LCR-Net: Localization-Classification-Regression for Human Pose

Gregory Rogez, Philippe Weinzaepfel, Cordelia Schmid

LCR-Net: Localization-Classification-Regression for Human Pose

Grégory Rogez[1] Philippe Weinzaepfel[2] Cordelia Schmid[1]

[1]Inria* [2]Xerox Research Centre Europe

Abstract

We propose an end-to-end architecture for joint 2D and 3D human pose estimation in natural images. Key to our approach is the generation and scoring of a number of pose proposals per image, which allows us to predict 2D and 3D pose of multiple people simultaneously. Hence, our approach does not require an approximate localization of the humans for initialization. Our architecture, named LCR-Net, contains 3 main components: 1) the pose proposal generator that suggests potential poses at different locations in the image; 2) a classifier that scores the different pose proposals; and 3) a regressor that refines pose proposals both in 2D and 3D. All three stages share the convolutional feature layers and are trained jointly. The final pose estimation is obtained by integrating over neighboring pose hypotheses, which is shown to improve over a standard non maximum suppression algorithm. Our approach significantly outperforms the state of the art in 3D pose estimation on Human3.6M, a controlled environment. Moreover, it shows promising results on real images for both single and multi-person subsets of the MPII 2D pose benchmark.

1. Introduction

State-of-the-art methods for 2D human pose estimation in real images obtain excellent performance using Convolutional Neural Network (CNN) architectures [4, 19]. However, occlusion still remains a significant challenge as analyzed in [19]. Numerical evaluations do not clearly reflect this fact since occluded joints are often not labeled, and never evaluated in standard datasets. In many cases of occlusions, the pose is not ambiguous and can still be estimated entirely. One way to recover body part locations in such cases is to reason about the full-body 3D pose. Methods for 3D human pose understanding require training data that is only available through Motion Capture (MoCap) systems. Even if they show accurate pose estimation results (including occluded joints) in controlled environments, these approaches do not generalize well to real images, with the exception of recent work based on data synthesis that shows promising results in the wild [5, 23]. In this paper, we propose a method that results in multiple full-body 2D and 3D pose hypotheses in different regions of the image. These *pose proposals* are efficiently sampled, scored and refined using an end-to-end CNN architecture inspired by the latest work on object detection [22]. Finally, the pose proposals are combined to estimate both the location and the 2D/3D pose of the individuals present in the observed scene. Our method recovers full-body poses, even when the persons are partially occluded or truncated by the image boundary, see Figure 1.

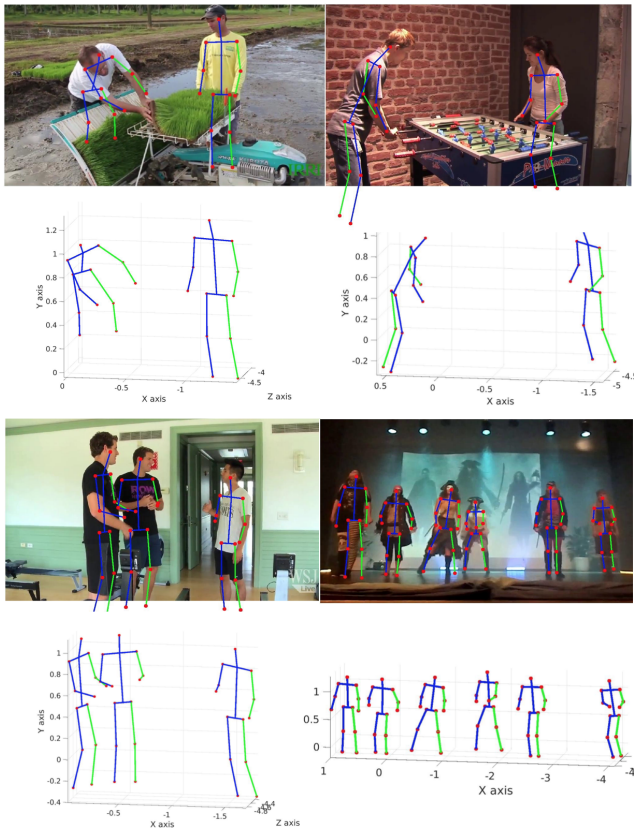

Figure 1. Examples of joint 2D-3D pose detections in natural images. Even in case of occlusion or truncation, we estimate the joint locations by reasoning in term of full-body 2D-3D poses.

CNNs have been used for full-body pose estimation both

*Thoth team, Inria, Laboratoire Jean Kuntzmann, Grenoble, France.

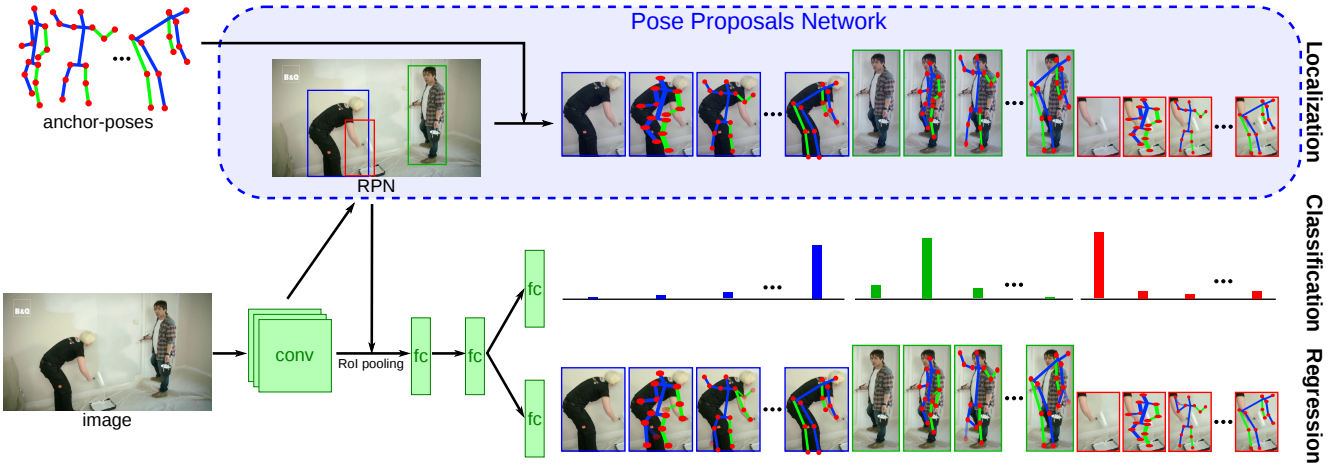

Figure 2. Overview of our LCR-Net architecture (poses only shown in 2D for better readability). We first extract candidate regions using a RPN network and obtain pose proposals by placing a fixed set of anchor-poses into these boxes (top). These pose proposals are then scored by a classification branch and regressed using a regressor, learned independently for each anchor-pose.

