UNIFYING PART DETECTION ASSOCIATION FOR MULTI-PERSON POSE ESTIMATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Current bottom-up approaches for 2D multi-person pose estimation (MPPE) detect joints collectively without distinguishing between individuals. Associating the joints into individual poses is done independently of the learning algorithm, therefore requires formulating a separate problem in a post-processing step, which relies on relaxations or sophisticated heuristics. We propose a differentiable learning-based model that performs part detection and association jointly, thereby eliminating the need for further post-processing. The approach introduces a recurrent neural network (RNN), which takes dense low-level features as input and predicts the heatmaps of a single person's joints in each iteration, then refines them in a feedback loop. In addition, the network learns a stopping criterion in order to halt once it has identified all individuals in an image, allowing it to output any number of poses. Furthermore, we introduce an efficient implementation that allows training on memory-constrained machines. The approach is evaluated on the challenging COCO and OCHuman datasets and substantially outperforms the baseline. On OCHuman, which contains severe occlusions, we achieve state-of-the-art results even compared to top-down approaches. Our results demonstrate the advantage of a learning-based detection and association framework, and the advantage of bottom-up approaches over top-down approaches in complex scenarios.

1 INTRODUCTION

The task of multiperson human pose estimation is defined as the localization of a predefined set of anatomical joints and their grouping distinctly into individuals. It is an integral component for many applications in computer vision in which humans are active participants such as assisted driving Fridman et al. (2017), assisted living Planinc et al. (2016) and sports analysis Shih (2017). Challenges associated with the tasks include variation in joint articulation, occlusions, overlaps or close proximity of joints. Current prevalent approaches are largely based on two paradigms, the first follows a top-down design Iqbal & Gall (2016), in which first a single person is detected and then his/her joints are localized. Such approaches rely on bounding box proposals and are therefore not robust to occlusions or partial visibility of people. Incorrect detections in this stage carry over to the joint localization stage, which is not aware of association. The second paradigm follows a bottom-up approach design in which first a set of joint candidates of all people are collectively identified without distinction, and then grouped into poses for each person individually Pishchulin et al. (2016); Cao et al. (2017); Newell et al. (2017). In current state-of-the-art bottom-up approaches, grouping is performed independently of the training, and entails formulating a separate problem that usually relies on greedy heuristics and relaxations. In these approaches, the ultimate objective remains unknown to the training algorithm as it optimizes over an intermediate objective, thereby limiting the network in using its full potential. Furthermore, when such heuristics are employed, it remains unclear how the association can be further refined or generalized. Cao et al. (2017) who follow a bottom-up approach, train two different branches for the part confidence map and for the association, referred to as Part Affinity Fields, which are used to capture the relationships between the joints and are used as edge weights for optimization in a graph matching problem only during inference. Since the formulation results in a k-dimensional matching problem, an NP-hard problem, the problem was solved using greedy relaxation thereby yielding a suboptimal solution. Newell et al. (2017) have devised a loss that tries to guide the network in learning distinct embeddings for each individual.

However, these embeddings are not employed for association during training and are only used in a greedy post-processing step. In this work, we propose a end-to-end differentiable approach which conflates part detection and association into a single model. We aim to show the benefit of such a learning-based approach for detection and association, as well as the advantage of bottom-up approaches over top-down. The approach is evaluated on the challenging COCO keypoint dataset Lin et al. (2014) and extremely challenging OCHuman dataset Zhang et al. (2019a).

1.1 Related work

In multi-person pose estimation, methods that follow a top-down design such as Carreira et al. (2016); Iqbal & Gall (2016); Chen et al.; Xiao et al. (2018); Huang et al. (2017); He et al. (2017) rely on person detectors and estimate the pose for each detected bounding box individually. The performance of these approaches therefore is tightly coupled with the performance of the underlying person detector. Furthermore, if people overlap heavily, non-maximum-suppression (NMS) results in eliminating some of them. Conversely, bottom up approaches such as Pishchulin et al. (2016); Cao et al. (2017); Newell et al. (2017); Iqbal et al. (2017); Doering et al. (2018); Raaj et al. (2018); Papandreou et al. (2018); Haoshu Fang & Lu (2017); Papandreou et al. (2017); Huang et al. (2017) directly estimate joint locations for individuals all at once without distinction, and require assembling the joints into individual poses subsequently. Several works have proposed to merge the joint estimates into poses by generating a fully-connected graph Pishchulin et al. (2016); dee (2016); Iqbal et al. (2017); Insafutdinov et al. (2017) based on the joint estimates, and solving a matching problem by utilizing ILP (Integer Linear Programming). A closely related work is Wang et al. (2018) who introduce a minimum weight set packing formulation to solve the association problem of the part detections while applying Nested Benders Decomposition to achieve a more efficient inference time. Other works Cao et al. (2017); Newell et al. (2017); Raaj et al. (2018) predict features such as vector fields Cao et al. (2017); Raaj et al. (2018) which indicate the correspondence between detected joints. This allows reducing the pose assembly into a greedy bipartite graph matching problem as in Cao et al. (2017). These approaches formulate association as a separate optimization problem that does not participate in the training. Kocabas et al. (2018) propose a method that combines person and part detections, such that keypoints and person bounding boxes are simultaneously detected, followed by assigning the keypoints to person boxes using a learned function. Carreira et al. (2016) introduce a corrective iterative feedback method, in which the CNN predicts an additive correction to the current joint estimate in the Cartesian representation. However, part detection based on confidence maps has been shown as more powerful at capturing context and relationships between the joints Tompson et al. (2014); Newell et al. (2017; 2016); Cao et al. (2017); Pishchulin et al. (2016); Wang et al. (2018) than regression based bottom-up methods.

Another problem that is related to multi-person pose estimation is the task of instance segmentation. Romera-Paredes & Torr (2016) propose using an RNN for instance segmentation, but without inferring the label of an instance. Salvador et al. (2017) eliminate this shortcoming by introducing an additional branch that predicts the class of each instance. However, they do not localize the exact joint locations nor address the distinct case of people instances which have unique articulation features. Zhang et al. (2019a) extract human pose features in order to perform person instance segmentation instead of relying on bounding-box proposals.

