	39.1
Ours:Combined	**37.8**	58.8	66.2	**41.0**	**45.1**	**30.1**	**34.6**

Table 2: Per-class joint localisation accuracy (3D)

Method / Action / Error (pixel)	Balance	Bend	Box	Clap	Dance	Wave 1	Wave 2
Ours:A. D. Forest	6.1	**10.2**	13.1	6.6	7.3	6.7	5.1
Ours:Regr. Forest	6.6	13.1	**9.7**	6.4	8.9	6.4	6.4
Ours:Combined	**5.6**	10.7	10.4	**5.4**	**7.0**	**4.8**	**5.2**
Eichner et al.[9]	20.6	28.6	26.4	22.5	22.6	22.3	24.8
Yang et al. [33]	14.2	23.7	21.6	17.1	16.7	16.5	19.3

Table 3: Per-class joint localisation accuracy (2D)

the poses to image coordinates. Besides the combined pose estimation Θ, we also evaluated each of the forests alone, and compared it with the latest 2D HPE algorithms [9] and [33]. In order to cope with actions performed in different speeds, testing videos are preprocessed by normalising with respect to their action speeds estimated from the first 25 frames of the videos. In order to make a fair comparison, the joint coordinates from the frame-based algorithms [9] and [33] are temporally smoothed by a 10-frame median filter, as our approach estimates poses from multiple-frame snippets.

Action classification rates of individual frames by both forests are presented in figure 7. The action detection forest achieves excellent accuracy, as it has been optimised for classification during learning, the video-based input, snippet, also provides temporal cues that improve classification. It complements the regression forests that focus on the localisation accuracy of joints. The average 3D joint localisation errors of the experiments were reported in figure 6. Sample results of the proposed method are also presented in figure 8[1].

The comparison our method and other 2D HPE algorithms is illustrated in figure 5. The proposed framework showed promising results, by extending the flexibility of [33], the proposed method showed high robustness in 3D pose estimation and outperformed both state-of-the-arts in the 2D tests. The hand parts have the highest localisation errors because of their large movements and frequent occlusions, which are also indicated by the big variance ellipsoids in figure 8.

The per-class localisation errors are listed in table 2 (3D) and 3 (2D). While some classes reported significant improvements after combining the results of action detection and pose regression, *e.g.* "clap" and "wave 1", the "bend" and "box" class reported the highest error rates. The 2D part detections obtained from the "box" and "bend" classes are less accurate than those from other classes. For the "box" action, self-occlusion happens frequently such that the part detector is confused about the left and right hand positions, making the hand position distributions spread around the torso as in figure 8(p). Similarly, when the arms are stretched overhead and occluded, the 2D DPM model used in the experiments gives incorrect results.

Qualitative Evaluation. Knowledge transfer was evaluated by reusing the models trained in section 4.2 to other datasets without retraining. The KTH [1] and Weizmann [2] dataset were used in the experiments as they shared action categories with the APE dataset.

The experimental results are reported qualitatively in figure 9. Even though the input videos are of extremely low resolutions, rendering them inapplicable to typical 3D HPE methods, our framework is still able to estimate their actions and poses simultaneously with encouraging accuracy. Incorrect poses were estimated when too many false positive parts are detected from the low resolution images, *e.g.* figure 9(g–h).

Discussion. The experimental results have demonstrated high feasibility in the idea of using action detection to estimate 3D poses under challenging conditions. Coupling the outputs from the random forests, the 3D pose estimation accuracy is further enhanced. Action detection gives

[1] Please refer to the supplementary materials for more results.

