DeeperCut: A Deeper, Stronger, and Faster Multi-Person Pose Estimation Model

Eldar Insafutdinov\textsuperscript{1}, Leonid Pishchulin\textsuperscript{1}, Bjoern Andres\textsuperscript{1}, Mykhaylo Andriluka\textsuperscript{2}, and Bernt Schiele\textsuperscript{1}

\textsuperscript{1}Max Planck Institute for Informatics, Germany, \textsuperscript{2}Stanford University, USA

Abstract. The goal of this paper is to advance the state-of-the-art of articulated pose estimation in scenes with multiple people. To that end we contribute on three fronts. We propose (1) improved body part detectors that generate effective bottom-up proposals for body parts; (2) novel image-conditioned pairwise terms that allow to assemble the proposals into a variable number of consistent body part configurations; and (3) an incremental optimization strategy that explores the search space more efficiently thus leading both to better performance and significant speed-up factors. We evaluate our approach on two single-person and two multi-person pose estimation benchmarks. The proposed approach significantly outperforms best known multi-person pose estimation results while demonstrating competitive performance on the task of single person pose estimation.\footnote{Models and code available at http://pose.mpi-inf.mpg.de}

1 Introduction

Human pose estimation has recently made dramatic progress in particular on standard benchmarks for single person pose estimation \cite{1,2}. This progress has been facilitated by the use of deep learning-based architectures \cite{3,4} and by the availability of large-scale datasets such as “MPII Human Pose” \cite{2}. In order to make further progress on the challenging task of multi-person pose estimation we carefully design and evaluate several key-ingredients for human pose estimation.

The first ingredient we consider is the generation of body part hypotheses. Essentially all prominent pose estimation methods include a component that detects body parts or estimates their position. While early work used classifiers such as SVMs and AdaBoost \cite{1,5,6,7}, modern approaches build on different flavors of deep learning-based architectures \cite{8,9,10,11}. The second key ingredient are pairwise terms between body part hypotheses that help grouping those into valid human pose configurations. In earlier models such pairwise terms were essential for good performance \cite{1,5,6}. Recent methods seem to profit less from such pairwise terms due to stronger unaries \cite{8,10,11}. Image-conditioned pairwise terms \cite{7,9} however have the promise to allow for better grouping. Last but not
least, inference time is always a key consideration for pose estimation models. Often, model complexity has to be treated for speed and thus many models do not consider all spatial relations that would be beneficial for best performance.

In this paper we contribute to all three aspects and thereby significantly push the state of the art in multi-person pose estimation. We use a general optimization framework introduced in [10] as a test bed for all three key ingredients proposed in this paper, as it allows to easily replace and combine different components. Our contributions are three-fold, leading to a novel multi-person pose estimation approach that is deeper, stronger, and faster compared to the state of the art [10]:

- “deeper”: we propose strong body part detectors based on recent advances in deep learning [12] that – taken alone – already allow to obtain competitive performance on pose estimation benchmarks.
- “stronger”: we introduce novel image-conditioned pairwise terms between body parts that allow to push performance in the challenging case of multi-people pose estimation.
- “faster”: we demonstrate that using our image-conditioned pairwise along with very good part detection candidates in a fully-connected model dramatically reduces the run-time by 2–3 orders of magnitude. Finally, we introduce a novel incremental optimization method to achieve a further 4x run-time reduction while improving human pose estimation accuracy.

We evaluate our approach on two single-person and two multi-person pose estimation benchmarks and report the best results in each case. Sample multi-person pose estimation predictions by the proposed approach are shown in Fig. 1.

---

2 Implementation of DeepCut [10] was kindly provided by the authors.
**Related work.** Articulated human pose estimation has been traditionally formulated as a structured prediction task that requires an inference step combining local observations of body joints with spatial constraints. Various formulations have been proposed based on tree \cite{13,14,15,6} and non-tree models \cite{16,17}. The goal of the inference process has been to refine observations from local part detectors into coherent estimates of body configurations. Models of this type have been increasingly superseded by strong body part detectors \cite{18,19}, which has been reinforced by the development of strong image representations based on convolutional networks. Recent work aimed to incorporate convolutional detectors into part-based models \cite{9} or design stronger detectors by combining the detector output with location-based features \cite{20}.

Specifically, as suggested by \cite{10}, in the presence of strong detectors spatial reasoning results in diminishing returns because most contextual information can be incorporated directly in the detector. In this work we elevate the task to a new level of complexity by addressing images with multiple potentially overlapping people. This results in a more complex structured prediction problem with a variable number of outputs. In this setting we observe a large boost from conducting inference on top of state-of-the-art part detectors.

Combining spatial models with convnets allows to increase the receptive field that is used for inferring body joint locations. For example \cite{11} iteratively trains a cascade of convolutional parts detectors, each detector taking the scoremap of all parts from the previous stage. This effectively increases the depth of the network and the receptive field is comparable to the entire person. With the recent developments in object detection newer architectures are composed of a large number of layers and the receptive field is large automatically. In this paper, we introduce a detector based on the recently proposed deep residual networks \cite{12}. This allows us to train a detector with a large receptive field \cite{11} and to incorporate intermediate supervision.

The use of purely geometric pairwise terms is suboptimal as they do not take local image evidence into account and only penalize deviation from the expected joint location. Due to the inherent articulation of body parts the expected location can only approximately guide the inference. While this can be sufficient when people are relatively distant from each other, for closely positioned people more discriminative pairwise costs are essential. Two prior works \cite{7,9} have introduced image-dependent pairwise terms between connected body parts. While \cite{7} uses an intermediate representation based on poselets our pairwise terms are conditioned directly on the image. \cite{9} clusters relative positions of adjacent joints into $T = 11$ clusters, and assigns different labels to the part depending on which cluster it falls to. Subsequently a CNN is trained to predict this extended set of classes and later an SVM is used to select the maximum scoring joint pair relation.