in regression [14, 31] and classification [23] approaches. Regression networks are trained to directly estimate the 2D or 3D location of the body joints, whereas a classification approach defines pose classes and returns the average pose of the top scoring class. Increasing the number of clusters improves precision of the estimation in classification approaches but makes discrimination harder. Regression methods can only predict one pose for a given image and fail to model multi-modal outputs, *e.g.* for ambiguous cases. In this paper, we argue that for full-body human pose estimation, the discriminative power of classification networks can be combined with the smoothness of regression methods by a simple yet elegant modification within the learning procedure. The architecture is similar in spirit to Faster R-CNN [22] which jointly localizes and classifies objects while regressing a refined bounding box. The key idea of our approach is to quantify the space of valid full-body poses and jointly train a K-way classifier on this partitioned space as well as local pose regression models. To this end, we formulate a joint classification-regression loss function that combines coarse pose classification and class-specific pose regression. Given a set of K hypothetical pose classes, we output for each proposed image region a list of K refined 2D/3D poses and the associated classification scores.

In summary, we propose an end-to-end architecture that detects 2D and 3D poses in natural images, see Figure 2. The network proceeds by extracting candidate regions for the person localization. We obtain *pose proposals* by locating the set of K hypothetical pose classes, denoted as anchor-poses, in these candidate boxes. Each pose proposal is then scored using a classification branch and regressed independently for each anchor-pose. The localization, *i.e.*, extraction of the pose proposals, classification and per anchor-pose regression, share layers and can be trained end-to-end. Our final output consists in a number of 2D/3D poses per images that are obtained by aggregating similar pose proposals, in terms of location and 3D pose. Our approach significantly outperforms the state of the art for 3D pose estimation in a controlled environment, even when compared to methods that leverage temporal smoothing and/or rely on initial localization of the human. We show promising results in real images, estimating the poses both in 2D and 3D, even in case of occlusions and truncations.

This paper is organized as follows. After reviewing the related work in Section 2, Section 3 introduces our proposed LCR-Net for pose detection. We present extensive experimental results, both in 2D and 3D, in Section 4.

2. Related work

Human localization and 2D pose estimation. Most state-of-the-art approaches for 2D human pose estimation employ CNNs [4, 6, 8, 14, 15, 19, 20, 30, 31]. They can be divided into two groups: (a) methods which first search the image for local body parts and model their dependencies using biologically inspired graphical models [6, 30]; and (b) holistic approaches that directly estimate the full body [4, 14, 15, 19, 31]. Most of these approaches assume that the individuals have been localized. Most similar to our approach are the methods that jointly localize humans and estimate their 2D pose [10, 11, 21]. They often rely on multi-stage architectures, whereas our network is trained in an end-to-end fashion. Importantly, they estimate the 2D location of the visible joints while we provide an estimation of the full-body 2D and 3D poses, even in case of occlusions.

3D human pose from a single image. Due to the lack of large scale training data, 3D methods are usually trained (and tested) on 3D MoCap data in constrained environments [14, 15, 35]. Some recent approaches employ CNNs

for 3D pose estimation in monocular images [5, 15, 23] or in videos [29, 37]. Some work also tackles 3D pose estimation from 2D poses assuming that the 2D joints are available [1, 9] or provided by a 2D pose detector [3, 26, 32]. Most of them reason about geometry. Finally, other methods solve 2D and 3D pose estimation jointly or iteratively [25, 35, 36]. Most similar to us are [5, 23] who train and compare the performance of 2D/3D pose regressors and classifiers in real images. They require a well-aligned bounding box around the subject while we jointly localize and estimate 2D and 3D pose. Moreover, we combine classification and regression in an effective manner.

3. LCR-Net

We propose to detect human poses using a Localization-Classification-Regression Network (LCR-Net). In this paper, a human pose (p, P) is defined as the 2D pose p, *i.e.*, the pixel coordinates of each joint in the image; and the 3D pose P, *i.e.*, 3D location of each joint relative to the body center (in meters). We consider poses with 13 joints. We assume that a fixed set of K 2D/3D anchor-poses is given, denoted by $\{(a_k, A_k)\}_{k=1..K}$. In this paper, they are obtained by clustering a large set of poses and using the center of each cluster as anchor pose, see Section 4 for details.

Figure 2 shows an overview of our LCR-Net architecture. Given an image, we first compute convolutional features. The *Localization* component, also called Pose Proposals Network in the context of pose detection, outputs a list of pose proposals. Pose proposals consist of a set of candidate locations where the anchor-poses are hypothesized. Next, a Region-of-Interest (RoI) pooling layer aggregates the features inside each candidate region. After two fully-connected layers, the network is split into two components. The *Classification* branch estimates the probability of anchor-poses to be correct at each location. It thus jointly learns to localize humans, as well as to estimate which anchor-pose is more probable. The *Regression* branch computes an anchor-pose-specific regression that estimates the difference between the true human pose and the pose proposal (Fig. 3). Our loss is the sum of three losses that we describe in more details in the following:

$$\mathcal{L} = \mathcal{L}_{Loc} + \mathcal{L}_{Classif} + \mathcal{L}_{Reg} \ . \tag{1}$$

Note that the convolutional features are shared between the three components and that the classification and regression branches also share features from two fully-connected layers. The architecture allows end-to-end training for localizing humans and estimating their poses, in contrast to most previous works which run a human detector before estimating pose.