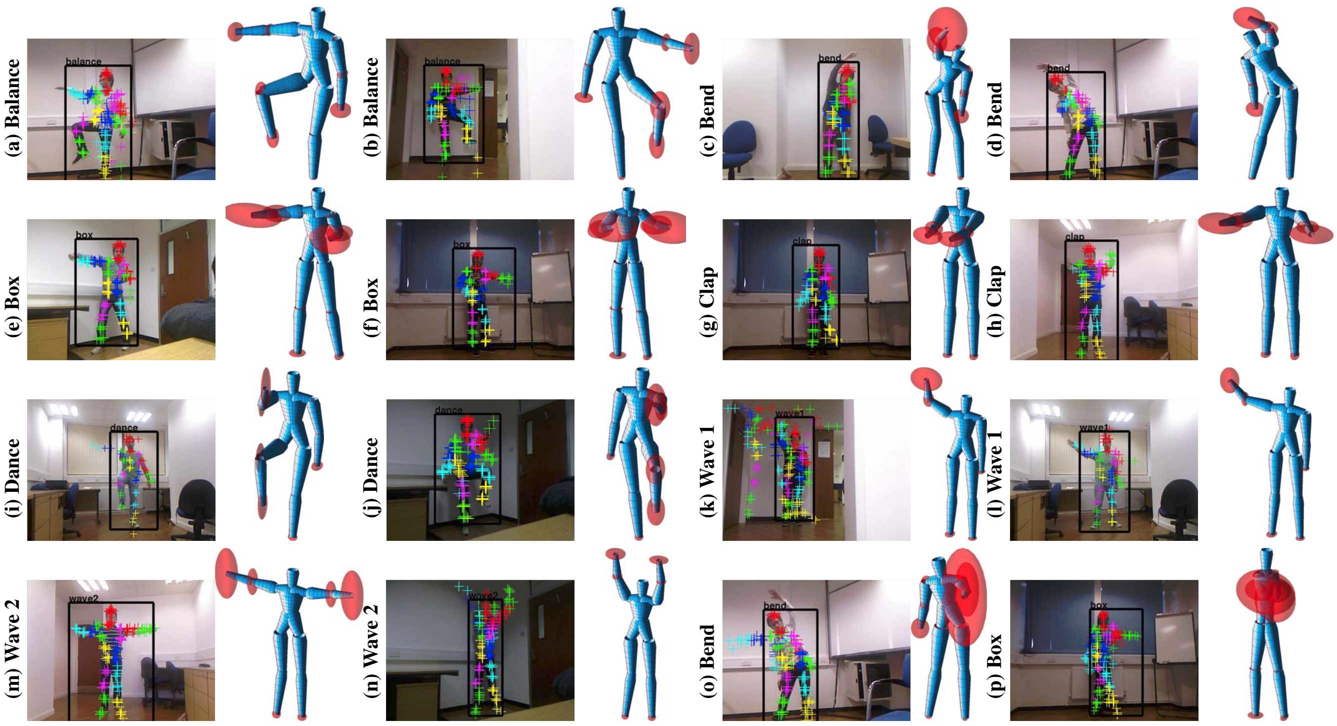


Figure 8: 3D pose estimation results with different action classes from the APE dataset: (left) Detected 2D parts and bounding box from action detection (right) the 3D pose estimated. Red ellipsoids represent the confidence region of pose estimation, Λ, in equation (13). Sample (o) and (p) shows the wrong pose estimations when the 2D body part detector fails.

a global pose estimation and the corresponding class label. Meanwhile, errors in the initial global estimation, due to the differences among individual action patterns, are corrected locally by regression forests, which improves the accuracy of the final pose estimation as shown in figure 6.

On the other side, there is still room for improvement in the proposed approach. The proposed method relies on DPM as the only source of input. Albeit great flexibility, *c.f.* [33], the performance of a DPM depends on its training data. Our method handles minor errors gracefully by allowing multiple hypotheses and snippet-based input, but large errors cannot be recovered completely, *e.g.* figure 8(o) and 8(p) for APE dataset, and 9(g) and 9(h) for KTH and Weizmann dataset respectively. Besides, the proposed method runs at 0.31fps. Feature extraction from DPM is the runtime bottleneck. The pose estimatior alone runs at about 5fps by precomputing the DPM features.

