Compositional Human Pose Regression

Shuang Liang a,∗, Xiao Sun b, Yichen Wei b

a Tongji University, Shanghai, China
b Microsoft Research, Beijing, China

A R T I C L E   I N F O

Communicated by Nikos Paragios

MSC:
41A05
41A10
65D05
65D17

A B S T R A C T

Regression based methods are not performing as well as detection based methods for human pose estimation. A central problem is that the structural information in the pose is not well exploited in the previous regression methods. In this work, we propose a structure-aware regression approach. It adopts a reparameterized pose representation using bones instead of joints. It exploits the joint connection structure to define a compositional loss function that encodes the long range interactions in the pose. It is simple, effective, and general for both 2D and 3D pose estimation in a unified setting. Comprehensive evaluation validates the effectiveness of our approach. It establishes the new state-of-the-art on Human3.6M dataset. It is also competitive on MPII and COCO datasets.

1. Introduction

Human pose estimation has been extensively studied for both 3D (Ionescu et al., 2014) and 2D (Andriluka et al., 2014). Recently, deep convolutional neural networks (CNNs) have achieved significant progresses.

Existing approaches can be categorized into detection based and regression based. Detection based methods generate a likelihood heat map for each joint and locate the joint as the point with the maximum value in the map. The ambiguity in the heat maps is reduced by exploiting the dependence between the joints in various ways, e.g., by adopting a multi-stage architecture (Newell et al., 2016; Wei et al., 2016). Such methods are dominant for 2D pose estimation but do not easily generalize to 3D pose estimation (Pavlakos et al., 2016), because the 3D heat maps are too demanding for memory and computation.

Regression based methods directly map the input image to the output joints. The formulation is more direct and general. However, they are not performing very well. As an evidence, almost all leading methods on 2D pose benchmark (Andriluka et al., 2014) is detection based. While they are widely used for 3D pose estimation (Zhou et al., 2016; Moreno-Noguer, 2016; Mehta et al., 2016; Tekin et al., 2016b; Li et al., 2015; Tekin et al., 2016a; Park et al., 2016), the performance is not satisfactory. A central problem is that they simply minimize the per-joint location errors independently but ignore the internal structures of the pose.

In this work, we propose a structure-aware approach, called compositional pose regression. It is based on two ideas. First, it uses bones instead of joints as pose representation, because the bones are more primitive, more stable, and easier to learn than joints. Second, it exploits the joint connection structure to define a compositional loss function that encodes long range interactions between the bones.

The approach is simple and efficient. It only re-parameterizes the pose representation, i.e., the network output, and enhances the loss function, which relates the output to ground truth. It does not affect other algorithm choices and is compatible with them, such as network architecture. Thus, it can be easily combined with existing regression approaches with little overhead in memory and computation, in both training and inference.

The recent work "Integral Human Pose Regression" (Sun et al., 2018a,b) shows that the heat map based methods can be easily converted to regression based ones. The resulting approach unifies the two categories and is the state of the art on both 2D and 3D human pose estimation tasks. Our proposed approach can also be adopted in Sun et al. (2018b).

The approach is general and can be used for both 3D and 2D pose regression, indistinguishably. 2D and 3D data can be easily mixed simultaneously in the training. This property makes our approach different from all existing ones that target at either 3D or 2D task.

The effectiveness of our approach is validated by comprehensive evaluation and comparison with state-of-the-art on both 3D and 2D benchmarks. Specifically, it sets the state-of-the-art on 3D Human3.6M dataset (Ionescu et al., 2014) with 55.5 mm average joint error. On 2D MPII dataset (Andriluka et al., 2014), it is the best-performing regression based method (87.4% PCKh@0.5) and competitive with the state-of-the-art detection based methods.

A previous version of this work is in ICCV 2017 (Sun et al., 2017). In this work, all experiments are re-implemented (from Caffe to MxNet). More related works are compared. Our all results are considerably better than the previous versions.
improved. New results on COCO dataset are added. Obsolete metric and experiment results are removed. The technical content is restructured.

2. Related work

Human pose estimation has been extensively studied for years. A complete review is beyond the scope of this work. We refer the readers to Moeslund and Granum (2001) and Sarafianos et al. (2016) for a detailed survey.

