Direct Dense Pose Estimation

Liqian Ma[1] Lingjie Liu[2] Christian Theobalt[2] Luc Van Gool[1,3]

[1]KU-Leuven/PSI, Toyota Motor Europe (TRACE)
[2]Max Planck Institute for Informatics, Saarland Informatics Campus [3]ETH Zurich

{liqian.ma,luc.vangool}@esat.kuleuven.be
{theobalt, lliu}@mpi-inf.mpg.de vangool@vision.ee.ethz.ch

Abstract

Dense human pose estimation is the problem of learning dense correspondences between RGB images and the surfaces of human bodies, which finds various applications, such as human body reconstruction, human pose transfer, and human action recognition. Prior dense pose estimation methods are all based on Mask R-CNN framework and operate in a top-down manner of first attempting to identify a bounding box for each person and matching dense correspondences in each bounding box. Consequently, these methods lack robustness due to their critical dependence on the Mask R-CNN detection, and the runtime increases drastically as the number of persons in the image increases. We therefore propose a novel alternative method for solving the dense pose estimation problem, called Direct Dense Pose (DDP)*. DDP first predicts the instance mask and global IUV representation separately and then combines them together. We also propose a simple yet effective 2D temporal-smoothing scheme to alleviate the temporal jitters when dealing with video data. Experiments demonstrate that DDP overcomes the limitations of previous top-down baseline methods and achieves competitive accuracy. In addition, DDP is computationally more efficient than previous dense pose estimation methods, and it reduces jitters when applied to a video sequence, which is a problem plaguing the previous methods.*

1. Introduction

Human dense pose estimation aims to learn the correspondence between RGB images and 3D human model surfaces. It is a fundamental and essential problem in human-centric analysis and synthesis applications, such as 3D body shape estimation [53, 56], human pose transfer [11, 38], unselfie [30], and character animation [8, 46]. Güler *et al*. [1] densely map each person pixel of an ‘in the wild” RGB image to a location of a 3D human model surface using a two-stage top-down framework based on Mask R-CNN [14]. They sparsely annotate a human subset of the COCO dataset [27] for training. Follow-up works also adopt the same top-down principle to further improve the performance via uncertainty modeling [33], multi-scale strategy [13], knowledge-transfer [48], simulated data [57], or continuous embedding [32]. All these methods are built upon Mask R-CNN [14] framework and follow the two-stage top-down detect-then-segment pipeline. In particular, these top-down methods first employ an object detector based on Faster R-CNN [37] to predict a bounding box for each person, and then crop and resize regions-of-interest (ROIs) feature maps into a fixed size for each person. Such a top-down pipeline has several limitations. (1) The bounding box causes early commitment, *i.e*. if the bounding box fails to detect the whole body, there is no path to recovery, as shown in Fig. 1. Furthermore, several people may exist in the same bounding box which may not well align with a human due to overlap ambiguity (also mentioned in [48]), (2) The detected person patches are cropped and resized, thus the resolution per person is usually heavily compressed. (3) Top-down methods cannot fully leverage the sharing computation mechanism of convolutional networks, and thus, their inference time scales unfavorably with the number of instances (see Fig. 4).

In contrast, we propose an end-to-end direct method inspired by the success of direct instance segmentation methods [42, 49], which is faster and more accurate under multi-person occlusion than previous methods. In particular, we formulate the dense pose estimation task into two inter-related sub-tasks: global dense pose IUV representation [1] estimation and instance segmentation. Inferring IUVs in a direct approach avoids early commitment, overlapping ambiguity, and heavy resolution compression, as well as makes computational complexity less dependent on the instance

[1] The IUV representation, introduced by [1], is an image-based UV map with multiple channels.