To the best of our knowledge, there is no bottom-up method that estimates the heatmaps of each person individually or performs learning-based association of detected parts.

2 Background

2.1 Associative Embedding (AE)

In implementing our approach, we make use of the extracted features of AE's Stacked Hourglass network Newell et al. (2017). It is noteworthy, however, that the proposed approach is independent of the underlying network and can be combined with any bottom-down approach. In their work, the authors have devised a network for dense predictions that can be used for solving multi-person pose estimation. Since we use the extracted features of Newell et al. (2017) and compare our approach with theirs, we first give a brief description of their work. In the Stacked Hourglass architecture, several CNNs that have a symmetric structure are stacked together. The output of the stacks are similar, and each stack output comprises maps of part detections, associative embeddings, and representa-

tion features extracted at the intermediate layers. A part detection heatmap I_j detects all joints of type j without distinguishing which person they belong to. The embedding maps aim at producing distinct values for each person instance, with similar values for the same instance.

2.2 HEURISTIC-BASED ASSOCIATION

The embeddings of the AE network are utilized in a post-processing step in order to group the joints into individual poses. Since the heatmaps are shared for all people, they perform several additional steps that are based on hand-crafted heuristics in order to find the grouping. In what follows we describe these operations in order to show their potential disadvantages: before taking candidate detections of every joint confidence map I_j, NMS is applied on each of the confidence maps. This operation requires specifying a neighborhood size from which the maximum detection is taken, therefore may erroneously suppress nearby joint detections. The candidate detections are thresholded such that only detections that are above a predefined threshold can be associated with individuals. While iterating over the joint types $j \in J$ in order to decide which person every detection of this type is assigned to, a greedy approach is employed, in which an assignment problem is solved with a cost matrix consisting of the distance between the embeddings corresponding to unassigned detections and the average embedding values of the joints that have been aggregated up until this point. The assignment problem finds a locally optimal choice since it uses only the knowledge about the joints that have been aggregated up until joint j, indicating that it is not necessarily the optimal solution. For associating these joint detections with a person or deciding that a new person is discovered, an embedding threshold value needs to be specified such that only joints within this distance may be grouped together. The authors also favor matching an unassigned joint to a higher confidence detection, that is, if an unassigned joint is close to more than one person by the embedding distance, then it is assigned to the person with the highest score. Such a heuristic assumption is hard, and does not necessarily hold true. A non-learned heuristic approach can work well in scenarios when people do not overlap or occlude each other, but will struggle in more challenging scenarios. For instance, in AE, when deciding if a new person is discovered, the method relies solely on a threshold value. Unlike a learning algorithm, it has no notion of what a person is and it tends to overestimate the number of person poses, many of which are invalid.

3 APPROACH

3.1 UNIFIED MODEL OF DETECTION AND ASSOCIATION

Having motivated a learning-based model, we now introduce our approach. We use a recurrent neural network (RNN) as a decoder network intended to decode the extracted features of the Stacked Hourglass network into the final distinct human poses. In general, RNNs have a remarkable expressiveness power whose mechanism design allows for weight sharing across time steps and for predicting a variable-length output. In addition, an RNN is an effective model for predicting variable-length sequences that contain dependency along the domain relevant to the prediction. RNNs have been successfully applied for modeling sequential outputs with dependency along the temporal domain in various tasks such as machine language translation Sutskever et al. (2014), image captioning Donahue et al. (2015), action recognition Richard et al. (2017), and motion prediction Martinez et al. (2017). They have been additionally employed for various visual tasks where the sequential dependency lies in the spatial domain, such tasks include object recognition Visin et al. (2015); Mnih et al. (2014), semantic segmentationVisin et al. (2016), scene labeling Byeon et al. (2015); Pinheiro & Collobert (2014), instance segmentation Romera-Paredes & Torr (2016). The motivation behind using them in these tasks is their ability in learning long range dependencies and propagating global contextual information across the time steps. In the case of multi-person pose estimation, the sequence consists of the individual poses appearing in the image and the RNN decides where in the image to estimate a new person's pose using its state, such that it should not estimate the same person's pose repetitively. Such behavior is prevented by the memory maintained in the RNN hidden state, which accumulates information in its internal representation as it processes the sequence. Every iteration in the RNN is responsible for estimating a single person's part heatmaps. Since any permutation of the output sequence is valid, we allow the network to decide its ordering as well. This freedom allows the network to learn an ordering that bests fits its objective, based on its current hidden state and input features. The RNN variant used in this task is based on Convolutional Short-

Term Memory network (ConvLSTM) SHI et al. (2015). A ConvLSTM replaces the fully connected layers of LSTM network Hochreiter & Schmidhuber (1997) with convolutional ones and attains the same advantages that convolutional layers do over fully connected layers for image recognition, such as more efficient training due to a reduced number of parameters, better encoding of spatial information and less proneness to overfitting.

3.1.1 TRAINING PROCEDURE

As input to the LSTM, we use intermediate features of AE which have been extracted from the Stacked Hourglass module in Newell et al. (2017)] and are attuned towards part detection. The input is fed-forward through several ConvLSTM layers which decode the extracted features into the final part heatmaps. Additionally, in order to further refine the heatmap predictions, we introduce a feedback loop which takes the distinct confidence maps for each person and concatenates them along with the rest of the input features to the LSTM in two consecutive iterations. The motivation behind introducing this loop is to encourage the network to produce a single peak in each of its estimates of the disjoint confidence maps. It is noteworthy that since the input is fed again through a feed-back loop to the same network, there is no overhead of additional parameters. The hidden state of the last LSTM unit is then used as input features to a convolution layer at the output, which yields J heatmaps belonging to a single person. In every iteration t of the LSTM, J confidence maps for a single person are inferred directly. If $\hat{z}$ poses are predicted, then the LSTM is unrolled $2 \times (\hat{z} + 1)$ times before it finds all the poses, where the additional $\hat{z}$ iterations are for the refinement feedback loop. Each value in the heatmap $\hat{H}_{t,j}$ represents the network's confidence about the corresponding pixel being the location of joint j. The ground-truth (GT) confidence maps $H_{t,j} \in \mathbb{R}^{m \times n}$ for person t and joint j are created with a 2D Gaussian distribution centered at the GT location of the joint with a small variance. Unlike in Newell et al. (2017), the GT heatmaps are created disjointly for each person and are not aggregated into a single heatmap. The number of GT heatmaps is therefore equal to $z \times J$ instead of J, where z is the true number of people in the image. Accordingly, a GT heatmap $H_{t,j}$ contains a single peak corresponding to the joint's location. This implies that a single prominent peak should be predicted in a single output heatmap $\hat{H}_{t,j}$ as opposed to k peaks in shared heatmaps.