3.1. Localization: pose proposals network

The Pose Proposal Network outputs a set of pose proposals, *i.e.*, candidate localized poses. To this end, we hypothesize a set of anchor-poses into a set of bounding boxes, that will be scored and refined by the classification and regression branches respectively. The set of bounding boxes is obtained using a Region Proposal Network (RPN) [22], see Figure 2. The loss of the localization component is the loss of the RPN network:

$$\mathcal{L}_{Loc} = \mathcal{L}_{RPN} \ . \tag{2}$$

During training, each bounding box B is labeled with a ground-truth class $c_B \in \{0 \dots K\}$ and a pose regression target t_{c_B}. The ground-truth class c_B is set to 0 (corresponding to background) if the bounding box has an Intersection over Union (IoU) below 0.5 with all ground-truth poses. The IoU between a box and a pose is computed using the bounding box around all joints of the pose, with a fixed additional margin of 10%. If B has a high overlap with several poses, let (p, P) be the ground-truth pose with the highest IoU with the box. The label c_B is set to $c_B = \text{argmin}_k \, D(A_k, P)$ where $D(.,.)$ is the distance between oriented 3D poses centered at the torso. This label will be used by the classification branch (Section 3.2). If the label c_B is non-zero, we also define a pose regression target, used in the regression branch (Section 3.3), t_{c_B} for the box B as $t_{c_B} = (\tilde{p} - \tilde{a}_{c_B}, P - A_{c_B})$ where $\tilde{p}$ and $\tilde{a}_{c_B}$ denote the 2D pose and anchor-pose normalized in the range $[0..1]$ according to the box coordinates (see Fig. 3). This normalization makes the regression independent of scale and position of the person and the box in the image.

3.2. Classification

The classification component aims at predicting the closest anchor-pose, *i.e.*, the correct label, for each bounding box B. In other words, each bounding box is assigned a probability for each anchor-pose (and the background class). Let u be the probability distribution estimated by the network, obtained by three fully-connected layers after RoI pooling, see Figure 2, followed by a softmax. The classification loss is defined using the standard log loss of the true class:

$$\mathcal{L}_{Classif}(u, c_B) = -log \, u(c_B) \ . \tag{3}$$

3.3. Regression

The regression component aims at refining the coarse anchor-poses located in the region proposals as depicted in Figure 3. The specificity of our approach is that the regression is anchor-pose-specific and a regressor is learned independently for each anchor-pose. The regression outputs v are obtained by using a fully-connected layer after

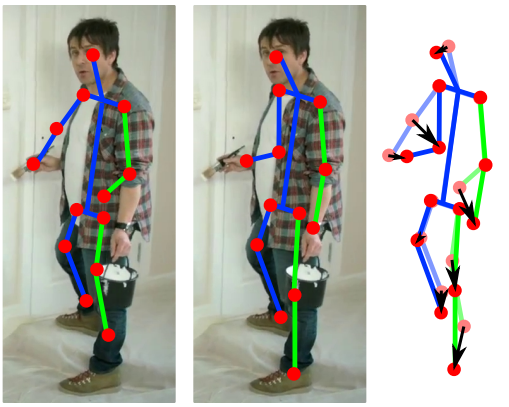

Figure 3. The regression aims at refining the anchor-pose to match the ground-truth pose of the individual (poses only shown in 2D for better readability).

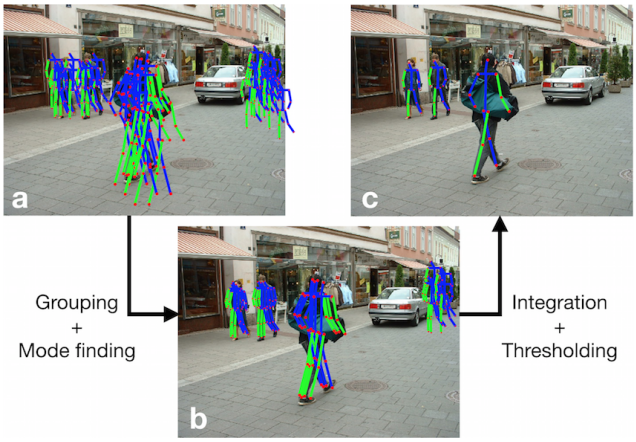

Figure 4. Pose proposal integration (PPI). The pose proposals (a) are grouped based on 2D overlap and 3D pose to identify the persons and the modes (b). Final pose estimates are obtained by averaging the 2D poses in the selected modes (c).

the two fully-connected layers shared with the classification branch (see Figure 2). The dimension of v is equal to $(K+1) \times 5 \times \#joints$, where the factor of 5 reflects the components of the 2D and 3D coordinates. We denote by v_{c_B} the subvector of v corresponding to the regression for anchor-pose c_B. The regression loss is defined as:

$$\mathcal{L}_{Reg}(v, t_{c_B}) = [c_B \geqslant 1] \ \|t_{c_B} - v_{c_B}\|_S \ , \qquad (4)$$

with $\|.\|_S$ the smooth-L1 loss, a robust version of the L2 loss which is less sensitive to outliers:

$$\|x\|_S = \begin{cases} 0.5x^2 & \text{if } |x| < 1, \\ |x| - 0.5 & \text{otherwise.} \end{cases} \qquad (5)$$

3.4. Implementation details

Similar to Faster R-CNN, we use an approximate joint training version, in which boxes are considered as fixed by the RoI pooling layer. We use the same parameters as [22] for RPN. For the classification and regression loss, we use 256 boxes per batch, with 32 boxes coming from 8 different images, *i.e.*, from more images than in the standard version. We have more labels and, consequently, we need more diversity inside each batch. One quarter of the boxes are on humans, the remaining ones on background. The network is based on VGG-16 architecture [27] and the weights are initialized with ImageNet pretraining.

3.5. Pose proposals integration

Our LCR-Net outputs a set of refined pose proposals with associated classification scores $s(p, P) = u(c_B)$ from Equation 3. Multiple proposals cover each person present in the image. One possibility is to use a non-maximum suppression algorithm (NMS) and return the top scoring proposal for a given region as estimated pose. Instead, we propose to aggregate proposals which are close in terms of image location and 3D pose. We refer to this post processing stage as the pose proposal integration (PPI), see Figure 4.