5. Conclusions

The challenging problem of 3D human pose estimation is discussed in this paper. While traditional methods for 3D human pose estimation emphasise accuracy over their compatibility with realistic applications, we present a novel practical approach without using any scene-dependent constraints. We investigate the new area of using action for pose estimation. The proposed method combines human action detection and deformable part model-based 2D human pose estimation to estimate 3D poses from unconstrained, monocular videos. The new APE dataset is introduced to evaluate the feasibility of our approach. Experimental results have shown promising results and also high flexibility by transferring the knowledge obtained from training data to other unseen datasets. In the future we plan to apply kinematic constraints in our system for pose refinement. We suggest that the collaboration between the techniques in human action and pose analysis will be beneficial to both areas of computer vision research in the coming future.

Acknowledgements T-K. Kim was partially supported by EPSRC grant (EP/J012106/1) 3D intrinsic shape recognition. APE dataset was collected with the help of the Imperial Computer Vision and Learning Lab.

References

[1] http://www.nada.kth.se/cvap/actions/. 6

[2] http://www.wisdom.weizmann.ac.il/~vision/SpaceTimeActions.html. 6

[3] A. Agarwal and B. Triggs. Recovering 3d human pose from monocular images. *PAMI*, 28, 2006. 2

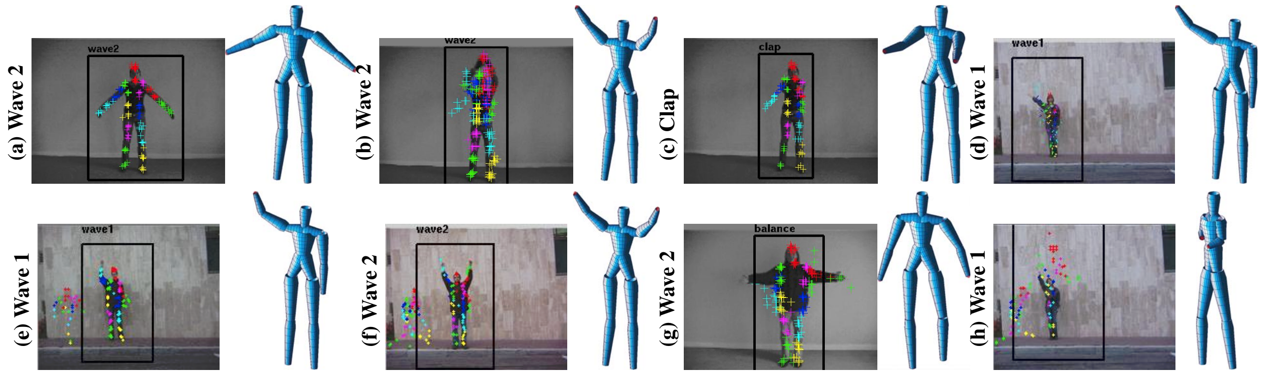


Figure 9: Sample results obtained from applying the same model trained in Section 4.2 to KTH (a–c, g) and Weizmann dataset (d–f, h).

[4] M. Andriluka, S. Roth, and B. Schiele. Pictorial structures revisited: People detection and articulated pose estimation. *CVPR*, 2009. 2

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

[6] A. Baak, M. Müller, G. Bharaj, H.-P. Seidel, and C. Theobalt. A data-driven approach for real-time full body pose reconstruction from a depth camera. In *ICCV*, 2011. 2

[7] L. Breiman. Random forests. *Machine Learning*, 2001. 4

[8] A. Criminisi, J. Shotton, D. Robertson, and E. Konukoglu. Regression forests for efficient anatomy detection and localization in ct studies. In *Int. MICCAI workshops*, 2011. 1, 4

[9] M. Eichner, M. Marin-Jimenez, A. Zisserman, and V. Ferrari. 2d articulated human pose estimation and retrieval in (almost) unconstrained still images. *IJCV*, 2012. 2, 6

[10] A. Elgammal and C.-S. Lee. Inferring 3d body pose from silhouettes using activity manifold learning. In *CVPR*, 2004. 1