We review the related work from two relevant viewpoints. **Exploit structure information for pose estimation**. Graphical models are widely adopted to represent the pose and its structural constraints. Representative works in early years include Felzenszwalb and Huttenlocher (2005), Johnson and Everingham (2011), Yang and Ramanan (2011), Pishchulina et al. (2013a), Pedersoli et al. (2015), Yang and Ramanan (2013), Lindner et al. (2015) and Chen and Yuille (2014). These works formulate pose estimation as an inference problem on the graph. The inference is usually complex, slow, and not integrated with deep networks. Recently, several works (Tompson et al., 2014; Ouyang and Wang, 2013; Chu et al., 2016; Yang et al., 2016) integrate the graphical models into the deep networks and have shown promising results.

Another category of methods use a multi-stage architecture (Wei et al., 2016; Newell et al., 2016; Carreira et al., 2016; Bulat and Tzimiropoulos, 2016; Gkioxari et al., 2016; Chu et al., 2017; Yang et al., 2017). In these works, the joint heat maps of the previous stage are used as inputs to regularize the learning of the next stage. The dependency between joint is implicitly learnt from data in the multi stage architecture.

Recent works in Chou et al. (2017) and Chen et al. (2017) use adversarial learning to distinguish physically valid poses and invalid ones, thus enforcing the structure constraints.

Most above works are based on heat maps and mostly used for 2D pose estimation. Extending the 2D heat maps to 3D is memory and computation demanding. Typically, this is infeasible. One exception is the work in Pavlakos et al. (2016), which extends the multi-stage framework in Newell et al. (2016) from 2D to 3D. A coarse-to-fine approach is used to alleviate the memory and computation problem.

3D pose estimation is more challenging. There lacks 3D data. Estimating depth from 2D images is ill-posed. Therefore, exploiting 3D structural information is more important as means of regularization during training.

Some methods implicitly learn the structure from data. Tekin et al. (2016a) represents the 3D pose with an over-complete dictionary. A latent pose representation is learned to account for joint dependencies. Li et al. (2015) uses an image-pose embedding sub-network to regularize the 3D pose prediction.

Some methods adopt two separate steps. 2D joints are first estimated and then used to estimate the 3D pose via learning a 2D to 3D mapping (Moreno-Noguer, 2016; Martinez et al., 2017; Fang et al., 2018), fitting a 3D model via optimization (Zhou et al., 2016a; Bogo et al., 2016; Zhou et al., 2017b; Mehta et al., 2017; Kanazawa et al., 2017), searching over a database (Chen and Ramanan, 2016), or generating multiple hypotheses (Jahangiri and Yuille, 2017). There is no end-to-end learning, because the images are not used in the second step.

Some methods explicitly use a 3D geometric model in learning. Zhou et al. (2016a,b) embed a kinematic model layer in the networks and estimate the model parameters instead of the joints. They need an accurate 3D model, e.g., with bone lengths in metric. They do not generalize to the 2D case, as a good 2D geometric model is absent.

Some methods exploit the temporal coherency or dependency between frames to learn a sequence of consistent 3D poses (Hossain and Little, 2017; Dabral et al., 2017). They are complementary to single frame methods.

Different to all above methods, our approach only changes the joint based pose representation and enhances the loss function by considering the joint connection structure in the pose. It does not make further assumptions. It is simple, fast, and complementary with existing methods.

**Leveraging in the wild 2D data for 3D pose estimation**. Motion capture is difficult. The largest 3D human pose dataset Human3.6M (Jonescu et al., 2014) is captured under controlled environment. It has limited complexity and diversity in the subjects, environment, and the poses presented by the subjects. Models trained on such data do not generalize well to other domains, such as in the wild images.

In contrast, in the wild images and 2D pose annotation are abundant. They are widely leveraged for 3D pose estimation. Most existing works consist of two separate steps.