Single person pose estimation has advanced considerably, but the setting is simplified. Here we focus on the more challenging problem of multi-person pose estimation. Previous work has addressed this problem as sequence of person detection and pose estimation \cite{21,22,23}. \cite{21} use a detector for initialization and reasoning across people, but rely on simple geometric body part relationships
and only reason about person-person occlusions. [23] focus on single partially
occluded people, and handle multi-person scenes akin to [6]. [10] propose to
jointly detect and estimate configurations, but rely on simple pairwise terms only,
which limits the performance and, as we show, results in prohibitive inference
time to fully explore the search space. Here, we innovate on multiple fronts both
in terms of speed and accuracy.

2 DeepCut Recap

This section summarizes DeepCut [10] and how unary and pairwise terms are used
in this approach. DeepCut is a state-of-the-art approach to multi-person pose
estimation based on integer linear programming (ILP) that jointly estimates poses
of all people present in an image by minimizing a joint objective. This objective
aims to jointly partition and label an initial pool of body part candidates into
consistent sets of body-part configurations corresponding to distinct people. We
use DeepCut as a general optimization framework that allows to easily replace
and combine different components.

Specifically, DeepCut starts from a set $D$ of body part candidates, i.e. putative
detections of body parts in a given image, and a set $C$ of body part classes, e.g.,
head, shoulder, knee. The set $D$ of part candidates is typically generated by body
part detectors and each candidate $d \in D$ has a unary score for every body part
class $c \in C$. Based on these unary scores DeepCut associates a cost or reward
$\alpha_{dc} \in \mathbb{R}$ to be paid by all feasible solutions of the pose estimation problem for
which the body part candidate $d$ is a body part of class $c$.

Additionally, for every pair of distinct body part candidates $d, d' \in D$ and
every two body part classes $c, c' \in C$, the pairwise term is used to generate a cost
or reward $\beta_{dd'cc'} \in \mathbb{R}$ to be paid by all feasible solutions of the pose estimation
problem for which the body part $d$, classified as $c$, and the body part $d'$, classified
as $c'$, belong to the same person.

With respect to these sets and costs, the pose estimation problem is cast
as an ILP in two classes of 01-variables: Variables $x : D \times C \rightarrow \{0, 1\}$ indicate
by $x_{dc} = 1$ that body part candidate $d$ is of body part class $c$. If, for a $d \in D$
and all $c \in C$, $x_{dc} = 0$, the body part candidate $d$ is suppressed. Variables
$y : (\binom{D}{2}) \rightarrow \{0, 1\}$ indicate by $y_{dd'} = 1$ that body part candidates $d$ and $d'$
belong to the same person. Additional variables and constraints described in [10] link the
variables $x$ and $y$ to the costs and ensure that feasible solutions $(x, y)$ well-define
a selection and classification of body part candidates as body part classes as well
as a clustering of body part candidates into distinct people.

The DeepCut ILP is hard and hard to approximate, as it generalizes the
minimum cost multicut or correlation clustering problem which is APX-hard
[24,25]. Using the branch-and-cut algorithm proposed in [10] to compute constant-
factor approximative feasible solutions of instances of the DeepCut ILP is not
necessarily practical. In Sec. 5 we propose an incremental optimization approach
that uses branch-and-cut algorithm to incrementally solve several instances of
ILP, which results into 4–5x run-time reduction with increased pose estimation accuracy.

3 Part Detectors

As argued before, strong part detectors are an essential ingredient of modern pose estimation methods. In this section we propose and evaluate a deep fully-convolutional human body part detection model drawing on powerful recent ideas from semantic segmentation, object classification \cite{26,27,12} and human pose estimation \cite{19,10,11}.

3.1 Model

Architecture. We build on the recent advances in object classification and adapt the extremely deep Residual Network (ResNet) \cite{12} for human body part detection. This model achieved excellent results on the recent ImageNet Object Classification Challenge and specifically tackles the problem of vanishing gradients by passing the state through identity layers and modeling residual functions. Our best performing body part detection model has 152 layers (c.f. Sec. 3.2) which is in line with the findings of \cite{12}.

Stride. Adapting ResNet for the sliding window-based body part detection is not straight forward: converting ResNet to the fully convolutional mode leads to a 32 px stride which is too coarse for precise part localization. \cite{10} showed that using a stride of 8 px leads to good part detection results. Typically, spatial resolution can be recovered by either introducing up-sampling deconvolutional layers \cite{26}, or blowing up the convolutional filters using the hole algorithm \cite{27}. The latter has shown to perform better on the task of semantic segmentation. However, using the hole algorithm to recover the spatial resolution of ResNet is infeasible due to memory constraints. For instance, the 22 residual blocks in the conv4 bank of ResNet-101 constitute the major part of the network and running it at stride 8 px does not fit the net into GPU memory\footnote{We use NVIDIA Tesla K40 GPU with 12 GB RAM}. We thus employ a hybrid approach. First, we remove the final classification as well as average pooling layer. Then, we decrease the stride of the first convolutional layers of the conv5 bank from 2 px to 1 px to prevent down-sampling. Next, we add holes to all 3x3 convolutions in conv5 to preserve their receptive field. This reduces the stride of the full CNN to 16 px. Finally, we add deconvolutional layers for 2x up-sampling and connect the final output to the output of the conv3 bank.