In order to halt the LSTM, we need an additional prediction value that indicates when all poses present in the input image have been estimated. To achieve that, we predict an additional value $\hat{p_t} \in [0, 1]$ associated with $\hat{H}_t$, which signifies the network's confidence in its current prediction being a new valid pose and halt when $\hat{p} < \omega$ for a predefined threshold $\omega \in [0, 1]$. This value is calculated using an additional fully-connected layer with the hidden state matrices from the LSTM layers as input, followed by a sigmoid function that converts the output to a valid probability value in the range $[0, 1]$. The GT for these values is a vector p created as follows:

$$\mathrm{p} = \begin{cases} [\underbrace{1 \ldots 1}_{z} \, 0] & \hat{z} \leq z \\ [\underbrace{1 \ldots 1}_{z} \; \underbrace{0 \ldots 0}_{(\hat{z}-z)+1} \;] & \text{otherwise} \end{cases} \tag{1}$$

the length of the vector $|p| = \max\{\hat{z}, z\} + 1$, where the purpose of the additional iteration is providing knowledge to the network about the stopping criterion. During training, we have set the stop probability threshold to $\omega = sigmoid(-1) \approx 0.26$. For a threshold value close to 0.5, false negatives such as blurred people can be missed, and for a threshold value close to 0.1 more false positives are predicted.

To learn the part heatmaps and the stop probability, we add two loss terms on top of the LSTM, the first is the squared l_2 loss between the distinct confidence maps per person and corresponding GT heatmaps. And the second term is calculated using the cross entropy loss between the prediction $\hat{p}$ and vector p defined in eq. 1. During training, the network may fail to identify all individuals, i.e. $\hat{z} < z$, in this case we let the RNN iterate until $(z + 1)$ to obtain all values of $\hat{p}$, but penalize only heatmaps with $\hat{p}_t \geq \omega$. Conversely, if the number of people is overestimated, i.e. $\hat{z} > z$, we take into account only heatmaps until z. The motivation is to encourage the network to learn to predict the correct number of people and to make the correct prediction of the heatmaps from the first attempt, i.e. before overestimating z. This reasoning is supported by additional experiments in table 4.

3.1.2 APPENDING A SECOND LSTM

The order in which the poses are estimated by the LSTM is determined by its weights, while the network will implicitly try to learn an ordering that best benefits its objective. Since ordering is irrelevant in pose estimation, it is not clear how to define a GT permutation for a given image. As such, if an image contains k persons, explicitly optimizing over ordering entails searching in a solution space of $k!$ permutations, which is intractable for a large k. However, since in LSTMs the weights are shared across all iterations, as opposed to feed-forward models, explicitly optimizing over ordering as done in feed-forward models Rezatofighi et al. (2018) is not required. Predictably, when we implemented our approach using a feed-forward CNN from samples with a fixed number of people, the approach broke down. This is caused by discontinuity in assigning a certain ordering to the network's outputs, where a certain weight is attuned to a specific output, as shown in Zhang et al. (2019b). Several works have tackled the ordering problem in deep learning. Vinyals et al. (2015) have shown that in Seq2Seq the ordering in either the input or output affects performance. Rezatofighi et al. (2018) estimate the posterior distribution of all permutations using alternating optimization during learning, which requires searching in a solution space ok $k!$ for a permutation size of size k. In Murphy et al. (2019), the authors propose permutation sampling to overcome the complexity of searching in the entire permutation space and show that it has similar convergence properties to an optimal solution with SGD. In order to examine the optimality of the prediction of the first LSTM (LSTM1) and compare it with random sampling, we introduce a second LSTM (LSTM2) whose input at every iteration t is a concatenation of the intermediate features and the individual HM $\hat{H}_t$. The advantage of the second LSTM is twofold. Firstly, it enables further refinement of the predicted poses. Secondly, having the poses in a sequence provides a simple way to enforce a certain ordering at LSTM2's output. For instance, we can feed the poses in a random, inverse or canonical ordering, and then force LSTM2 to output a refined heatmap of its input heatmap in the same ordering as the input.

3.2 LOSS TERMS

Since LSTM1 is not constrained to output the poses in any particular ordering, in order to compute the heatmap loss, the pose instances need to be associated with the correct individual GT heatmap. This can be done by solving an assignment problem using the Hungarian algorithm, in which given the cost matrix $C \in \mathbb{R}^{z \times \hat{z}}$, the problem requires finding an assignment $S \in \mathbb{N}^r$ such that $\sum_{t=1}^{r} \sum_{j=1}^{J} ||H_{s_t,j} - \hat{H}_{t,j}||$ is minimized, $r = \min\{z, \hat{z}\}$. Each element in the cost matrix C is given by $C_{kk'} = \sum_{j=1}^{J} ||H_{k,j} - \hat{H}_{k',j}||$, which is the sum of distances between each pair of a person's heatmaps in the prediction and GT. Taking the sum of the pixel-wise distance between each pair of heatmaps in the GT and output is less sensitive to inaccurate predictions than taking the distance between the maximum activations.The loss term of LSTM2 consists of the heatmap loss between the GT heatmaps reordered in the same input permutation u. $\hat{H}_1, \hat{H}_2$ denote the HM output of LSTM1 and LSTM2 respectively. In summary, the total loss is a sum of all loss terms:

$$l_{HM_1} = \sum_{t=1}^{r} \sum_{j=1}^{J} l_2(H_{s_t,j}, \hat{H}_{1t,j}) \tag{2}$$

$$l_{stop} = \sum_{t=1}^{|p|} p_t \log \hat{p_t} + (1 - p_t) \log(1 - \hat{p_t}) \tag{3}$$

$$l_{HM_2} = \sum_{t=1}^{r} \sum_{j=1}^{J} l_2(H_{u(s_t),j}, \hat{H}_{2t,j}) \tag{4}$$

and all loss terms are weighted equally. Rearranging the GT heatmaps based on the input permutation enforces the input permutation at the output. Figure 3 illustrates the approach.