Some methods (Moreno-Noguer, 2016; Martinez et al., 2017; Fang et al., 2018; Zhou et al., 2016a; Bogo et al., 2016; Zhou et al., 2017b; Mehta et al., 2017; Kanazawa et al., 2017; Chen and Ramanan, 2016; Jahangiri and Yuille, 2017) firstly generate the 2D pose results (joint locations or heat maps) and then use them as input for recovering the 3D pose, as reviewed above. The 2D images are discarded in the second step.

Some methods (Pavlakos et al., 2016; Tome et al., 2017) firstly train the deep network model on 2D data and fine-tune the model on 3D data. The information in 2D data is partially retained by the pre-training, but not fully exploited as the second fine-tuning step cannot use 2D data.

Some methods (Mehta et al., 2016; Park et al., 2016) train both 2D and 3D pose networks simultaneously by sharing intermediate CNN features. Yet, they use separate sub-networks and data for 2D and 3D tasks.

Unlike the above methods, our approach treats the 2D and 3D data in the same way and combine them in a unified training framework. The abundant information in the 2D data is fully exploited during training. As a result, our method achieves strong performance on both 3D and 2D benchmarks. As a by-product, it generates plausible and convincing 3D pose results for in the wild images. Note that the contemporary work in Zhou et al. (2017a) exploits the 2D data in the similar way.

3. Compositional pose regression

The pose estimation problem is to obtain the 2D or 3D position of all the K joints, \( J = \{ J_k | k = 1 \ldots K \} \), from an image of a person. Typically, the pose position coordinate is in **pixel** for 2D and metric, e.g., **millimeter (mm)**, for 3D.

Without loss of generality, the joints are defined with respect to a **constant origin point** in the image coordinate system. For notation convenience, let it be \( J_0 \). For 2D pose estimation, it is the top-left point of the image. For 3D pose estimation, it is the ground truth pelvis joint (Zhou et al., 2016a; Park et al., 2016).

For regression learning, normalization is necessary to compensate for the differences in variables’ magnitude. We use the standard normalization by subtraction of mean and division of standard deviation. For a variable \( \var \), it is normalized as

\[
\tilde{\var} = N(\var) = \frac{\var - \text{mean}(\var^{\text{gt}})}{\text{std}(\var^{\text{gt}})}.
\]

The inverse function for **unnormlization** is

\[
\var = N^{-1}(\tilde{\var}) = \var \cdot \text{std}(\var^{\text{gt}}) + \text{mean}(\var^{\text{gt}}).
\]

Note that both \( \text{mean}(\cdot) \) and \( \text{std}(\cdot) \) are constants. They are calculated from the ground truth training samples. Both functions \( N(\cdot) \) and \( N^{-1}(\cdot) \) are parameter free and embedded in the network. For notation simplicity, we use \( \tilde{\var} \) for \( N(\var) \).
3.1. Direct joint regression: A baseline

Most previous regression based methods (Carreira et al., 2016; Zhou et al., 2016a; Park et al., 2016; Tekin et al., 2016a,b) directly minimize the squared difference (L2 norm) of the predicted and ground truth joints. We use the absolute difference (L1 norm) as we found it better in the experiments. The loss is

\[ L(J) = \sum_{k=1}^{K} \| \mathbf{J}_k - \hat{\mathbf{J}}_k \|_1. \]  

(3)

Note that both the prediction \( \mathbf{J}_k \) and ground truth \( \hat{\mathbf{J}}_k \) are normalized.

Loss in Eq. (3) is flawed in that the joints are independently estimated. The joint correlation, or the internal structure in the pose, is not well exploited. For example, certain geometric constraints (e.g., bone length is fixed) are not satisfied.

3.2. A bone based representation

We show that a simple reparameterization of the pose is effective to address the above issue. As shown in Fig. 1 (left), a pose is structured as a directed tree. Without loss of generality, let pelvis be the root joint \( \mathbf{J}_1 \) and the edges be directed from the root to the end joints such as wrists and ankles. Let the function \( \text{parent}(k) \) return the index of the parent joint for \( k \)th joint. For notation consistency, let the parent of the root joint \( \mathbf{J}_1 \) be the origin \( \mathbf{J}_0 \), i.e., \( \text{parent}(1) = 0 \).