We start with grouping pose proposals with a sufficient spatial overlap in the 2D image, *i.e.*, an IoU above a certain threshold for the bounding boxes around the 2D joints. We take the top scoring proposal in the image and determine all the pose proposals that overlap sufficiently with this top scoring proposal. We repeat this step with the remaining pose proposals and their top scoring elements until no pose proposals are left.The resulting groups are coherent in terms of spatial overlap but can consist of very different 3D poses and hence the modes in 3D pose space need to be identified. Let $\mathcal{P} = \{(p, P)\}$ be the set of pose proposals in a group, each one with a classification score $s(p, P)$. We first pick the proposal with the highest score, *i.e.*, $(p^*, P^*) = \text{argmax}_{(p,P)\in\mathcal{P}} s(p, P)$. We then select the set $\mathcal{P}'$ of pose proposals in the group $\mathcal{P}$, for which the 3D distance D from P^* is below a threshold T_{3D}:

$$\mathcal{P}' = \left\{(p, P) \in \mathcal{P} \mid D(P^*, P) < T_{3D}\right\} \ . \qquad (6)$$

This selection ensures that we do not average poses that belong to different modes. We then obtain our final 2D pose p (and similarly the 3D pose) by averaging the 2D poses in mode $\mathcal{P}'$ weighted by their scores:

$$p = \frac{1}{S} \sum_{(q,Q)\in\mathcal{P}'} s(q, Q) \times q \ , \qquad (7)$$

with S the sum of the individual scores, *i.e.*, $S = \sum_{(q,Q)\in\mathcal{P}'} s(q, Q)$. The score for this pose p is set to S, which results in a higher score for poses with multiple pose proposals. We iterate this process, starting from the highest scored pose among the ones that have not yet been covered by a mode.

Number K of anchor-poses	1	50	100	200	500
LCR-Net + NMS	74.2	71.7	70.7	77.0	85.6
LCR-Net + NMS + Align.	64.5	59.3	**58.0**	63.6	66.1

Table 1. Average 3D pose error (mm) with respect to the number K of anchor-poses on Human3.6M, protocol 1 (P1), after 100k iterations. Results are reported for NMS with/without rigid alignment.

4. Experimental results

In this paper, we address joint 2D and 3D human pose detection in natural images. To the best of our knowledge, there exists no dataset with 3D annotations for real-world images. To evaluate our method, we thus perform separate experiments on (a) 3D pose estimation in a controlled environment on the Human3.6M dataset [12] (Section 4.1), and (b) 2D and 3D pose estimation in natural images on the MPII human pose dataset [2] (Section 4.2).

4.1. 3D pose detection on Human3.6M

The Human3.6M dataset [12] contains 3.6M human poses from 11 actors performing 17 different scripted actions. The videos are captured in a controlled environment from 4 different camera viewpoints while accurate 3D poses are measured using a MoCap system. Accurate 2D poses are also available for each camera view. To exhaustively compare our results with the state of the art, we use three different protocols. The first one, denoted as P1, is introduced in [13] and employed in [23, 35]: six subjects (S1, S5, S6, S7, S8 and S9) are used for training and every 64^{th} frame of subject S11/camera 2, *i.e.*, a total of 928 frames, are used for testing. We report the 3D pose error (mm), averaged over the 13 joints. As in [35], we report a 3D pose error that measures accuracy of pose aligned with a rigid transformation (Align.), but also report the absolute error (Abs.). The second protocol, denoted as P2, is used in [15, 29, 37]. All the frames from subjects S9 and S11 are used for testing and only S1, S5, S6, S7 and S8 are used for training. We evaluate only on every 5^{th} frame as in [37] , *i.e.*, on a test set of 110,000 images, as we did not observe a significant impact on performance when evaluating on all the frames. The last protocol P3, introduced by Bogo et al. [3], uses the same subjects for training and testing as P2. However, evaluation is performed only on sequences from camera 3 / trial 1 after rigid alignment.

Anchor-poses. We select a subset of the training set, *i.e.* 300,000 images and the corresponding 3D poses, and build a set of anchor-poses by clustering the 3D poses using K-means. Table 1 shows performance when varying the number K of anchor-poses after 100k iterations. We can see that best performance is obtained for K=100. When K is too small, for instance if K=1 which corresponds to a standard regression, the number of anchor-poses is not sufficient to cover the pose space. When K becomes too large, the error also increases since the anchor-poses are too similar, resulting in ambiguities in the classification.

	Abs. P1	Align. P1	Abs. P2	Align. P2	Align. P3
Kostrikov & Gall [13]	-	115.7	-	-	-
Iqbal *et al.* [35]	-	108.3	-	-	-
Rogez & Schmid [23]	126	88.1	121.2	87.3	-
Bogo *et al.* [3]	-	-	-	-	82.3
LCR-Net + NMS	65.9	55.6	89.8	72.3	73.1
LCR-Net + PPI	**63.2**	**53.4**	**87.7**	**71.6**	**72.7**

Table 2. Comparison with state-of-the-art results on Human3.6M for 3 different protocols. The average 3D pose error (mm) is reported before (Abs.) and after rigid 3D alignment (Align.) for protocols P1 and P2. See text for details. The errors are globally higher with protocols P2 and P3 that provide less training subjects and have a larger and more varied test set.

Comparison with the state of the art. Table 2 compares our methods (with $K = 100$, 150k iterations) to the state of the art on the three protocols P1, P2 and P3. Many approach report results only on P2, we compare to them in Table 3 and also present a per-class comparison. We significantly outperform other methods for the 3 protocols. This is despite the fact that we perform also localization, in contrast to most methods such as [23, 37] that assume bounding box annotation of the human. Some of the competing methods on P2 only evaluate on 6 actions [14, 15, 16, 28], other leverage temporal information [7, 29, 37]. We can observe that our proposed postprocessing PPI improves over a simple NMS for all the 15 actions and that we outperform all competing methods with an average 3D pose error of 87.7 mm for 15 actions and 83.0 mm for 6 actions. Our method is state of the art for 9 out of 15 actions but performs lower than [24] for 4 different actions. The method from [24] relies on 2D joints detected by [34] while our architecture is trained end-to-end using Human3.6M training set only. For the "Walk" and "WalkTogether" actions, we perform lower than [29] who leverages temporal information, an important clue for such actions. Our method could be extended to leverage additional temporal information, which should further improve the performance.