Receptive field size. A large receptive field size allows to incorporate context when predicting locations of individual body parts. \cite{8,11} argue about the importance of large receptive fields and propose a complex hierarchical architecture predicting parts at multiple resolution levels. The extreme depth of ResNet allows for a very large receptive field (on the order of 1000 px compared to VGG’s 400 px \cite{28}) without the need of introducing complex hierarchical architectures. We
empirically find that re-scaling the original image such that an upright standing person is 340 px high leads to best performance.

**Intermediate supervision.** It has been argued that providing additional supervision addresses the problem of vanishing gradients in deep neural networks \[29,30,11\]. In addition to that, \[11\] reports that using part scoremaps produced at intermediate stages as inputs for subsequent stages helps to encode spatial relations between parts. ResNets address the first problem by introducing identity connections and learning residual functions. To address the second concern, we make a slightly different choice: we add part loss layers inside the conv4 bank of ResNet. We argue that it is not strictly necessary to use part scoremaps as inputs for the subsequent stages. The activations from such intermediate part predictions are different only up to a linear transformation and contain all information about part presence that is available at that stage of the network. In Sec. 3.2 we empirically show a consistent improvement of part detection performance when including intermediate supervision.

**Loss functions.** We use sigmoid activations for the final predictions and cross entropy loss function during training \[10\]. Similarly to \[10,31\] we perform location refinement by predicting offsets from the locations on the scoremap grid to the ground truth joint locations.

**Training.** We use the publicly available ResNet implementation (Caffe) and initialize from the ImageNet-pre-trained models. We train networks with SGD for 1M iterations, starting with the learning rate $\text{lr}=0.001$ for 10k, then $\text{lr}=0.002$ for 420k, $\text{lr}=0.0002$ for 300k and $\text{lr}=0.0001$ for 300k. This corresponds to roughly 17 epochs of the combined MPII \[2\] and LSPET \[15\] datasets. We found enabling batch normalization \[32\] to worsen performance, as the batch size of 1 in fully convolutional training is not enough to provide a reliable estimate of activation statistics. During training we switch off collection of statistics and use the mean and variance that were gathered on the ImageNet dataset.

### 3.2 Evaluation of Part Detectors

**Datasets.** We use three public datasets: “Leeds Sports Poses” (LSP) \[1\] (person-centric (PC) annotations) including 1000 training and 1000 test images of people doing sports; “LSP Extended” (LSPET) \[15\] consisting of 10000 training images; “MPII Human Pose” (“Single Person”) \[2\] consisting of 19185 training and 7247 testing people in every day activities. We train part detectors on the union of MPII, LSPET and LSP training sets to be directly comparable to \[10\] and evaluate on MPII Single Person and LSP testing sets.

**Evaluation measures.** We use the standard “Percentage of Correct Keypoints (PCK)” evaluation metric \[33,34,8\]. We use the evaluation scripts available from the web page of \[2\] and are thus directly comparable to other methods. In addition to PCK at fixed threshold, we report “Area under Curve” (AUC) computed for the entire range of PCK thresholds.

**Results on LSP.** The results are shown in Tab. \[1\] ResNet-50 with 8 px stride achieves 87.8% PCK and 63.7% AUC. Increasing the stride size to 16 px and
A Deeper, Stronger, and Faster Multi-Person Pose Estimation Model

| Setting                                      | Head | Sho | Elb | Wri | Hip | Knee | Ank | PCK | AUC |
|----------------------------------------------|------|-----|-----|-----|-----|------|-----|-----|-----|
| ResNet-50 (8 px)                             | 96.9 | 90.3| 85.0| 81.5| 88.6| 87.3 | 84.8| 87.8| 63.7|
| ResNet-50 (16 px + 2x up-sample)             | 96.7 | 89.8| 84.6| 80.4| 89.3| 86.4 | 82.8| 87.2| 63.1|
| ResNet-101 (16 px + 2x up-sample)            | 96.9 | 91.2| 85.8| 82.6| 90.2| 85.9 | 89.1| 64.6|     |
| ResNet-152 (16 px + 2x up-sample)            | 97.4 | 91.7| 85.7| 82.4| 90.1| 89.2 | 86.9| 89.1| 65.1|
| + intermediate supervision                  | 97.4 | 92.7| 87.5| 84.4| 91.5| 89.9 | 87.2| 90.1| 66.1|
| DeepCut [10]                                 | 97.0 | 91.0| 83.8| 78.1| 91.0| 86.7 | 82.0| 87.1| 63.5|
| Wei et al. [11]                              | 97.8 | 92.5| 87.0| 83.9| 91.5| 90.8 | 89.9| 90.5| 65.4|
| Tompson et al. [8]                           | 90.6 | 79.2| 67.9| 63.4| 69.5| 71.0 | 64.2| 72.3| 47.3|
| Chen & Yuille [9]                             | 91.8 | 78.2| 71.8| 65.5| 73.3| 70.2 | 63.4| 73.4| 40.1|
| Fan et al. [35]                              | 92.4 | 75.2| 65.3| 64.0| 75.7| 68.3 | 70.4| 73.0| 43.2|

Table 1. Pose estimation results (PCK) on LSP (PC) dataset.