Now, for \( k \)th joint, its associated bone is defined as a vector

\[ \mathbf{B}_k = \mathbf{J}_{\text{parent}(k)} - \mathbf{J}_k. \]  

(4)

The joints \( \mathbf{J} \) are defined in the global coordinate system. In contrast, bones \( \mathbf{B} = \{ \mathbf{B}_k \mid k = 1, \ldots, K \} \) are defined in the local coordinate systems, more primitive, and brings several benefits.

**Stability** Bones are more stable than joints. Thus, they are easier to learn. Fig. 2 shows that the standard deviation of bones is much smaller than that of their corresponding joints, especially for parts (ankle, wrist, head) far away from the root pelvis, in both 2D and 3D datasets.

**Geometric convenience** Bones encode the local geometric structure more easily than joints. For example, geometric constraint of “bone length is fixed” involves one bone but two joints. Constraint of “joint rotation angle is in limited range” involves two bones but three joints. Experiments show that bone based representation is better than joint based representation to satisfy such constraints.

**Application convenience** Many pose-driven applications only need the local bones instead of the global joints. For example, the local and relative “elbow to wrist” motion can sufficiently represent a “pointing” gesture that would be useful for certain human computer interaction scenarios.

3.3. Compositional loss function

Similar to the joint loss in Eq. (3), bones can be learnt by minimizing the bone loss function

\[ L(B) = \sum_{k=1}^{K} \| \mathbf{B}_k - \hat{\mathbf{B}}_k \|_1. \]  

(5)

However, there is a drawback in this loss. As the bones are local and independently estimated in Eq. (5), the errors in the individual bone predictions would propagate along the skeleton tree and accumulate into large errors for joints at the far end. For example, in order to predict \( \mathbf{J}_{\text{elbow}} \) we need to concatenate \( \mathbf{B}_{\text{elbow}}, \mathbf{B}_{\text{forearm}}, \ldots, \mathbf{B}_{\text{pelvis}} \). Errors in these bones will accumulate and affect the accuracy of \( \mathbf{J}_{\text{elbow}} \) in an uncontrolled manner.

Long range loss functions are introduced to address the problem, as individual errors on the intermediate bones can be better balanced and the bones are jointly optimized. Specifically, let \( \mathbf{J}_i \) and \( \mathbf{J}_j \) be two arbitrary joints. Suppose that the path from \( \mathbf{J}_i \) to \( \mathbf{J}_j \) along the skeleton tree has \( M \) joints. Let the function \( I(m) \) return the index of the \( m \)th joint on the path, e.g., \( I(1) = u, I(M) = v \). Note that \( M \) and \( I(*) \) are constants but depend on \( u \) and \( v \). Such dependence is omitted in the notations for clarity.

As illustrated in Fig. 1 (right), the relative joint position \( \Delta \mathbf{J}_{I(m)} \) is the summation of the bones along the path, as

\[ \Delta \mathbf{J}_{I(m)} = \sum_{n=1}^{M-1} \Delta \mathbf{J}_{I(m+1)} - \Delta \mathbf{J}_{I(m)} = \sum_{n=1}^{M-1} \text{sgn}(\text{parent}(I(m)), I(m+1)) \cdot N^{-1}(\mathbf{B}_{I(m)}). \]  

(6)

The sign function \( \text{sgn}(*, *) \) indicates whether the bone \( \mathbf{B}_{I(m)} \) direction is along the path direction or not. It returns 1 when \( \text{parent}(I(m)) = I(m+1) \) and -1 otherwise. Note that the predicted bone \( \hat{\mathbf{B}}(*) \) is normalized. It is unnormalized via Eq. (2) before summation, as in Eq. (6).

Eq. (6) is differentiable with respect to the bones. It is efficient and has no free parameters. It is implemented as a special compositional layer in the neural networks.

The ground truth relative joint position is

\[ \Delta \mathbf{J}_{I(m)}^g = \mathbf{J}_{I(m)} - \hat{\mathbf{J}}_{I(m)}. \]  

(7)

Then, given a set \( P \) of joint pair, the compositional loss function is defined over all joint pairs as

\[ L(R, P) = \sum_{(u,v) \in P} \| \Delta \mathbf{J}_{I(u)} - \Delta \mathbf{J}_{I(v)} \|_1. \]  

(8)

In this way, every joint pair \( (u, v) \) constrains the bones along the path from \( u \) to \( v \). Each bone is constrained by multiple paths given a large number of joint pairs. The errors are better balanced over the bones during learning.