Impact of PPI. We experimentally set T_{3D} to 200 mm and found that the IoU threshold has no influence on the performance for this dataset, as only one individual is observed and all highly scored proposals are localized on the subject. In most cases, the best scoring pose proposal (NMS) is already an accurate estimation but, on average, the improvement achieved by our PPI over the NMS estimates is non negligible. In Figure 5, we show some qualitative results where examples are sorted by increasing 3D pose error. A green upward peak with respect to the blue curve corresponding to PPI indicates an important improvement by the PPI, whereas a red peak downward indicates poses where the rigid alignment helps correct the most. For the 928 test frames of protocol P1, less than 20 have an error greater than 130 mm. This occurs in cases of unseen poses in the training set, see rightmost example in Figure 5.

Method	Im	Loc	Directions	Discussion	Eat	Greet	Phone	Pose	Purchase	Sit	SitDown
Tekin *et al.* [29]		✓	102.4	147.7	88.8	125.3	118.0	112.3	129.2	138.9	224.9
Zhou *et al.* [37]			87.4	109.3	87.0	103.2	116.2	106.9	99.8	124.5	199.2
Du *et al.* [7]		✓	85.1	112.7	104.9	122.1	139.1	105.9	166.2	117.5	226.9
Li *et al.* [14]	✓		-	148.8	104.0	127.2	-	-	-	-	-
Li *et al.* [15]	✓		-	134.1	97.4	122.3	-	-	-	-	-
Li *et al.* [16]	✓		-	133.5	97.6	120.4	-	-	-	-	-
Tekin *et al.* [28]	✓		-	129.1	91.4	121.7	-	-	-	-	-
Rogez & Schmid [23]	✓		94.5	110.4	109.3	143.9	125.9	95.5	89.8	134.2	179.2
Sanzari *et al.* [24]	✓	✓	**48.8**	**56.3**	96.0	84.8	96.5	**66.3**	107.4	116.9	129.6
LCR-Net + NMS	✓	✓	79.8	84.5	76.4	86.6	94.2	81.6	74.2	106.3	129.4
LCR-Net + PPI	✓	✓	76.2	80.2	**75.8**	**83.3**	**92.2**	79.0	**71.7**	**105.9**	**127.1**

Method	Im	Loc	Smoke	Photo	Wait	Walk	WalkDog	WalkTogether	Avg. (All)	Avg. (6)
Tekin *et al.* [29]		✓	118.4	182.7	138.7	**55.1**	126.3	**65.8**	125.0	121.0
Zhou *et al.* [37]			107.4	143.3	118.1	79.4	114.2	97.7	113.0	106.1
Du *et al.* [7]		✓	120.0	135.9	117.6	99.3	137.4	106.5	126.5	118.7
Li *et al.* [14]	✓		-	189.1	-	77.6	146.6	-	-	132.2
Li *et al.* [15]	✓		-	166.2	-	68.5	132.5	-	-	121.3
Li *et al.* [16]	✓		-	163.3	-	73.7	135.2	-	-	121.6
Tekin *et al.* [28]	✓		-	162.2	-	65.7	130.5	-	-	116.8
Rogez & Schmid [23]	✓		123.8	160.3	133.0	77.4	129.5	91.3	121.2	119.5
Sanzari *et al.* [24]	✓		97.8	**105.6**	65.9	92.6	130.5	102.2	93.1	-
LCR-Net + NMS	✓	✓	90.5	106.5	86.5	64.8	92.5	84.2	89.8	85.2
LCR-Net + PPI	✓	✓	**88.0**	105.7	**83.7**	64.9	**86.6**	84.0	**87.7**	**83.0**

Table 3. Per-class results on Human3.6M with protocol P2 without pose alignment. Im refers to image-based approaches working at the frame level, *i.e.*, that do not leverage temporal information. Loc refers to methods that also perform localization of the person, *i.e.*, do not assume that a bounding box around the human is given. Note that Du *et al.* [7] only evaluate on camera 2.