[11] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *PAMI*, 2010. 2

[12] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient matching of pictorial structures. *CVPR*, 2000. 2

[13] M. A. Fischler and R. A. Elschlager. The representation and matching of pictorial structures. *IEEE Trans. Comput.*, 1973. 2

[14] P. Guan, A. Weiss, A. O. Balan, and M. J. Black. Estimating human shape and pose from a single image. In *ICCV*, 2009. 2

[15] D. Hogg. Model-based vision: a program to see a walking person. *Im. and Vis. Comput.*, 1983. 2

[16] S. Ioffe and D. Forsyth. Finding people by sampling. In *ICCV*, 1999. 2

[17] P. Kohli, M. Sun, and J. Shotton. Conditional regression forests for human pose estimation. *CVPR*, 2012. 2

[18] A. S. Micilotta, E.-J. Ong, and R. Bowden. Real-time upper body detection and 3d pose estimation in monoscopic images. In *ECCV*, 2006. 2

[19] R. Navaratnam, A. W. Fitzgibbon, and R. Cipolla. Semi-supervised learning of joint density models for human pose estimation. In *BMVC*, 2006. 2

[20] G. Pons-Moll, A. Baak, J. Gall, L. Leal-Taixé, M. Müller, H.-P. Seidel, and B. Rosenhahn. Outdoor Human Motion Capture using Inverse Kinematics and von Mises-Fisher Sampling. In *ICCV*. IEEE, 2011. 2

[21] R. Poppe. Vision-based human motion analysis: An overview. *Comput. Vis. Image Underst.*, 2007. 2

[22] K. Raja, I. Laptev, P. Perez, and L. Oisel. Joint pose estimation and action recognition in image graphs. In *ICIP*, 2011. 2

[23] V. Ramakrishna, T. Kanade, and Y. Sheikh. Reconstructing 3d human pose from 2d image landmarks. In *ECCV*, 2012. 2

[24] G. Rogez, J. Rihan, C. Orrite-Uruuela, and P. Torr. Fast human pose detection using randomized hierarchical cascades of rejectors. *IJCV*, 2012. 2

[25] K. Schindler and L. van Gool. Action snippets: How many frames does human action recognition require? In *CVPR*, 2008. 4

[26] L. Sigal, A. O. Balan, and M. J. Black. Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *IJCV*, 2010. 5

[27] L. Sigal, M. Isard, H. Haussecker, and M. J. Black. Loose-limbed people: Estimating 3d human pose and motion using non-parametric belief propagation. *IJCV*, 2012. 2

[28] E. Simo-Serra, A. Ramisa, G. Alenya, C. Torras, and F. Moreno-Noguer. Single image 3d human pose estimation from noisy observations. In *CVPR*, 2012. 3

[29] M. Sun, M. Telaprolu, H. Lee, and S. Savarese. An efficient branch-and-bound algorithm for optimal human pose estimation. In *CVPR*, 2012. 2

[30] J. Taylor, J. Shotton, T. Sharp, and A. W. Fitzgibbon. The vitruvian manifold: Inferring dense correspondences for one-shot human pose estimation. In *CVPR*, 2012. 2

[31] J. Wang, Z. Liu, Y. Wu, and J. Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *CVPR*, 2012. 1

[32] X. K. Wei and J. Chai. Modeling 3d human poses from uncalibrated monocular images. In *ICCV*, 2009. 2

[33] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, 2011. 1, 2, 3, 6, 7

[34] A. Yao, J. Gall, and L. Gool. Coupled action recognition and pose estimation from multiple views. *IJCV*, 2012. 1, 2, 5

[35] M. Ye, X. Wang, R. Yang, L. Ren, and M. Pollefeys. Accurate 3d pose estimation from a single depth image. In *ICCV*, 2011. 2

[36] T.-H. Yu, T.-K. Kim, and R. Cipolla. Real-time action recognition by spatiotemporal semantic and structural forest. In *BMVC*, 2010. 2