The joint pair set \( P \) can be defined arbitrarily. To validate the effectiveness of Eq. (8), we test four variants:

- \( P_{\text{joint}} = \{(u,v) | u = 1, \ldots, K \} \). It only considers the global joint locations. It is similar to joint loss Eq. (3).
- \( P_{\text{bone}} = \{(u, \text{parent}(u)) | u = 1, \ldots, K \} \). It only considers the bones. It degenerates to the bone loss Eq. (5).
- \( P^*_{\text{both}} = P_{\text{joint}} \cup P_{\text{bone}} \). It combines the above two and verifies whether Eq. (8) is effective.
- \( P_{\text{all}} = \{(u,v) | u < v, u, v = 1, \ldots, K \} \). It contains all joint pairs. The pose structure is fully exploited.

Our method corresponding to these variants is named as \( \text{JOINT}, \text{BONE}, \text{BOTH}, \) and \( \text{ALL} \), respectively. Similarly, the direct joint regression method described in Section 3.1 is named as \( \text{BASELINE} \), or \( \text{BL} \) for short.

3.4. Unified 2D and 3D pose regression

Our method is applicable for both 3D and 2D pose estimation in the same way. 3D and 2D data can be simply mixed during training. All the variables, such as joint \( \mathbf{J} \), bone \( \mathbf{B} \), the relative joint position \( \Delta \mathbf{J}_{I(m)} \), as well as the loss functions, are decomposed into \( x \)y part and \( z \) part. For example, \( \text{compositional loss function} \) Eq. (8) becomes

\[ L(R, P) = L_{xy}(R, P) + L_z(R, P). \]  

(9)

The \( x \)y term \( L_{xy}(*, *) \) is always valid for both 3D and 2D samples. The \( z \) term \( L_z(*, *) \) is only computed for 3D samples. For 2D samples, the term is set to 0 and no gradient is back-propagated from it.

The loss decomposition in Eq. (9) is for practical reasons, because it is the most convenient way to deal with mixed 2D and 3D data. For \( x \)y part, the data dimension is in pixel, as in common 2D pose dataset. For \( z \) part, the data dimension is in millimeter, as in Human 3.6M dataset. The difference in dimension is not a problem because of the normalization (Eq. (8), (1), (6), (2)) during training. During inference, the \( x \)y part is back-projected into 3D camera space using the provided camera parameters and a perspective projection model.
4. Datasets and evaluation metrics

Our approach is validated on three benchmark datasets. Human3.6M (Ionescu et al., 2014) is the largest 3D human pose benchmark. The dataset is captured in controlled environment. It consists of 3.6 millions of video frames. 11 subjects (5 females and 6 males) are captured from 4 camera viewpoints, performing 15 activities. The image appearance of the subjects and the background is simple. Accurate 3D human joint locations are obtained from motion capture devices. For evaluation, many previous works use the mean per joint position error (MPJPE) (see Table 3) (bottom). We call this metric

\[ \text{Joint Error} \]

Some works firstly align the predicted 3D pose and ground truth 3D pose with a rigid transformation using Procrustes Analysis (Gower, 1975) and then compute MPJPE (see Table 3) (top). We call this metric PA Joint Error.

MPII (Andriluka et al., 2014) is the benchmark dataset for single person 2D pose estimation. The images were collected from YouTube videos, covering daily human activities with complex poses and image appearances. There are about 25k images. In total, about 29k annotated poses are for training and another 7k are for testing. For evaluation, Percentage of Correct Keypoints (PCK) metric is used. An estimated joint location. They do not directly measure the accuracy of bones, or in general, internal structures in the pose.

Bone Error

\[ \sum_{i=1}^{T} \sqrt{(x_{i} - x_{i}^*)^2 + (y_{i} - y_{i}^*)^2 + (z_{i} - z_{i}^*)^2} \]

is the standard deviation of bone length, or bone length and zero variance. It is applicable for both 3D and 2D pose, therefore for all datasets.