| Setting                                      | Head | Sho | Elb | Wri | Hip | Knee | Ank | PCK₉ | AUC  |
|----------------------------------------------|------|-----|-----|-----|-----|------|-----|------|------|
| ResNet-152                                   | 96.3 | 94.1| 88.6| 83.9| 87.2| 82.9 | 77.8| 87.8 | 60.0 |
| + intermediate supervision                   | 96.6 | 94.6| 88.5| 84.4| 87.6| 83.9 | 79.4| 88.3 | 60.7 |
| DeepCut [10]                                 | 94.1 | 90.2| 83.4| 77.3| 82.6| 75.7 | 68.6| 82.4 | 56.5 |
| Tompson et al. [8]                           | 95.8 | 90.3| 80.5| 74.3| 77.6| 69.7 | 62.8| 79.6 | 51.8 |
| Carreira et al. [36]                         | 95.7 | 91.7| 81.7| 72.4| 82.8| 73.2 | 66.4| 81.3 | 49.1 |
| Tompson et al. [19]                          | 96.1 | 91.9| 83.9| 77.8| 80.9| 72.3 | 64.8| 82.0 | 54.9 |
| Wei et al. [11]                              | 97.8 | 95.0| 88.7| 84.0| 88.4| 82.8 | 79.4| 88.5 | 61.4 |

Table 2. Pose estimation results (PCK₉) on MPII Single Person.

up-sampling the scoremaps by 2x to compensate for the loss on resolution slightly drops the performance to 87.2% PCK. This is expected as up-sampling cannot fully compensate for the loss of information due to a larger stride. Larger stride minimizes memory requirements, which allows for training a deeper ResNet-152. The latter significantly increases the performance (89.1 vs. 87.2% PCK, 65.1 vs. 63.1% AUC), as it has larger model capacity. Introducing intermediate supervision further improves the performance to 90.1% PCK and 66.1% AUC, as it constrains the network to learn useful representations in the early stages and uses them in later stages for spatial disambiguation of parts.

The results are compared to the state of the art in Tab. 1. Our best model significantly outperforms the best known result by DeepCut [10] (90.1% PCK vs. 87.1% PCK). This clearly shows the advantages of using deeper models for body part detection. Our model performs on par with the recent approach of Wei et al. [11] (90.1 vs. 90.5% PCK, 66.1 vs. 65.4% AUC). This is interesting, as they use a much more complex multi-scale multi-stage architecture.

**Results on MPII Single Person.** The results are shown in Tab. 2. ResNet-152 achieves 87.8% PCK₉ and 60.0% AUC while intermediate supervision slightly improves the performance further to 88.3% PCK₉ and 60.7% AUC. The fact that we use the same trained model on both LSP and MPII benchmarks and achieve roughly the same performance demonstrates the generality of the proposed approach. Comparing the obtained results to the state of the art we observe significant improvement over DeepCut [10] (+5.9% PCK₉, +4.2% AUC), which again underlines the importance of using extremely deep model. The proposed approach performs on par with the best know result by Wei et al. [11] (88.3 vs. 88.5% PCK₉) for the maximum distance threshold, while slightly losing when using the entire range of thresholds (60.7 vs. 61.4% AUC). We envision that
Fig. 2. Visualizations of regression predictions. Top: from left shoulder to the right shoulder (green), right hip (red), left elbow (light blue), right ankle (purple) and top of the head (dark blue). Bottom: from right knee to the right hip (green), right ankle (red), left knee (dark blue), left ankle (light blue) and top of the head (purple). Longer-range predictions, such as e.g. shoulder – ankle may be less accurate for harder poses (top row, images 2 and 3) compared to the nearby predictions. However, they provide enough information to constrain the search space in the fully-connected spatial model.

Fig. 3. Visualization of features extracted to score the pairwise. See text for details.

extending the proposed approach to incorporate multiple scales as in [11] should improve the performance.

4 Image-Conditioned Pairwise Terms

As discussed in Sec. 3, a large receptive field for the CNN-based part detectors allows to accurately predict the presence of a body part at a given location. However, it also contains enough evidence to reason about locations of other parts in the vicinity. We draw on this insight and propose to also use deep networks to make pairwise part-to-part predictions. They are subsequently used to compute the pairwise probabilities and show significant improvements for multi-person pose estimation.

4.1 Model

Our approach is inspired by the body part location refinement described in Sec. 3. In addition to predicting offsets for the current joint, we directly regress from the
A naive solution would be to replace a detection with \( n \) which allows to distribute the detection candidates in a better way. We employ a simple trick by splitting a detection \( d \) into several if \( k = (x_k, y_k) \) that is marked positive w.r.t. the joint \( c \in C \) and for each remaining joint \( c' \in C \setminus c \), we define a relative position of \( c' \) w.r.t. \( c \) as a tuple \( t_{cc'}^k = (x_{c'} - x_k, y_{c'} - y_k) \). We add an extra layer that predicts relative position \( o_{cc'}^k \) and train it with a smooth L1 loss function. We thus perform joint training of body part detectors (cross-entropy loss), location regression (L1 loss) and pairwise regression (L1 loss) by linearly combining all three loss functions. The targets \( t \) are normalized to have zero mean and unit variance over the training set. Results of such predictions are shown in Fig. 2.

We then use these predictions to compute pairwise costs \( \beta_{dd'cc'} \). For any pair of detections \((d, d')\) (Fig. 3) and for any pair of joints \((c, c')\) we define the following quantities: locations \( l_d, l_d' \) of detections \( d \) and \( d' \) respectively; the offset prediction \( o_{cc'}^d \) from \( c \) to \( c' \) at location \( d \) (solid red) coming from the CNN and similarly the offset prediction \( o_{c'c}^d \) (solid turquoise). We then compute the offset between the two predictions: \( \hat{o}_{dd} = l_{d'} - l_d \) (marked in dashed red). The degree to which the prediction \( o_{cc'}^d \) agrees with the actual offset \( \hat{o}_{dd'} \) tells how likely \( d, d' \) are of classes \( c, c' \) respectively and belong to the same person. We measure this by computing the distance between the two offsets \( \Delta_f = \| \hat{o}_{dd'} - o_{cc'}^d \|_2 \), and the absolute angle \( \theta_f = |\angle(\hat{o}_{dd'}, o_{cc'}^d)| \) where \( f \) stands for forward direction, i.e. from \( d \) to \( d' \). Similarly, we incorporate the prediction \( o_{c'c}^d \) in the backwards direction by computing \( \Delta_b = \| \hat{o}_{dd'} - o_{c'c}^d \|_2 \) and \( \theta_b = |\angle(\hat{o}_{dd'}, o_{c'c}^d)| \). Finally, we define a feature vector by augmenting features with exponential terms: \( f_{dd'cc'} = (\Delta_f, \theta_f, \Delta_b, \theta_b, \exp(-\Delta_f), \ldots, \exp(-\theta_b)) \).