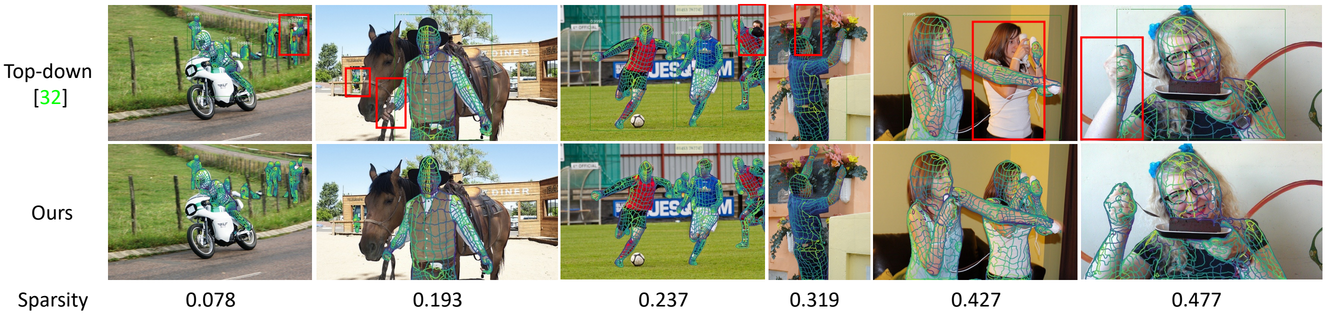


Figure 1: Top: illustration of top-down methods [32] early commitment and overlapping ambiguity issues (red boxes). Bottom: data sparsity, *i.e*. the ratio of human pixels over the whole image pixels.

number. Moreover, a simple fully convolutional implementation (as experimented in [1]) produces inferior results due to the interference from multi-person feature normalization and wastes much computation and GPU memory in the empty background area. To address this, we propose an instance-aware normalization (IAN) technique to improve the results and sparse residual FCN to save computation. Furthermore, previous methods are designed for per-frame prediction and produce flicker on video sequences. To alleviate flickering, we propose a simple and effective 2D temporal smoothing scheme which naturally fits our direct method.

We make three contributions: 1) We propose a direct framework DDP for human dense pose estimation, which runs faster and detects humans better in multiple person overlap situations than top-down methods. 2) We propose an instance-aware normalization (IAN) technique to remove the interference from multi-person feature normalization and utilize sparse convolution to skip the background computation. 3) We introduce an effective 2D temporal-smoothing scheme tailored for dense 2D inference problems, which preserves the coherence of projected 2D shapes.

2. Related Work

Instance segmentation. Instance segmentation methods combine instance-level object detection and pixel-level semantic segmentation. Most existing approaches can be categorized into top-down methods and bottom-up methods. Top-down methods [25, 14, 29, 3, 18, 5] operate in a detect-then-segment way, *i.e*. detect the bounding box and then segment the object within the box. Among these methods, Mask R-CNN [14] is the most widely known top-down framework and has seen improved variants [29, 18]. Due to the high performance of Mask R-CNN, all prior dense pose methods build upon it. Bottom-up methods [35, 7, 28] operate in a label-then-cluster way, *e.g*. learn pixel-level embeddings and then cluster them into groups. Recently, some effective direct methods [42, 49, 50] solved instance segmentation in one stage, *i.e*. without a detect-then-segment or label-then-cluster strategy. In their SOLO approach, Wang *et al*. [49] segment objects by locations, *i.e*. using instance location categorization to predict object center locations. They further improved accuracy and speed in their improved SOLOv2 method [50]. Tian *et al*. propose CondInst [42], which employs dynamic instance-aware networks to infer an instance mask from global feature maps directly. In this work, we take inspiration from CondInst [42] and propose a new direct framework for dense pose estimation.

Multi-person 2D pose estimation. Multi-person 2D pose estimation methods estimate the number of people, their positions, and their body keypoints (joints) from an image. Similar to instance segmentation methods, multi-person 2D pose estimation algorithms can be classified into top-down and bottom-up methods. Top-down methods [36, 52, 17, 39, 45, 16, 55] employ off-the-shelf person detectors to obtain a bounding box for each person and then apply a single-person pose estimator within each box. Bottom-up methods [4, 20, 19, 35, 24, 21, 6] first locate all the body joints in one image, and group them into individual person instances during post-processing. Both top-down and bottom-up methods require multiple steps to obtain the final keypoint detection results.

Recently, Tian *et al*. propose DirectPose [41] to handle the multi-person pose estimation. They extend the anchor-free single-stage object detector FCOS [43], with one new output branch for keypoint detection. Our method DDP follows a similar design principle to avoid the limitations of top-down methods. Note that, unlike sparse keypoints pose estimation, it is very challenging to formulate the dense correspondence prediction problem as a keypoint classification problem like [41]. Thus, we predict the instance mask and global IUV separately.