To obtain the final joint location x_j, an $\arg\max$ operation is applied on each joint's heatmap, i.e. $x_j = \arg\max(\hat{H}_{t,j})$. We specify a threshold value $\tau = 0.015$, such that if $\hat{H}_{t,j}(x) < \tau$, the joint is discarded. For further refinement and in order to provide insight to the optimality on the ordering decided by the LSTM, we add an additional LSTM whose input is a concatenation of the AE features

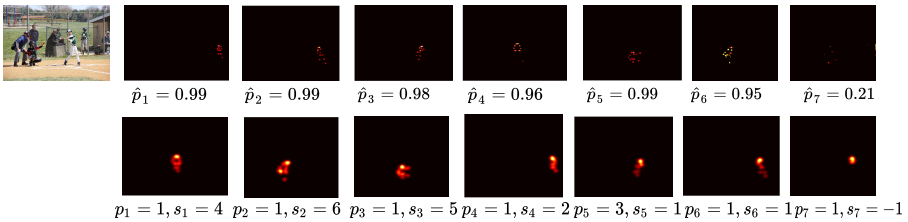


Figure 1: The first raw are the confidence map predictions of the RNN, where each person i is a sum of the J confidence maps predicted at iteration i. $s_k = i$ indicates H_k is matched with $\hat{H}_i$. $S = \{s\}_1^r$ is the assignment of the GT heatmaps, which appear in some arbitrary order. In this example, $z = 8$ and $\hat{z} = 6$, hence $s_{7,8} = -1$.

Figure 2: An example output of a single LSTM iteration, in which J heatmaps are predicted for a single person. We overlay them as is on the image, and sum them up for the last image to show correspondence for all the joints of the person.

and an individuals heatmap $\hat{H}_t$ in each iteration. The loss on the second LSTM is the square loss between the output $\hat{H}_2$ and the GT heatmaps H.

3.2.1 Within-Image Batching Using Approximate SGD

For an image with a large number of people, a GPU will run out of memory. Our 1080 Ti machine for instance supports a maximum of 8 iterations. Simply cutting and stopping the LSTM after $T = 8$ iterations despite $\hat{p}_t \geq \omega$ means losing within-image person samples. Accordingly, in order not to overlook these remaining samples, we apply within-image batching, wherein we allow the number of iterations to grow until $\hat{p} < \omega$ or until a predefined number of iterations is reached (suitably the maximum number of people in a dataset), but we backpropagate the accumulated gradient after every $T = 8$ iterations, where T denotes the mini-batch size. The LSTM's hidden state is not cleared after every T, since it will be needed to keep track of poses that have already been estimated, enabling the network to pick up estimating the remaining poses from where it last backpropagated. In order not to assign the same GT pose to a predicted pose again, we remove the assigned GT heatmaps from the vector of GT heatmaps. The assignment in this case is not necessarily optimal, as it may happen that an assigned GT heatmap in the first mini-batch should rather be assigned in the second mini-batch, in practice, however, we observed that this is not an issue. The subsequent gradient in this case no longer depends on the earlier hidden states in the chain rule. The algorithm is described in alg. 1. Note that if we ignore the remaining within-image samples by simply stopping after only 8 iterations, the performance drops by only 0.7 in the average precision (AP).

3.3 Implementation

We implement our approach in Pytorch on top of the publicly available library [1] of Newell et al. (2016). The AE model had been trained on COCO dataset, and we use the same dataset for training our model. The input to the AE model is an image resized to a 512×512 resolution, and its output is a set of maps $M \in \mathbb{R}^{4\times 68\times m\times n}$, $m = n = 128$, where 4 corresponds to the number of

[1] https://github.com/princeton-vl/pose-ae-train

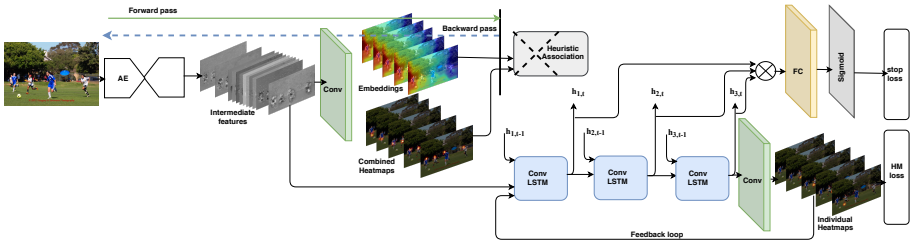
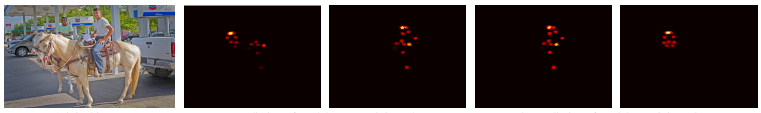


Figure 3: Overview of the approach. AE network predicts joint heatmaps collectively. It follows this stage by a heuristic-based association using the embeddings. Instead, our approach estimates an individual human pose directly in every iteration t. The input to the first ConvLSTM is a concatenation of the current hidden state from the previous iteration, the intermediate features extracted from the AE network, and the LSTM prediction of the individual heatmaps using the feedback loop. The input to the subsequent layers is a concatenation of the output of the previous layer and previous hidden state. The features used to calculate the confidence value $\hat{p}$ are the concatenation of the maximum activations of the hidden state from each layer (Due to space, LSTM2 is not shown).