We then use the features \( f_{dd'cc'} \) and define logistic model:

\[
p(z_{dd'cc'} = 1 | f_{dd'cc'}, \omega_{cc'}) = \frac{1}{1 + \exp(-\langle \omega_{cc'}, f_{dd'cc'} \rangle)}.
\]

where \( K = (|C| \times (|C| + 1))/2 \) parameters \( \omega_{cc'} \) are estimated using ML.

### 4.2 Sampling Detections

**Location refinement NMS.** DeepCut samples the set of detections \( D \) from the scoremap by applying non-maximum suppression (NMS). Here, we utilize location refinement and correct grid locations with the predicted offsets before applying NMS. This pulls detections that belong to a particular body joint towards its true location thereby increasing the density of detections around that location, which allows to distribute the detection candidates in a better way.

**Splitting of part detections.** DeepCut ILP solves the clustering problem by labeling each detection \( d \) with a single part class \( c \) and assigning it to a particular cluster that corresponds to a distinct person. However, it can happen that the same spatial location is occupied by more than one body joint, and therefore, its corresponding detection can only be labeled with one of the respecting classes. A naive solution would be to replace a detection with \( n \) detections for each part class, which would result though in a prohibitive increase in the number of detections. We employ a simple trick by splitting a detection \( d \) into several if
more than one part has unary probability that is higher than a chosen threshold $s$ (in our case $s = 0.4$).

### 4.3 Evaluation of Pairwise

**Datasets and evaluation measure.** We evaluate on the challenging public “MPII Human Pose” (“Multi-Person”) benchmark [2] consisting of 3844 training and 1758 testing groups of multiple overlapping people in highly articulated poses with a variable number of parts. We perform all intermediate experiments on a validation set of 200 images sampled uniformly at random from the training set and refer to it as MPII Multi-Person Val. In addition, we report the major results on the full testing set, as well as on the subset of 288 images for the direct comparison to [10]. The AP evaluation measure [10] evaluating consistent body part detections is used for performance comparison. Additionally, we report median running time measured in seconds.

**Evaluation of unaries and pairwise.** The results are shown in Tab. 3. Baseline DeepCut achieves 33.3% AP. Using the proposed pairwise significantly improves performance achieving 47.7% AP. This clearly shows the advantages of using image-conditioned pairwise to disambiguate the body part assignment for multiple overlapping individuals. Remarkably, the proposed pairwise dramatically reduce the run-time by two orders of magnitude (1987 vs. 259220 s). This underlines the argument that using strong pairwise in the fully-connected model allows to significantly speed-up the inference. Using additionally the proposed part detectors further boosts the performance (52.3 vs. 47.7% AP), which can be attributed to better quality part hypotheses. Run-time is again almost halved, which clearly shows the importance of obtaining high-quality part detection candidates for more accurate and faster inference. Performing location refinement before NMS slightly improves the performance, but also reduces the run-time by

---

4 Evaluation is performed by the authors of [2], as testing ground truth is not public.
5 Run-time is measured on a single core Intel Xeon 2.70GHz
2x: this allows to increase the density of detections at the most probable body part locations and thus suppresses more detections around the most confident ones, which leads to better distribution of part detection candidates and reduces confusion generated by the near-by detections. Overall, we observe significant performance improvement and dramatic reduction in run-time by the proposed DeeperCut compared to the baseline DeepCut.

Ablation study of pairwise. An ablation study of the proposed image-conditioned pairwise is performed in Tab. 4. Regressing from both joints onto the opposite joint’s location and including angles achieves the best performance of 52.6% AP and the minimum run-time of 578 s. Regressing from a single joint only slightly reduces the performance to 51.1% AP, but significantly increases run-time by 4x: these pairwise are less robust compared to the bi-directional, which confuses the inference. Removing the angles from the pairwise features also decreases the performance (51.3 vs. 52.6% AP) and doubles run-time, as it takes out the information about body part orientation.

5 Incremental Optimization

Solving one instance of the DeepCut ILP for all body part candidates detected for an image, as suggested in [10] and summarized in Sec. 2 is elegant in theory but disadvantageous in practice:

Firstly, the time it takes to compute constant-factor approximative feasible solution by the branch-and-cut algorithm proposed in [10] can be exponential in the number of body part candidates in the worst case. In practice, this limits the number of body part candidates that can be processed by this algorithm. Due to this limitation, it does happen that body parts and, for images showing many persons, entire persons are missed, simply because they are not contained in the set of candidates.

Secondly, solving one instance of the optimization problem for the entire image means that no distinction is made between body part classes that are detected reliably, e.g. head and shoulders, and body part classes that are detected less reliably, e.g. wrists, elbows and ankles. Therefore, it does happen that unreliable detections corrupt the solution.