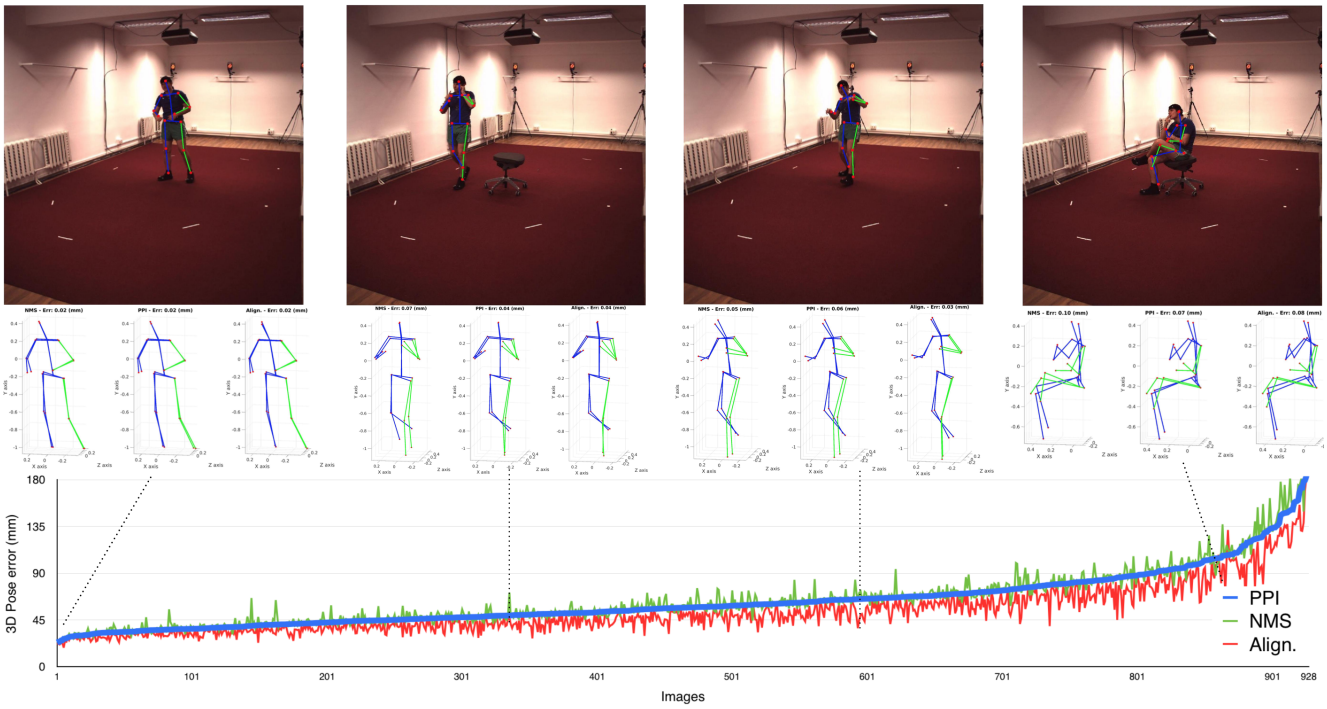

Figure 5. Average 3D pose error on Human3.6M test images (protocol P1). We order the examples by increasing error of PPI results (blue) and also report the performance with a simple NMS (green) and after rigid alignment of the PPI estimation (red). We show qualitative results for 4 particular cases, from left to right: 1) an image where NMS estimation is already accurate, thus PPI and alignment do not further improve, 2) a case in which the PPI achieves an accurate pose estimate, 3) a case where PPI does not improve over NMS but the alignment helps to correct the pose estimate and 4) a failure case where the pose is not satisfactory, even after rigid alignment. For each case, we show the image with the estimated 2D pose (with PPI). We also show the 3D poses estimated by NMS, PPI and after alignment overlaid with the ground-truth 3D pose.

4.2. 2D and 3D pose detection on MPII

We now present experimental results for 2D and 3D pose detection in real-world images. We use the challenging MPII human pose dataset [2] that consists of around 40k annotated 2D poses in around 25k images (17,400 for training and 7k for testing). It contains a large variety of camera viewpoints and poses, originating from around 400 different actions. Each scene can contain multiple people, that are often occluded or truncated by the image boundary. This makes the dataset challenging for human pose estimation. While most other papers on 3D pose estimation only show qualitative examples on real images, we analyze our results on a validation set of 1000 images that we used for both single (1088 poses) and multi-person (209 groups) protocols. This set is obtained by randomly splitting the training dataset to create a training set of 16k images and a validation set of 1000 images, making sure that images from the same video all belong to the same set. For training, we also use the annotated images from LSPE as in [21, 23] and a subset of 17k images from Human3.6M. After mirroring, we obtain a training set of 90k images.

Pseudo ground-truth 3D pose. To train our network, we need 3D ground-truth poses associated with each training image but MPII and LSPE only provide 2D joint locations. We infer ground-truth 3D poses from 2D annotations using a simple nearest neighbor (NN) search on the annotated joints. MoCap 3D poses are projected orthographically on multiple random virtual views to generate a very large set of 2D poses and a search is performed on the normalized 2D poses to estimate the closest match, *i.e.*, 3D pose + camera view. As in [23, 35], we consider the CMU MoCap dataset as 3D pose source. However, both MPII and LSPE datasets present rare poses (*e.g.* gymnastic) that are absent from this dataset. To cover a wider set of poses, we merge several MoCap datasets available on the internet, such as Pose Prior [1] and HDM05 [18], and observed a 13% reduction in the matching error, *i.e.*, distance between query 2D pose and best match, when using this augmented dataset. These recovered 2D poses are also used to complete the missing 2D annotations (due to occlusions or truncations) so that each training instance is associated with full-body 2D and 3D annotations. The set of anchor-poses is obtained by running K-means on the 3D poses of the extended MoCap dataset. Compared to Human3.6M, the diversity in pose is significantly higher but we found that $K = 100$ was still performing well.

Dealing with truncation. To deal with truncations by the image boundaries, we double the number of clusters by considering also upper-body region proposals. More precisely, for the K anchor-poses, we adjust the full-body anchor-pose such that only the upper-body covers the candidate box. At training, we define an upper-body ground-truth box for each annotated pose plus a fully-body ground-truth box when at least one joint from the lower limbs is visible.

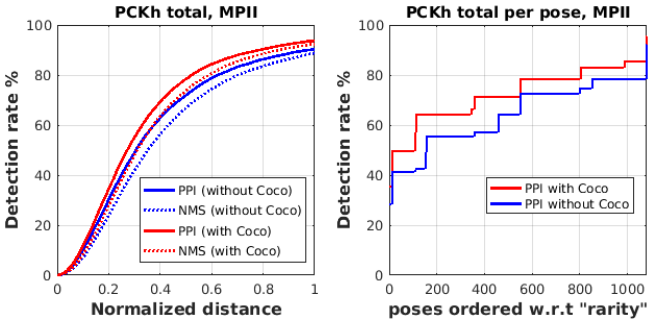

Figure 6. Average PCKh on MPII validation set. *Left:* Detection rate with respect to the normalized distance. *Right:* "Per-pose" PCKh@0.5 when ordering the poses with respect to"pose rarity". See text for details. For both plots, we show the effect of adding Coco [17] to the training set.