Original image (a) Prediction of model not containing a loop (b) Prediction of model containing a loop

Figure 4: Showing the effect of the feedback loop. As expected, it helped in suppressing wrong peaks in its confidence map. This can be observed in top left photo, where the person in the back is not distinctly detected from the person in the front.

Hourglass stacks, and 64 are the channels for the heatmaps and embeddings each of size $J = 17$, and intermediate features of size $f = 34$. In total, the input to the first LSTM is a concatenation of the intermediate features B of all stacks, the distinct heatmap features of the feedback loop $\hat{H}_{t,j}$, and the previous hidden state h_{t-1}; $F \in \mathbb{R}^{c \times m \times n}, \hat{H}_{t,j} \in \mathbb{R}^{m \times n}, c = 4 \cdot f$. The number of hidden units in LSTM1 is 3, and 2 units in LSTM2. The kernel size of the convolution in the ConvLSTM is 3. The hidden state size is reduced by factor 2 at the output of every LSTM. In the first iteration of the feedback loop, the confidence maps are initialized with a uniform distribution $\sim [0, 0.1]$. Since the batch size b is 1, we normalize the input features with an affine transform using the formula $y = \gamma\hat{x} + \beta$ Ioffe & Szegedy (2015), in which the running statistics of the entire data are used instead of those of a single batch. This achieves a 2% improvement. The network has been trained for 9 epochs with batch size $b = 1$, with every epoch taking approximately 10 hours on a 1080 Ti GPU. A larger batch size decreased performance. We used the Adam optimizer, with an initial learning rate of $1e^{-4}$, dropped to $1e^{-5}$ at epoch 4. In the multi-scale evaluation, the authors of Newell et al. (2017) apply single-person refinement for missing joint detection. Since our implementation is based on theirs, for fair comparison, we apply the same procedure.
Invariance to multi-scale transformations remains a challenge for CNNs which are not explicitly designed to be invariant to different scales during training, even when random transformations are applied as part of the data augmentation Jaderberg et al. (2015). As such, similarly to Newell et al. (2017), we feed-forward the image using multiple scales in AE and average the output confidence maps for the single-person refinement.

4 Experiments

We evaluate our model model on two datasets, COCO validation dev and OCHuman validation and test sets. We summarize the results on COCO in table 1. On COCO, the gap in the single scale performance is larger than in multi-scale (MS) inference and the heuristics of AE seem to benefit more with from MS. It can also be seen that current top-down (TD) approaches outperform bottom-up (BU) approaches on COCO. Coco, however, contains too many easy examples where the persons do not occlude or overlap with each other heavily, therefore does not reflect the performance of TD approaches in more challenging real-life scenarios. In contrast, OCHuman dataset contains

Input: extracted features F, H_{GT}, P_{GT}
Initialize $\hat{H} \leftarrow [], \hat{P} \leftarrow [], t \leftarrow 1, S \leftarrow (0)$
1 **while true and** $t <$ MAX_ITERATIONS :
2 $\quad \hat{H}_t \leftarrow uniform\_random(0, 0.1)$
3 $\quad i \leftarrow 0$
4 $\quad$ **repeat** :
5 $\quad\quad B \leftarrow concat(F, \hat{H}_t)$
6 $\quad\quad \hat{H}_t, \hat{p}_t, S \leftarrow LSTM(B, S)$
7 $\quad\quad i \leftarrow i + 1$
8 $\quad$ **until** $i = 2$
9 $\quad \hat{H}.append(\hat{H}_t)$; $\hat{P}.append(\hat{p}_t)$
10 $\quad$ **if** $\hat{p}_t < \omega$ **or** $t \; mod \; T = 0$:
11 $\quad\quad assignment \leftarrow match(\hat{H}, H)$
12 $\quad\quad H \leftarrow H[assignment]$
13 $\quad\quad loss \leftarrow calculate\_loss(\hat{H}, H_{GT}, \hat{P}, P_{GT})$
14 $\quad\quad backpropagate()$
15 $\quad\quad$ **if** $\hat{p}_t < \omega$ **and** $t = |p|$:
16 $\quad\quad\quad$ **break**
17 $\quad H \leftarrow H \setminus H[assignment], \hat{H} \leftarrow [], \hat{P} \leftarrow []$
18 $\quad t \leftarrow t + 1$
19 **end while**

Algorithm 1: The training algorithm. Line 17 removes GT heatmaps that were last assigned.

Method	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}
Bottom-Up					
OpenPoseCao et al. (2017)	61.8	84.9	67.5	57.1	68.2
PersonLabPapandreou et al. (2018)	68.7	89.0	75.4	64.1	75.5
AE singlescaleNewell et al. (2017)	56.6	81.8	61.8	49.8	67.0
Ours singlescale	60.1	82.2	65.4	51.9	72.5
AE multiscale Newell et al. (2017)	65.5	86.8	72.3	60.6	72.6
Ours multiscale	66.8	85.2	73.0	59.7	77.3
Top-Down					
Mask-RCNN He et al. (2017)	63.1	87.3	68.7	57.8	71.4
G-RMI Papandreou et al. (2017)	64.9	85.5	71.3	62.3	70.0
CPN Chen et al.	72.1	91.4	78.9	68.7	77.2
RMPE Haoshu Fang & Lu (2017)	72.3	89.2	79.1	68.0	78.6
CFN Huang et al. (2017)	72.6	86.1	69.7	78.3	64.1
MSRA Xiao et al. (2018)	73.7	91.9	81.1	70.3	80.0

Table 1: Results on COCO validation set.

only difficult cases of occluded and interwined persons and the average IoU of the bounding boxes is 67%. It consists of 4731 images that comprise a validation and test set with a total of 8110 annotated humans. The purpose of this dataset is to examine the limitations of human detection in highly challenging scenarios. Accordingly, it does not contain training samples and is intended to be used for evaluating existing models. The results are summarized in table 2. The results on this dataset show the limitations of approaches in which association is not part of the learning algorithm, and of TD approaches. Our approach achieves state-of-the-art results on OCHuman and outperforms both TD approaches on both the validation and test subsets although both Haoshu Fang & Lu (2017); Xiao et al. (2018) use a much stronger backbone architecture. Compared to AE Newell et al. (2017), the gain in accuracy is larger than on COCO since the separation of highly overlapping people is more difficult. In Zhang et al. (2019a), through evaluation of person detection on OCHuman, the authors show that for human detection in complex scenarios, thanks to the distinctiveness of the human pose,

Figure 5: Qualitative results from COCO in the first row, and OCHuman in the second row[add examples from OCHuman. maybe also show results of the other approaches

pose skeleton features obtained by a BU approach should be used, instead of the reverse direction employed in TD approaches, where bounding-box proposals of two heavily overlapped people will likely result in eliminating one of them when applying NMS.