The second measures the quality of bones for 3D poses in videos, that is, the mean per bone position error (MPJPE) (see Table 3) (bottom). We call this metric PA Joint Error.

COCO Keypoint Challenge (Lin et al., 2014) requires "in the wild" multi-person detection and pose estimation in challenging, uncontrolled conditions. The COCO train, validation, and test sets, containing more than 200k images and 250k person instances labeled with keypoints. 150k instances of them are publicly available for training and validation.

Bone based metrics

\[ \text{Bone Error} \]

\[ \sum_{i=1}^{T} \sqrt{(x_{i} - x_{i}^*)^2 + (y_{i} - y_{i}^*)^2 + (z_{i} - z_{i}^*)^2} \]

is the standard deviation of bone length, or Bone Std, computed over all testing frames of the same subject. It is motivated by the fact that the same subject should have constant bone length and zero variance. It therefore measures the stability of 3D pose estimation over time.

5. Experiments

Our training and network architecture is similar for all the three datasets. ResNet (He et al., 2016) (ResNet-50 on Human3.6M and MPII, ResNet-101 on COCO) is adopted as the backbone network. It is pre-trained on ImageNet classification dataset (Deng et al., 2009). The last fully connected layer is then modified to output 3K (or 2K) coordinates, and the model is fine-tuned on the target task and data.

MxNet (Chen et al., 2015) is used for implementation. Adam is used for optimization. Data augmentation includes random translation (±2% of the image size), scale (±25%), rotation (±30 degrees) and flip. The
Table 2
Results on all joints for the baseline and ours ALL method (same as in Table 1). The relative performance gain is shown in the subscript. Note that the left most column shows the names for both the joint (and the bone).

<table>
<thead>
<tr>
<th>Metric</th>
<th>Method</th>
<th>Joint Error</th>
<th>PA Joint Error</th>
<th>Bone Error</th>
<th>Bone Std</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>ALL</td>
<td>BL</td>
<td>ALL</td>
<td>BL</td>
</tr>
<tr>
<td>Average</td>
<td>60.6</td>
<td>55.5</td>
<td>46.8</td>
<td>44.7</td>
<td>43.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>16.6</td>
<td>15.4</td>
<td>23.4</td>
<td>25.8</td>
</tr>
<tr>
<td>Ankle(−Knee)</td>
<td>97.7</td>
<td>90.4</td>
<td>71.9</td>
<td>66.8</td>
<td>84.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>22.8</td>
<td>25.4</td>
<td>22.8</td>
<td></td>
</tr>
<tr>
<td>Kneef(−Hip)</td>
<td>56.7</td>
<td>55.2</td>
<td>53.2</td>
<td>52.4</td>
<td>56.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>18.4</td>
<td>17.8</td>
<td>18.4</td>
<td></td>
</tr>
<tr>
<td>Elif(−Pelvis)</td>
<td>24.2</td>
<td>24.1</td>
<td>40.3</td>
<td>38.1</td>
<td>24.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10.0</td>
<td>9.9</td>
<td>10.0</td>
<td></td>
</tr>
<tr>
<td>Head(−Neck)</td>
<td>53.5</td>
<td>45.1</td>
<td>32.7</td>
<td>29.6</td>
<td>28.9</td>
</tr>
<tr>
<td></td>
<td></td>
<td>14.1</td>
<td>15.3</td>
<td>14.1</td>
<td></td>
</tr>
<tr>
<td>Neck(−Thorax)</td>
<td>56.8</td>
<td>48.6</td>
<td>25.3</td>
<td>24.9</td>
<td>17.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>12.5</td>
<td>12.6</td>
<td>12.5</td>
<td></td>
</tr>
<tr>
<td>Thorax(−Pelvis)</td>
<td>39.8</td>
<td>36.3</td>
<td>30.5</td>
<td>28.2</td>
<td>39.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10.0</td>
<td>10.0</td>
<td>10.0</td>
<td></td>
</tr>
<tr>
<td>Wrist(−Elbow)</td>
<td>97.0</td>
<td>90.8</td>
<td>74.0</td>
<td>69.7</td>
<td>68.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>25.3</td>
<td>25.3</td>
<td>25.3</td>
<td></td>
</tr>
<tr>
<td>Elbow(−Shoulder)</td>
<td>76.2</td>
<td>70.3</td>
<td>52.7</td>
<td>50.8</td>
<td>55.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>16.1</td>
<td>14.9</td>
<td>14.9</td>
<td></td>
</tr>
<tr>
<td>Shoulder(−Thorax)</td>
<td>58.8</td>
<td>52.2</td>
<td>30.6</td>
<td>31.7</td>
<td>22.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8.0</td>
<td>7.4</td>
<td>7.4</td>
<td></td>
</tr>
</tbody>
</table>