To address both problems, we solve not one instance of the DeepCut ILP but several, starting with only those body part classes that are detected most reliably and only then considering body part classes that are detected less reliably. Concretely, we study two variants of this incremental optimization approach which are defined in Tab. 5. Specifically, the procedure works as follows:

For each subset of body part classes defined in Tab. 5, an instance of the DeepCut ILP is set up and a constant-factor approximative feasible solution computed using the branch-and-cut algorithm. This feasible solution selects, labels and clusters a subset of body part candidates, namely of those body part classes that are considered in this instance. For the next instance, each cluster of body part candidates of the same class from the previous instance becomes just one body part candidate whose class is fixed. Thus, the next instance is an optimization problem for selecting, labeling and clustering body parts that have
Table 5. As the run-time of the DeepCut branch-and-cut algorithm limits the number of body part candidates that can be processed in practice, we split the set of body part classes into subsets, coarsely and finely, as shown above, and solve the pose estimation problem incrementally.

| Setting                  | Head | Sho | Elb | Wri | Hip | Knee | Ank | UBody | FBody | time (s) |
|--------------------------|------|-----|-----|-----|-----|------|-----|-------|-------|----------|
| 1-stage optimize, 100 det, nms 1x | 70.3 | 61.6 | 52.1 | 43.7 | 50.6 | 47.0 | 40.6 | 56.9   | 52.6   | 578      |
| 1-stage optimize, 100 det, nms 2x | 71.3 | 64.1 | 55.8 | 44.1 | 53.8 | 48.7 | 41.3 | 58.8   | 54.5   | 596      |
| 1-stage optimize, 150 det, nms 2x | 74.1 | 65.6 | 56.0 | 44.3 | 54.4 | 49.2 | 39.8 | 60.0   | 55.1   | 1041     |
| 2-stage optimize         | 75.9 | 66.8 | 58.8 | 46.1 | 54.1 | 48.7 | 42.4 | 61.9   | 56.5   | 483      |
| 3-stage optimize         | 78.3 | 69.3 | 58.4 | 47.5 | 55.1 | 49.6 | 42.5 | 63.4   | 57.6   | 271      |
| + split detections        | 78.5 | 70.5 | 59.7 | 48.7 | 55.4 | 50.6 | 44.4 | 64.3   | 58.7   | 270      |

DeepCut [10] 50.1 44.1 33.5 26.5 33.0 28.5 14.4 38.6 33.3 259220

Table 6. Performance (AP) of different hierarchical versions of DeeperCut on MPII Multi-Person Val.

not been determined by previous instances. Overall, this allows us to start with more body part candidates consistently and thus improve the pose estimation result significantly.

5.1 Evaluation of Incremental Optimization

Results are shown in Tab. 6. Single stage optimization with |D| = 100 part detection candidates achieves 52.6% AP. (best from Tab. 3). Performing more aggressive NMS with radius of 24 px improves the performance (54.5 vs. 52.6% AP), as it allows to better distribute part detection candidates. Increasing |D| to 150 slightly improves the performance by +0.6% AP, but significantly increases run-time (1041 vs. 596 s). We found |D| = 150 to be maximum total number of detection candidates (11 per part) for which optimization runs in a reasonable time. Using incremental optimization of 2-stage inference slightly improves the performance (56.5 vs. 55.1% AP) as it allows for a larger number of detection candidates per body part (20) and leverages typically more confident predictions of the upper body parts in the first stage before solving for the entire body. Most importantly, it halves the median run-time from 1041 s to 483 s. Incremental optimization of 3-stage inference again almost halves the run-time to 271 s while noticeably improving the human pose estimation performance for all body parts but elbows achieving 57.6% AP. These results clearly demonstrate the advantages of the proposed incremental optimization. Splitting the detection candidates that simultaneously belong to multiple body parts with high confidence slightly improves the performance to 58.7% AP. This helps to overcome the limitation that each detection candidate can be assigned to a single body part and improves on cases where two body parts overlap thus sharing the same detection candidate.
We also compare the obtained results to DeepCut in Tab. 6 (last row). The proposed DeeperCut outperforms baseline DeepCut (58.7 vs. 33.3% AP) by almost doubling the performance, while run-time is reduced dramatically by 3 orders of magnitude from the infeasible 259220 s to affordable 270 s. This comparison clearly demonstrates the power of the proposed approach and dramatic effects of better unary, pairwise and optimization on the overall pose estimation performance and run-time.

5.2 Comparison to the State of the Art

We compare to the state of the art on MPII Multi-Person Test and “We Are Family” (WAF) [21] datasets.

Results on MPII Multi-Person. For direct comparison with DeepCut we evaluate on the same subset of 288 testing images as in [10]. Additionally, we provide the results on the entire testing set. Results are shown in Tab. 7. DeeperCut without incremental optimization already outperforms DeepCut by a large margin (66.2 vs. 54.1% AP). Using 3-stage incremental optimization further improves the performance to 69.7 improving by a dramatic 16.5% AP over the baseline. Remarkably, the run-time is reduced from 57995 to 230 s, which is an improvement by two orders of magnitude. Both results underline the importance of strong image-conditioned pairwise terms and incremental optimization to maximize multi-person pose estimation performance at the reduced run-time. When evaluating on the full testing set, a similar trend is observed: 3-stage incremental optimization improves over a single stage optimization (59.4 vs. 54.7% AP). Overall, we observe that the performance on the entire testing set is over 10% AP lower compared to the subset and run-time is doubled. This implies that the subset of 288 images is easier compared to the full testing set. We envision that performance differences between DeeperCut and DeepCut on the entire set will be at least as large as when compared on the subset.
### Table 9. Pose estimation results (AP) on WAF dataset.