Single person pose estimation. In this setting, most methods use person localization information before computing the pose. In our case, we detect the poses over the whole image and use the localization information only for evaluation, *i.e.*, to select the pose that corresponds to each ground-truth. We report the results using the PCKh metric that measures the ratio of estimated joints for which the distance to the ground-truth is below a threshold. The standard threshold is set to half the size of the head. In Figure 6 left, we show the PCK while varying the ratio δ of the size of the head between 0 and 1, denoted as PCKh@δ. On our validation set, we obtain around 75% for a standard PCKh@0.5 and around 90% for a PCKh@1. We can see that PPI, with $T_{3D} = 100$mm and $IoU = 0.08$, improves with respect to NMS. We made further experiments to understand the influence of the training data on the performance and added annotated images from Coco [17] to approximately double the size of our training set. We observed a significant improvement in performance reaching $PCKh@0.5 = 78.5\%$ (Figure 6 left) on our validation set. This indicates that our method requires a significant amount of training data that could be generated by synthesis in future work. When increasing the training data, we observed that performance is slightly better for K=200 (rather than K=100), meaning that we better populate the pose clusters. We also observed a correlation between performance and rarity of the pose measured as the distance to the closest cluster (Figure 6 right). Since our approach is holistic and reasons about the full-body pose, our architecture can interpolate new full-body poses but does not extrapolate well unseen body configurations. This is a drawback of learning-based 3D pose estimation methods that rely on a pose prior. In future work, we will study how images of rare poses could be synthesized to uniformly populate the space.

While we outperform the state of the art in 2D/3D human pose estimation in controlled environment, our 2D performance on real images is below the state of the art as indicated by our performance on the MPII test set re-

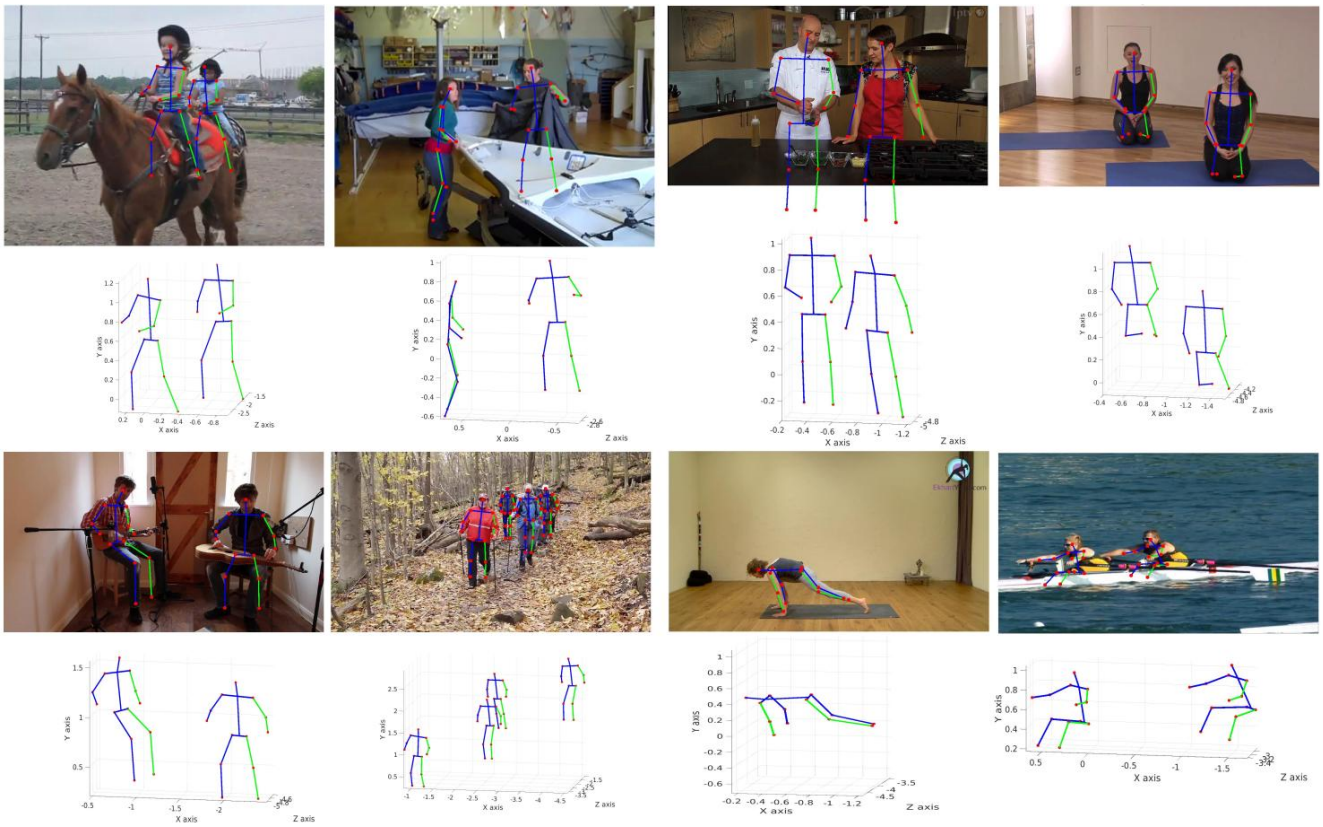

Figure 7. Qualitative examples on the MPII dataset. To visualize multiple 3D poses, which are expressed in a coordinate system centered on the torso, we find for each one of them the appropriate 3D displacements in front of the camera. This is obtained using a least square minimization of the reprojection error, defined as the distance between the 2D pose estimated by LCR-Net and the projection of the 3D pose in the image. When the camera is unknown, simply hypothesizing an orthographic camera leads to acceptable qualitative results.

Method	Human3.6M 2D pose error (pix)	MPII % of correct joints (PCKh@0.5)
Wei *et al*. [33]	10.04	**88.5**
LCR-Net + PPI	**8.50**	74.2

Table 4. 2D pose estimation results on Human3.6M and MPII test sets compared to state-of-the-art 2D method [33].

ported in Table 4. Note that in contrast to most other approaches, our holistic method also gives an estimation of the occluded joints that is not evaluated. Although globally correct (Fig. 7), our pose estimations can lack precision on the limb extremities resulting in lower PCKh score in 2D . One explanation is that we use a fully-connected layer for the regression. This could be improved by using fully convolutional architecture with deconvolution or upsampling [19].