Table 3
Comparison with previous works on Human3.6M, under protocol 1 (top) and protocol 2 (bottom). Ours ALL is the best single-image method. Note that Dabral et al. (2017) and Hossain and Little (2017) exploit temporal information and are complementary to our approach.

<table>
<thead>
<tr>
<th>Method</th>
<th>Score</th>
<th>Score</th>
<th>Score</th>
<th>Score</th>
<th>Score</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yasin et al. (2016)</td>
<td>108.3</td>
<td>88.1</td>
<td>55.3</td>
<td>46.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rogez and Schmid (2016)</td>
<td>108.3</td>
<td>88.1</td>
<td>55.3</td>
<td>46.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chen and Ramanan (2016)</td>
<td>108.3</td>
<td>88.1</td>
<td>55.3</td>
<td>46.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Moreno-Noguer (2016)</td>
<td>108.3</td>
<td>88.1</td>
<td>55.3</td>
<td>46.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Zhou et al. (2017b)</td>
<td>108.3</td>
<td>88.1</td>
<td>55.3</td>
<td>46.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dabral et al. (2017)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kanazawa et al. (2017)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hossain and Little (2017)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Martinez et al. (2017)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fang et al. (2018)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chenn et al. (2016)</td>
<td>114.2</td>
<td>88.4</td>
<td>87.3</td>
<td>79.9</td>
<td>72.9</td>
<td>66.9</td>
</tr>
<tr>
<td>Tome et al. (2017)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Moreno-Noguer (2016)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Zhou et al. (2017b)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mehta et al. (2016)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pavlakos et al. (2016)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dabral et al. (2017)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kanazawa et al. (2017)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hossain and Little (2017)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fang et al. (2018)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dabral et al. (2017)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Martinez et al. (2017)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 3. Joint errors (top) and bone errors (bottom) of baseline and our BONE method in Table 1 during training.

Fig. 4. (best viewed in color) Examples of 3D pose estimation for Human3.6M (top row) and MPII (bottom row), using ours ALL method in Table 3, trained with both 3D and 2D data. Note that the MPI 3D results are quite plausible and convincing.

Initial learning rate is 0.01. It drops to 0.001 after 90 epochs and 0.0001 after another 30 epochs. Mini-batch size is 128. Four GPUs are used. Other training details are provided in individual experiments.

5.1. Experiments on Human3.6M

In the literature, there are two widely used evaluation protocols. They have different training and testing data split.

Protocol 1 Six subjects (S1, S5, S6, S7, S8, S9) are used in training. Evaluation is performed on every 64th frame of Subject 11. PA Joint Error is used for evaluation.

Protocol 2 Five subjects (S1, S5, S6, S7, S8) are used in training. Evaluation is performed on every 64th frame of subjects (S9, S11). Joint Error is used for evaluation.

Training using mixed 3D and 2D data The training data in Human3.6M have limited diversity. Severe over fitting is observed when only Human3.6M training data is used. Consequently, most works uses additional training data. In this work, additional 2D data from MPII
dataset are used, as explained in Section 3.4.1. The input image is normalized to 256 × 256. Each mini-batch consists of half 2D and half 3D samples, randomly sampled and shuffled.

**Ablation study** Table 1 compares the baseline and our method, as summarized in Section 3.3. Protocol 2 is used. Two conclusions are observed. First, bone representation is superior than joint representation. This is observed by comparing the baseline with JOINT and BONE. Our two variants are better than the baseline on all the metrics. Interestingly, JOINT is better on two joint based metrics, while BONE is better on two bone based metrics. This is reasonable. Second, compositional loss is effective. When the loss function becomes better, e.g., in BOTH and ALL, result is further improved.