4.1 ORDERING

By adding a second LSTM, examining the ordering effect on performance has become straightforward. The optimal results are obtained when we use the same ordering determined by the first LSTM. This indicates that a data-driven learned ordering is nearly the best we can do at this complexity. Feeding the heatmaps in an inverse ordering degrades the performance substantially. Canonical ordering obtained by sorting based on the maximum activation of the first LSTM yields a low precision too. In table 3, the experiment "learned" is the permutation decided by LSTM1. Experiment "random" is a random permutation of LSTM1's output. "Confidence" is the confidence value of LSTM1 sorted in descending order and canonical is sorting of the permutation based on ascending order of the joint location (based on the maximum activation of every joint's heatmap). It can be seen that learned performs best followed closely by "random". Sorting by the confidence value is not optimal since the network does not always output poses in descending order of confidence, indicating that the ordering does not necessarily always correlate with the confidence value. "Inverse" shows the importance of ordering and that while the ordering of LSTM1 might not be optimal, the inverse solution is very detrimental. "Canonical" shows that enforcing a certain ordering is also detrimental, since the ordering that most benefits the network does not depend on the location for instance. We presume that a favorable ordering is one in which easy poses are estimated first, followed by hard ones. Hardness of a pose can be assessed by conditions such as resolution, scale, blurriness and illumination. In such an ordering, the network can use the accumulated contextual information in its hidden state in order to estimate an occluded person following an occluding person for instance. If LSTM2 consists of a larger number of hidden units, there's sometimes an advantage in a random permutation over a learned permutation (see 7 in the appendix). With two LSTMs, if the first LSTM does not have a feedback loop and the second does, the performance drops substantially, indicating that it is more critical to refine the heatmaps at an earlier stage. We designed an additional experiment in which we have a single loss on only LSTM2 only. In this case the performance drops from 60.3 to 53.2 indicating that simply increasing the number of layers is not advantageous and that having an initial prediction is very helpful for the additional layers.
To get a fair comparison between an exact SGD (E-SGD) and approximate SGD (A-SGD), we train our network with images that have a maximum 8 people so that approximate SGD, in which we backpropagate after estimating 4 persons and solve the assignment with respect to the first subset of 4 poses, and then the second subset, A-SGD achieves 56.8 while E-SGD achieves 56.2, which is an interesting result showing that the long term dependencies propagated by the gradient are not always beneficial.

OCHuman	Val	Test
AE Newell et al. (2017), SS	32.1	29.5
Ours, SS	39.3	32.5
AE Newell et al. (2017), MS	40.0	32.8
Ours, MS	**41.9**	**35.5**
RMPE Haoshu Fang & Lu (2017)*	38.8	30.7
MSRA Xiao et al. (2018), ResNet50 *	37.8	30.4
MSRA Xiao et al. (2018) ResNet152 *	41.0	33.3

Table 2: Evaluation on OCHuman Zhang et al. (2019a) validation and test sets. SS/MS indicates single/multi scale, respectively, and * indicates a top-down approach.

	learned	random	confidence	inverse	canonical
COCO	**60.1**	59.3	58.2	42.1	38.8
OC-val	**39.4**	39.1	37.7	25.4	23.4
OC-test	**32.5**	32.2	30.4	30.2	20.6

Table 3: AP with several ordering variants on the input to LSTM2.

	(a, a')	(a, b')	(b, a')	(b, b')
COCO	58.7	57.1	58.0	54.7
OC-val	37.8	35.7	36.9	35.8
OC-test	31.7	30.5	31.0	30.1

Table 4: Results with different configurations in calculating the heatmap loss.

4.2 Ablation studies

There are two cases in mispredicting $\hat{z}$: **(1)** $\hat{z} < z$, and then we either take into account heatmaps (a) only with $\hat{p}_t \geq \omega$, or (b) all heatmaps including $\hat{p}_t < \omega$. **(2)** $\hat{z} > z$ and then we solve the assignment (a') for only the first z of $\hat{z}$ heatmaps, or (b') for all $\hat{z}$ heatmaps. We train using all four configurations and summarize the results in 4. Configuration (aa') performs best, confirming our intuition that through (a) it is better to disregard predictions in the heatmap loss that the network is not confident about, and let the network improve its prediction by backpropagating the stop loss where $max\{|\hat{z}|, |z|\}$ is taken (eq. 1). Through a', it is better to penalize only heatmaps associated with a correct prediction of $\hat{p}$. Note that our ablative studies are conducted in single-scale with a single LSTM but generalize to two LSTMs and multi-scale inference too. For examining the effect of the feedback loop, we draw a comparison with a model that has been trained without the loop and observe that the network suppresses redundant or wrong peaks in the output. The difference is illustrated visually using output heatmaps in figure 4.

5 Conclusion

We have introduced a unified differentiable model for part detection and association. Our approached surpassed the performance of the baseline on COCO and achieved state-of-the-art results on OCHuman. We have provided an efficient implementation that allows controlling the within-image batch size thereby enabling training on memory-constrained GPUs. We have introduced a method that allows enforcing ordering in the input and given insight to their performance in different settings. Through our experiments, we were able to emphasize the importance of a unified learning-based framework for detection and association, in particular for difficult cases of human poses such as occlusions. Additionally, we have shown the limitations of top-down approaches in such cases and reinstated the importance of bottom-up approaches.