| Setting      | Head | Sho | Elb | Wri | UBody | time (s) |
|--------------|------|-----|-----|-----|-------|----------|
| DeeperCut    | 92.6 | 81.1| 75.7| 78.8| 82.0  | 13       |
| DeepCut [10] | 76.6 | 80.8| 73.7| 73.6| 76.2  | 22000    |
| Chen&Yuille [23] | 83.3 | 56.1| 46.3| 35.5| 55.3  | -        |

**Results on WAF.** Results using the official evaluation protocol [21] assuming mPCP and AOP evaluation measures and considering detection bounding boxes provided by [21] are shown in Tab. 8. DeeperCut achieves the best result improving over the state of the art DeepCut (86.3 vs. 84.7% mPCP, 88.1 vs. 86.5% AOP). Noticeable improvements are observed both for upper (+2.3% mPCP) and lower (+2.4% mPCP) arms. However, overall performance differences between DeeperCut and the baseline DeepCut are not as pronounced compared to MPII Multi-Person dataset. This is due to the fact that actual differences are washed out by the peculiarities of the mPCP evaluation measure: mPCP assumes that people are pre-detected and human pose estimation performance is evaluated only for people whose upper body detections match the ground truth. Thus, a pose estimation method is not penalized for generating multiple body pose predictions, since the only pose prediction is considered whose upper body bounding box best matches the ground truth. We thus re-evaluate the competing approaches [10,23] using the more realistic AP evaluation measure [6]. The results are shown in Tab. 9. DeeperCut significantly improves over DeepCut (82.0 vs. 76.2% AP). The largest boost in performance is achieved for head (+16.0% AP) and wrists (+5.2% AP). This is due to the fact that DeeperCut follows incremental optimization strategy by first solving for the most reliable body parts, such as head and shoulders, and then using the obtained solution to improve estimation of harder body parts, such as wrists. Most notably, run-time is dramatically reduced by 3 orders of magnitude from 22000 to 13 s. These results clearly show the advantages of the proposed approach when evaluated in the real-world detection setting. The proposed DeeperCut also outperforms [23] by a large margin. The performance difference is much more pronounced compared to using mPCP evaluation measure: in contrast to mPCP, AP penalizes multiple body pose predictions of the same person. We envision that better NMS strategies are likely to improve the AP performance of [23].

### 6 Conclusion

In this paper we significantly advanced the state of the art in articulated multi-person human pose estimation. To that end we carefully re-designed and thoroughly evaluated several key ingredients. First, drawing on the recent advances in deep learning we proposed strong extremely deep body part detectors that – taken alone – already allow to obtain state of the art performance on standard pose estimation benchmarks. Second, we introduce novel image-conditioned pairwise terms between body parts that allow to significantly push the performance in the
challenging case of multi-people pose estimation, and dramatically reduce the run-time of the inference in the fully-connected spatial model. Third, we introduced a novel incremental optimization strategy to further reduce the run-time and improve human pose estimation accuracy. Overall, the proposed improvements allowed to almost double the pose estimation accuracy in the challenging multi-person case while reducing the run-time by 3 orders of magnitude.

Appendix A Qualitative Evaluation on MPII Multi-Person Dataset

We perform qualitative analysis of the proposed DeeperCut approach on MPII Multi-Person dataset. First, we visualize novel image-conditioned pairwise terms. Then, we demonstrate successful and failure cases of the proposed approach.

A.1 Image-Conditioned Pairwise

In order to visually analyze the proposed pairwise terms for a particular body part $c$, we proceed as follows. First, we randomly select a person in the image. Next, for each body part $c' \neq c$ we fix location of $c'$ at its ground truth location (to separate the effects of possible misdetection) and predict the location of $c$ using the learned regressor. Then, we compute the pairwise probability $p(cc')$ for every possible location of $c$ in the image. Fig. 4, rows 1–5, shows the probability of $c=$right knee anywhere in the image given the fixed location of other body parts. It can be seen that individual pairwise scoremaps have shape of cone extending towards the correct location, but are visually quite fuzzy. In order to visualize the effects of the pairwise predictions interacting in the fully-connected spatial model, we combine individual scoremaps by multiplying them. Combined scoremap is shown in Fig. 4, row 6. Clearly, interplay of individual pairwise scoremaps produces a strong evidence for a single right knee of a selected individual. This is in contrast to multi-modal unary scoremaps (Fig. 4, row 7) that show a strong response of any knee in the image. This clearly shows that the proposed pairwise terms can be successfully used to filter out body part locations belonging to multiple people thus effectively disambiguating between individuals. Fig. 5 shows more examples of combined pairwise scoremaps for each body part of randomly selected individuals, and unary scoremaps. It can be seen that although combined pairwise scoremaps are more fuzzy compared to the unary scoremaps, they allow to filter out the body part detections of other individuals when predicting the pose of the person in question.

A.2 Examples of Successful and Failure Cases

In Fig. 6 we include additional examples of successful pose estimation results by DeeperCut. The proposed approach correctly resolves cases with only subset

\footnote{We intentionally simplify the notation compared to Eq. (1).}
Fig. 4. Visualizations of the pairwise probabilities when the right knee of the middle person is used as target and fixed locations of other body parts as source. See text for explanation.
Fig. 5. Visualizations of all unary scoremaps (rows 2, 4 and 6) and all combined pairwise scoremaps (rows 1, 3 and 5). Pairwise scoremaps are visualized for a randomly selected individual to avoid clutter: for each target body part individual predictions from all other body parts are obtained and combined the same way as in Fig. 4. Color-coding corresponds to different body parts. Multiple scoremaps are overlaid on the same image for visualization purposes.
of body parts of a person visible in the image. For example, in the image 4 it correctly outputs only visible parts for the person shown in red. *DeeperCut* is also able to correctly assemble body parts even for rare body articulations as in the image 25. Remarkably, the proposed approach also correctly handles cases of strong partial occlusions. For example, in the image 1 body parts are correctly associated to subjects shown in blue, cyan and magenta, even though only small portion of the cyan subject is visible in the image.