To further understand the benefit of bone representation, Fig. 3 compares the errors of two methods in Table 1 during training. Clearly, using bones (BONE) is better (faster convergence, more stable, smaller error) than using joints (BL).

Table 2 reports the results of all joints (and bones). Two conclusions are observed. First, limb joints are harder than torso joints and upper limbs are harder than lower limbs. This is consistent as Fig. 2. It indicates that the variance is a good indicator of difficulty. Second, our method improves the accuracy of all joints, especially the challenging ones like wrist, elbow and ankle.

**Comparison with the state-of-the-art** There are abundant previous works. They are compared to our method in Table 3. Note that such comparison is on system level and cannot be completely fair, because all works use extra 2D training data in different ways and adopt different network architectures.

Nevertheless, two observations validates the effectiveness of our approach. First, our baseline is strong enough to serve as a competitive reference. Compared to the previous state-of-the-art, it is only slightly worse than (Fang et al., 2018) in Table 3. Second, our method significantly improves the baseline (e.g., by 5.1 mm, relative 8.4%, in Table 3 bottom) and sets the new state-of-the-art (55.5 mm). Such improvement comes from the new pose representation and the loss function.

Example results are shown in Fig. 4. More results are at https://www.youtube.com/watch?v=c-hgHqVK90M.

### 5.2. Experiments on MPII

Training is almost exactly the same as in Section 5.1. The difference is that the network outputs 2D poses.2 Because the pose annotation on test set is not available, ablation experiments are performed on a validation set of 3k images, which is separated from the training set as in Newell et al. (2016). Training is performed on the remaining training data.

Ablation experiments in Table 4 show similar conclusions as in Table 1. ALL method is the best on all metrics. JOINT is good on joint metric but slightly worse on bone metric. BONE is on the contrary. They all perform better than the baseline.

Table 5 compares our approach to state-of-the-art methods, as reported on MPII benchmark.3 Best performing methods are mostly based on heat map representation, thus detection based. Ours is the best regression based method and is competitive with detection based methods.

---

1 Although a few works (Zhou et al., 2016c; Tekin et al., 2016b; Zhou et al., 2016a; Pavlakos et al., 2016) only use training data on Human3.6M, their results are usually much worse. See our previous work (Sun et al., 2017) for a comparison.

2 In our previous version (Sun et al., 2017), a two-stage network architecture is used, following the regression based method (Carreira et al., 2016). In this work, the result of single stage ResNet-50 network is significantly improved. The second stage is found no longer useful and discarded.


### 5.3. Experiments on COCO

We follow a two-stage top-down paradigm similar as in Papandreou et al. (2017), for both training and inference. First stage is for human detection. A Faster-RCNN (Ren et al., 2015) detector is used. ResNet-101 is adopted as the backbone network. The person detection AP on COCO test-dev is 0.49. For reference, the detector in Papandreou et al. (2017) is also based on ResNet-101 and achieves 0.487 AP, which is similar as ours.

The second stage training is similar as in Section 5.1 and 5.2. The ground truth bounding box is cropped from the original image and normalized to 353 × 257 as in Papandreou et al. (2017). For training data, we use the COCO train subset.

Ablation experiment is performed on COCO val2017 subset. Results in Table 6 are consistent with those in Tables 1 and 4. ALL is the best. It improves the bone error metric over the baseline considerably.

For completeness, we also report the results on COCO test-dev subset from public literature. The AP metric is 61.8 in Cao et al. (2016), 63.1 in He et al. (2017), and 64.9 in Papandreou et al. (2017), which compares favorably with ours 62.4. Note that such comparison is not on a fair basis due to many difference in these works but just as a reference. Also note that other works are heat map based, while ours is regression based.

### 6. Conclusion

We show that regression based approach is competitive to the leading detection based approaches for 2D pose estimation once pose structure is appropriately exploited. Our approach is more advantageous for 3D pose estimation, where considering structure constraints is more important.