References

DeeperCut: A Deeper, Stronger, and Faster Multi-Person Pose Estimation Model, 2016.

Wonmin Byeon, Thomas M Breuel, Federico Raue, and Marcus Liwicki. Scene labeling with lstm recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3547–3555, 2015.

Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human pose estimation with iterative error feedback. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *CVPR*.

Andreas Doering, Umar Iqbal, and Jürgen Gall. Jointflow: Temporal flow fields for multi person pose estimation. In *BMVC*, 2018.

Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2625–2634, 2015.

Lex Fridman, Daniel E Brown, Michael Glazer, William Angell, Spencer Dodd, Benedikt Jenik, Jack Terwilliger, Julia Kindelsberger, Li Ding, Sean Seaman, et al. Mit autonomous vehicle technology study: Large-scale deep learning based analysis of driver behavior and interaction with automation. *arXiv preprint arXiv:1711.06976*, 2017.

Shuqin Xie Haoshu Fang and Cewu Lu. RMPE: Regional multi-person pose estimation. *ICCV*, 2017.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *ICCV*, 2017.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Shaoli Huang, Mingming Gong, and Dacheng Tao. A coarse-fine network for keypoint localization. In *ICCV*, 2017.

Eldar Insafutdinov, Mykhaylo Andriluka, Leonid Pishchulin, Siyu Tang, Evgeny Levinkov, Bjoern Andres, and Bernt Schiele. ArtTrack: Articulated Multi-person Tracking in the Wild. In *CVPR*, 2017.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Umar Iqbal and Juergen Gall. Multi-person pose estimation with local joint-to-person associations. In Gang Hua and Hervé Jégou (eds.), *Computer Vision – ECCV 2016 Workshops*, pp. 627–642, Cham, 2016. Springer International Publishing. ISBN 978-3-319-48881-3.

Umar Iqbal, Anton Milan, and Juergen Gall. Posetrack: Joint multi-person pose estimation and tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Max Jaderberg, Karen Simonyan, Andrew Zisserman, and koray kavukcuoglu. Spatial transformer networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 28*, pp. 2017–2025. Curran Associates, Inc., 2015. URL http://papers.nips.cc/paper/5854-spatial-transformer-networks.pdf.

Muhammed Kocabas, Salih Karagoz, and Emre Akbas. MultiPoseNet: Fast multi-person pose estimation using pose residual network. In *ECCV*, 2018.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), *Computer Vision – ECCV 2014*, pp. 740–755, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10602-1.

Julieta Martinez, Michael J Black, and Javier Romero. On human motion prediction using recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2891–2900, 2017.

Volodymyr Mnih, Nicolas Heess, Alex Graves, et al. Recurrent models of visual attention. In *Advances in neural information processing systems*, pp. 2204–2212, 2014.

Ryan L. Murphy, Balasubramaniam Srinivasan, Vinayak Rao, and Bruno Ribeiro. Janossy pooling: Learning deep permutation-invariant functions for variable-size inputs. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=BJluy2RcFm.

Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (eds.), *Computer Vision – ECCV 2016*, pp. 483–499. Springer International Publishing, 2016. ISBN 978-3-319-46484-8.

Alejandro Newell, Zhiao Huang, and Jia Deng. Associative embedding: End-to-end learning for joint detection and grouping. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 30*, pp. 2277–2287. Curran Associates, Inc., 2017. URL http://papers.nips.cc/paper/6822-associative-embedding-end-to-end-learning-for-joint-detection-and-grouping.pdf.

George Papandreou, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, and Kevin Murphy. Towards accurate multi-person pose estimation in the wild. 2017.

George Papandreou, Tyler Zhu, Liang-Chieh Chen, Spyros Gidaris, Jonathan Tompson, and Kevin Murphy. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In *ECCV*, 2018.

Pedro HO Pinheiro and Ronan Collobert. Recurrent convolutional neural networks for scene labeling. Technical report, 2014.

Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter V. Gehler, and Bernt Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4929–4937, 2016.

Rainer Planinc, Alexandros Chaaraoui, Martin Kampel, and Francisco Florez-Revuelta. Computer vision for active and assisted living. 2016.

Yaadhav Raaj, Haroon Idrees, Gines Hidalgo, and Yaser Sheikh. Efficient online multi-person 2d pose tracking with recurrent spatio-temporal affinity fields. *arXiv-preprint*, 2018.

S Hamid Rezatofighi, Roman Kaskman, Farbod T Motlagh, Qinfeng Shi, Daniel Cremers, Laura Leal-Taixé, and Ian Reid. Deep perm-set net: learn to predict sets with unknown permutation and cardinality using deep neural networks. *arXiv preprint arXiv:1805.00613*, 2018.

Alexander Richard, Hilde Kuehne, and Juergen Gall. Weakly supervised action learning with rnn based fine-to-coarse modeling. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Bernardino Romera-Paredes and Philip Hilaire Sean Torr. Recurrent instance segmentation. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (eds.), *Computer Vision – ECCV 2016*, pp. 312–329, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46466-4.

Amaia Salvador, Miriam Bellver, Victor Campos, Manel Baradad, Ferran Marques, Jordi Torres, and Xavier Giro-i Nieto. Recurrent neural networks for semantic instance segmentation. *arXiv preprint arXiv:1712.00617*, 2017.

Xingjian SHI, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong, and Wang-chun WOO. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 28*, pp. 802–810. Curran Associates, Inc., 2015. URL http://papers.nips.cc/paper/5955-convolutional-lstm-network-a-machine-learning-approach-for-precipitation-nowc
pdf.

Huang-Chia Shih. A survey of content-aware video analysis for sports. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(5):1212–1231, 2017.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pp. 3104–3112, 2014.

# loop iterations	0	2	3
COCO	54.6	58.7	57.2
OC-val	35.1	37.8	30.0
OC-test	28.5	31.7	36.3

Table 5: Examining the influence of the number of iterations in the feedback loop. 0 indicates no loop is present.