Multi-person pose detection. For multi-person evaluation, our validation set contains 209 groups of multiple people in 187 images. We follow the standard protocol and evaluate AP averaged over joints. We obtain around 49% for a standard mAP@0.5 and near 60% for a mAP@1. Examples of multi-person pose detection are shown in Figure 7. Our method is able to detect multiple people even if they overlap (second row, second column). It is also robust to unusual poses (top right), truncation (top row, third column) or important occlusions (top row, second column).

5. Conclusion

This paper introduces a Localization-Classification-Regression network (LCR-Net) for joint 2D and 3D human pose detection in natural images. We demonstrate the benefit of an end-to-end architecture which relies on pose proposals that are hypothesized at different locations in the image, classified and refined by regression. The final pose estimation is obtained by integrating over neighboring pose hypotheses. We outperform the state of the art in 3D pose estimation on Human3.6M, *i.e.*, a controlled environment and show promising results on real images. Future work includes adding rare poses using synthetic training data and a fully convolutional architecture with deconvolution.

Acknowledgements. This work was supported by ERC advanced grant Allegro and an Amazon gift. We thank NVIDIA for donating the GPUs used for this research.

References

[1] I. Akhter and M. Black. Pose-conditioned joint angle limits for 3D human pose reconstruction. In *CVPR*, 2015. 3, 7

[2] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2D human pose estimation: New benchmark and state-of- the-art analysis. In *CVPR*, 2014. 5, 7

[3] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *ECCV*, 2016. 3, 5

[4] A. Bulat and G. Tzimiropoulos. Human pose estimation via convolutional part heatmap regression. In *ECCV*, 2016. 1, 2

[5] W. Chen, H. Wang, Y. Li, H. Su, Z. Wang, C. Tu, D. Lischinski, D. Cohen-Or, and B. Chen. Synthesizing training images for boosting human 3D pose estimation. In *3DV*, 2016. 1, 3

[6] X. Chen and A. L. Yuille. Articulated pose estimation by a graphical model with image dependent pairwise relations. In *NIPS*, 2014. 2

[7] Y. Du, Y. Wong, Y. Liu, F. Han, Y. Gui, Z. Wang, M. Kankanhalli, and W. Geng. Marker-less 3D human motion capture with monocular image sequence and height-maps. In *ECCV*, 2016. 5, 6

[8] X. Fan, K. Zheng, Y. Lin, and S. Wang. Combining local appearance and holistic view: Dual-source deep neural networks for human pose estimation. In *CVPR*, 2015. 2

[9] X. Fan, K. Zheng, Y. Zhou, and S. Wang. Pose locality constrained representation for 3D human pose reconstruction. In *ECCV*, 2014. 3

[10] G. Gkioxari, B. Hariharan, R. B. Girshick, and J. Malik. R-CNNs for pose estimation and action detection. *arXiv*, 2014. 2

[11] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele. Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In *ECCV*, 2016. 2

[12] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu. Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments. *IEEE Trans. PAMI*, 2014. 5

[13] I. Kostrikov and J. Gall. Depth sweep regression forests for estimating 3D human pose from images. In *BMVC*, 2014. 5

[14] S. Li and A. B. Chan. 3D human pose estimation from monocular images with deep convolutional neural network. In *ACCV*, 2014. 2, 5, 6

[15] S. Li, W. Zhang, and A. B. Chan. Maximum-margin structured learning with deep networks for 3D human pose estimation. In *ICCV*, 2015. 2, 3, 5, 6

[16] S. Li, W. Zhang, and A. B. Chan. Maximum-margin structured learning with deep networks for 3D human pose estimation. *IJCV*, 2016. 5, 6

[17] T. Lin, M. Maire, S. J. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: common objects in context. In *ECCV*, 2014. 7

[18] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber. Documentation MoCap database HDM05. Technical Report CG-2007-2, Universität Bonn, 2007. 7

[19] A. Newell, K. Yang, and J. Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, 2016. 1, 2, 8

[20] W. Ouyang, X. Chu, and X. Wang. Multi-source deep learning for human pose estimation. In *CVPR*, 2014. 2

[21] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, and B. Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. *CVPR*, 2016. 2, 7

[22] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 1, 2, 3, 4

[23] G. Rogez and C. Schmid. MoCap-guided data augmentation for 3D pose estimation in the wild. In *NIPS*, 2016. 1, 2, 3, 5, 6, 7

[24] M. Sanzari, V. Ntouskos, and F. Pirri. Bayesian image based 3D pose estimation. In *ECCV*, 2016. 5, 6

[25] E. Simo-Serra, A. Quattoni, C. Torras, and F. Moreno-Noguer. A joint model for 2D and 3D pose estimation from a single image. In *CVPR*, 2013. 3

[26] 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. 3

[27] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 4

[28] B. Tekin, I. Katircioglu, M. Salzmann, V. Lepetit, and P. Fua. Structured prediction of 3D human pose with deep neural networks. In *BMVC*, 2016. 5, 6

[29] B. Tekin, A. Rozantsev, V. Lepetit, and P. Fua. Direct prediction of 3D body poses from motion compensated sequences. In *CVPR*, 2016. 3, 5, 6

[30] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *NIPS*, 2014. 2

[31] A. Toshev and C. Szegedy. DeepPose: Human pose estimation via deep neural networks. In *CVPR*, 2014. 2

[32] C. Wang, Y. Wang, Z. Lin, A. L. Yuille, and W. Gao. Robust estimation of 3D human poses from a single image. In *CVPR*, 2014. 3

[33] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *CVPR*, 2016. 8

[34] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, 2011. 5

[35] H. Yasin, U. Iqbal, B. Krüger, A. Weber, and J. Gall. A dual-source approach for 3D pose estimation from a single image. In *CVPR*, 2016. 2, 3, 5, 7

[36] F. Zhou and F. D. la Torre. Spatio-temporal matching for human detection in video. In *ECCV*, 2014. 3

[37] X. Zhou, M. Zhu, S. Leonardos, K. Derpanis, and K. Daniilidis. Sparseness meets deepness: 3D human pose estimation from monocular video. In *CVPR*, 2016. 3, 5, 6