We illustrate and analyze the failure cases of *DeeperCut* in Fig. 7. The included examples further illustrate the difficulty of the task of jointly estimating body articulations of multiple people. We identify several prominent failure modes, and include examples for each mode. First row shows examples of cases when *DeeperCut* generates a body configuration by merging body parts of several people. In these examples the proposed pairwise terms failed to disambiguate between people due to their close proximity in the image. For example, in the image 7 *DeeperCut* groups left and right limbs of the dancing pair. In the resulting configuration the positions of upper limbs are geometrically consistent with each other, but are not consistent with respect to their appearance. Modeling consistency in appearance between left and right extremities should help to mitigate this type of errors and we will aim to address this in the future work. Another type of errors shown at the bottom of Fig. 7 corresponds to cases when *DeeperCut* outputs spurious body configurations that can not be assigned to any ground-truth annotation. This happens either when body parts of the same person are grouped into several distinct clusters, or when a consistent body configuration is formed from detections in background. Both of these cases are penalized by the AP evaluation measure. In Fig. 7 we also visualize examples of failures due to confusion between similarly looking left and right limbs as in the image 10 and occasional failures of pose estimation on rare body configurations as in image 14.

References

1. Johnson, S., Everingham, M.: Clustered pose and nonlinear appearance models for human pose estimation. In: BMVC’10
2. Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B.: 2d human pose estimation: New benchmark and state of the art analysis. In: CVPR’14
3. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS’12
4. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR,14
5. Andriluka, M., Roth, S., Schiele, B.: Discriminative appearance models for pictorial structures. IJCV’11
6. Yang, Y., Ramanan, D.: Articulated human detection with flexible mixtures of parts. PAMI’13
7. Pishchulin, L., Andriluka, M., Gehler, P., Schiele, B.: Poselet conditioned pictorial structures. In: CVPR’13
8. Tompson, J.J., Jain, A., LeCun, Y., Bregler, C.: Joint training of a convolutional network and a graphical model for human pose estimation. In: NIPS’14
Fig. 6. Examples of successful pose estimation results obtained with our DeeperCut model on the MPII Multi-Person dataset.
Fig. 7. Failure cases of our DeeperCut model on the MPII Multi-Person dataset.
9. Chen, X., Yuille, A.: Articulated pose estimation by a graphical model with image dependent pairwise relations. In: NIPS’14
10. Pishchulin, L., Insafutdinov, E., Tang, S., Andres, B., Andriluka, M., Gehler, P., Schiele, B.: Deepcut: Joint subset partition and labeling for multi person pose estimation. In: CVPR’16
11. Wei, S.E., Ramakrishna, V., Kanade, T., Sheikh, Y.: Convolutional pose machines. In: CVPR’16
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385 (2015)
13. Ramanan, D.: Learning to parse images of articulated objects. In: NIPS’06
14. Jiang, H., Martin, D.R.: Global pose estimation using non-tree models. In: CVPR’09
15. Johnson, S., Everingham, M.: Learning Effective Human Pose Estimation from Inaccurate Annotation. In: CVPR’11
16. Tran, D., Forsyth, D.A.: Improved human parsing with a full relational model. In: ECCV’10
17. Wang, F., Li, Y.: Beyond physical connections: Tree models in human pose estimation. In: CVPR’13
18. Gkioxari, G., Arbelaez, P., Bourdev, L., Malik, J.: Articulated pose estimation using discriminative armlet classifiers. In: CVPR’13
19. Tompson, J., Goroshin, R., Jain, A., LeCun, Y., Bregler, C.: Efficient object localization using convolutional networks. In: CVPR’15
20. Ramakrishna, V., Munoz, D., Hebert, M., Bagnell, A.J., Sheikh, Y.: Pose machines: Articulated pose estimation via inference machines. In: ECCV’14
21. Eichner, M., Ferrari, V.: We are family: Joint pose estimation of multiple persons. In: ECCV’10
22. Ladicky, L., Torr, P.H., Zisserman, A.: Human pose estimation using a joint pixel-wise and part-wise formulation. In: CVPR’13
23. Chen, X., Yuille, A.: Parsing occluded people by flexible compositions. In: CVPR. (2015)
24. Bansal, N., Blum, A., Chawla, S.: Correlation clustering. Machine Learning 56(1–3) (2004) 89–113
25. Demaine, E.D., Emanuel, D., Fiat, A., Immorlica, N.: Correlation clustering in general weighted graphs. Theoretical Computer Science 361(2–3) (2006) 172–187
26. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. CVPR (to appear) (November 2015)
27. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected crfs. In: ICLR. (2015)
28. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
29. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: CVPR’15
30. Lee, C.Y., Xie, S., Gallagher, P., Zhang, Z., Tu, Z.: Deeply-supervised nets. In: AISTATS’15
31. Girshick, R.: Fast r-cnn. In: ICCV’15
32. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)
33. Sapp, B., Taskar, B.: Multimodal decomposable models for human pose estimation. In: CVPR’13
34. Toshev, A., Szegedy, C.: Deeppose: Human pose estimation via deep neural networks. In: CVPR’14
35. 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’15
36. Carreira, J., Agrawal, P., Fragkiadaki, K., Malik, J.: Human pose estimation with iterative error feedback. In: CVPR’16
37. Ghiasi, G., Yang, Y., Ramanan, D., Fowlkes, C.: Parsing occluded people. In: CVPR’14