Jonathan J Tompson, Arjun Jain, Yann LeCun, and Christoph Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *Advances in neural information processing systems*, pp. 1799–1807, 2014.

Oriol Vinyals, Samy Bengio, and Manjunath Kudlur. Order matters: Sequence to sequence for sets. *arXiv preprint arXiv:1511.06391*, 2015.

Francesco Visin, Kyle Kastner, Kyunghyun Cho, Matteo Matteucci, Aaron Courville, and Yoshua Bengio. Renet: A recurrent neural network based alternative to convolutional networks. *arXiv preprint arXiv:1505.00393*, 2015.

Francesco Visin, Marco Ciccone, Adriana Romero, Kyle Kastner, Kyunghyun Cho, Yoshua Bengio, Matteo Matteucci, and Aaron Courville. Reseg: A recurrent neural network-based model for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 41–48, 2016.

Shaofei Wang, Alexander Ihler, Konrad Kording, and Julian Yarkony. Accelerating dynamic programs via nested benders decomposition with application to multi-person pose estimation. In *The European Conference on Computer Vision (ECCV)*, September 2018.

Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *European Conference on Computer Vision (ECCV)*, 2018.

Song-Hai Zhang, Ruilong Li, Xin Dong, Paul Rosin, Zixi Cai, Xi Han, Dingcheng Yang, Haozhi Huang, and Shi-Min Hu. Pose2seg: Detection free human instance segmentation. In *CVPR*, June 2019a.

Yan Zhang, Jonathon Hare, and Adam Prügel-Bennett. Deep set prediction networks. *arXiv preprint arXiv:1906.06565*, 2019b.

A Appendix

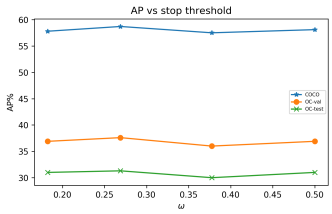
A.1 Additional ablative studies

Table 5 shows the advantage of having an feedback loop, where the performance with 3 loops slightly drops due to a larger cumulative gradient.
Table 6 shows the effect the in-image batch sizt T has on performance. $T = 1$ indicates cutting the dependency between each predicted pose in the image affects the performance very negatively.
In fig. 6, we plot the precision versus the stop threshold ω. During training, for too low a threshold, outliers may be penalized, whereas with too high a threshold, difficult cases like a blurred or low-resolution person can be missed. Too low a value, we would be penalizing outlier predictions, too high we would be missing potentially difficult cases like a low-resolution person. The results of a random permutation versus a learned permutation do not appear to be conclusive, as evident in table 7. For example, when increasing the number of hidden units of LSTM2, we still get comparable results to the LSTM with two hidden units, but in this setting, sometimes a random permutation performs better like in the experiment with 3 hidden units or with 4 hidden units on COCO. This indicates that there is no clear winner, unlike for instance an inverse permutation that would performs very bad in all settings.

T	1	2	4	6
COCO	36.6	58.3	57.6	58.7

Table 6: AP when varying the within-image batch size T.

Figure 6: Examining the effect of the stop threshold value during training.



A.2 COCO: Comparison with the baseline

We demonstrate the advantage of our approach over the baseline qualitatively in figure 5. We observe that in more challenging scenarios, the LSTM performs better at localizing joints and associating them. Such scenarios include overlapping people or occluded joints, in which the LSTM is more capable of reasoning about the occluded joints' locations. Additional challenging scenarios include crowded images with a cluttered background. False positives such as statues, human-shaped objects or people appearing in photos remain a challenge, but in one example in figure 5, we observe that the LSTM suppressed false positives in which AE could not, making the LSTM more robust to such outliers. It can be seen that oftentimes AE tends to overestimate the number of people resulting in redundant and invalid poses. In contrast, LSTM yields a much better estimate of the poses and their number.

A.3 OCHuman: Qualitative Results

In table 8, we present some qualitative results on OCHuman. Each estimated pose appears in a separate image for a clear distinction. As can be observed, this dataset contains extremely challenging cases with people overlapping or occluding each other to a high degree.

#hidden units-perm	2-learned	2-random	3-learned	3-random	4-learned	4-random
COCO	60.1	59.3	58.7	60.1	58.7	59.3
OC-val	39.4	39.1	36.1	38.7	39.2	37.3
OC-test	32.5	32.2	30.2	31.9	32.6	30.8

Table 7: Examining the influence of the number of hidden units on the performance with a learned and random permutation at LSTM2.

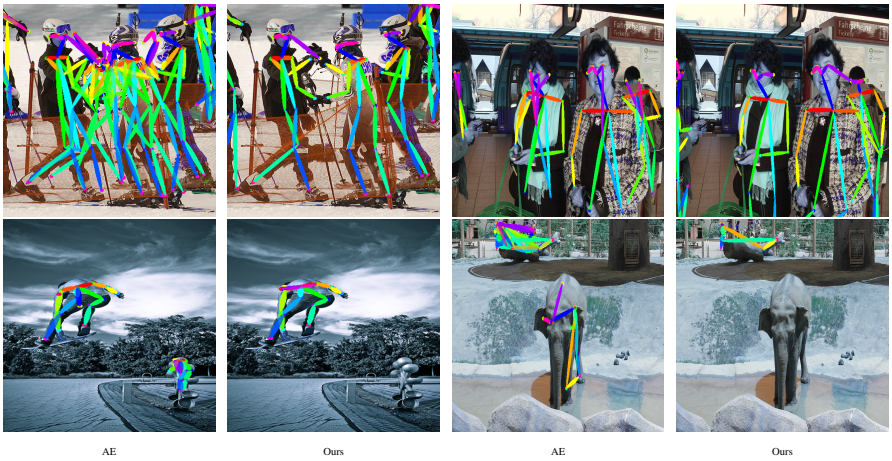


Figure 7: Qualitative comparison between Associative Embedding and our learning-based approach.

Figure 8: Qualitative results on OCHuman including some failure cases such as in the last row where the poses are mixed or slightly overestimated such as in the third row.