Dense human pose estimation. Some methods [2, 22] estimate pixel-wise correspondence to a 3D surface model by fitting a prior deformable surface model to the image via indirect supervision, *e.g*. through body silhouette and keyjoints. In [1], Güler *et al*. propose to directly map each pixel of a person region in the RGB image to the 3D human surface. The dense correspondence is represented in a chart-based format called IUV, *i.e*. 24 body surfaces plus one background category (25 classes in total). With each surface patch, the local correspondence is further represented by a local UV coordinate system. Therefore, their dense pose IUV representation has 25×3=75 dimensions, which are then summarized into three dimensions according to the 25 classes. [1] introduces a two-stage DensePose-RCNN built upon Mask R-CNN [14] and collects a large-scale image-to-surface sparse mapping dataset for direct supervision. DensePose-RCNN shows promising results on in-the-wild data. Following this pipeline, Guo *et al*. [13] propose a multi-scale method called AMANet with improved performance under scale variations. Wang *et al*. [48] improve dense pose estimation by utilizing external commonsense knowledge. Neverova *et al*. [34] propose to augment the data annotations with motion cues for performance improvement. Zhu *et al*. [57] introduce a new synthetic dense human pose dataset, together with a new estimator using a domain adaptation strategy to achieve good performance on real-world data. Recently, Neverova *et al*. [32] propose a more straightforward and universal representation, Continuous Surface Embeddings (CSE), to better represent dense correspondences and improve the performance. These works all follow the two-stage top-down strategy that suffers from early commitment, overlap ambiguities, heavy resolution compression, and unfavorable runtime scaling with the number of instances. In contrast, we propose a direct dense pose estimation framework with faster runtime and better resilience to the aforementioned challenges.

3. Framework

3.1. Overview

We propose a direct framework to divide the multi-person dense pose estimation problem into two subtasks: instance prediction and global IUV dense pose representation prediction. As illustrated in Fig. 2, our framework starts with a feature extractor backbone - a ResNet network (*e.g*. ResNet-50) followed by a feature pyramid network (FPN) [26]. Then, the extracted feature pyramid (1/4 to 1/32 scales) is aggregated via a feature aggregation (Sec. 3.2) module, and then fed into the instance branch (Sec. 3.3) and the global IUV branch (Sec. 3.4), respectively. The instance branch aims to predict the instance-level information, *i.e*. instance mask $\boldsymbol{M}_{ins}$ and dense pose mask $\boldsymbol{M}_{dp}$, for each person. In contrast, the global IUV branch aims to predict the dense pose IUV representation for the whole image. The final per-person dense pose estimation can be obtained simply by multiplying the dense pose mask $\boldsymbol{M}_{dp}$ with the global IUV representation in an element-wise way. In addition, we also propose a 2D temporal-smoothing scheme (Sec. 4) to alleviate the temporal jitters when dealing with video data.

3.2. Feature aggregation module

To alleviate instance scale variation issue in our direct method, we adopt a feature pyramid aggregation module [23] to aggregate features of different scales. Specifically, each FPN level is upsampled by convolutions and bilinear upsampling until it reaches 1/4 scale. These upscaled features are summed together as $\boldsymbol{X}_{agg}$ which are fed into our instance branch and global IUV branch, respectively.

3.3. Instance branch

The instance branch is built on top of a direct instance segmentation method CondInst [42] whose core idea is to dynamically generate the specific mask FCN head parameters for each instance[2]. For an image with K instances, K different mask FCN heads will be dynamically generated, each containing the characteristics of its target instance in the filters. In particular, as shown in Fig. 2 top, the instance branch first processes the received aggregated features $\boldsymbol{X}_{agg}$ with a Down Sample Conv module to extract global instance features $\boldsymbol{X}_D$ and reduce the feature resolution to 1/8 scale for saving computation. This Down Sample Conv module contains three convolution layers of stride 1 and one convolution layer with stride 2. Moreover, $\boldsymbol{X}_D$ is combined with a map of the coordinates, which are relative coordinates from all the locations on $\boldsymbol{X}_D$ to the location (h, w) (*i.e*. where the filters of the mask head are generated). Then, the combination, termed as $\tilde{\boldsymbol{X}}_D$, is sent to the mask FCN head, whose parameters $\theta_{h,w}$ is dynamically generated by the weight generator module, to predict the instance mask. In other words, $\tilde{\boldsymbol{X}}_D$ contains the global information for the whole image, while the parameters $\theta_{h,w}$ encode instance-aware characteristics (*e.g*. relative position, shape, and appearance). Different from CondInst, our mask FCN head will produce the instance mask $\boldsymbol{M}_{ins}$ and dense pose mask $\boldsymbol{M}_{dp}$ jointly for each instance. This joint learning design takes the advantages of the existing rich instance annotations for stabilizing training (explained in Sec. 5), as well as provides instance masks for the Global IUV branch (explained in Sec. 3.4). Note that one can also adopt other advanced direct instance segmentation networks for our instance branch, like SOLO [49] and SOLOv2 [50].

3.4. Global IUV branch

We aim to predict the dense pose IUV information globally where all IUVs of different people are represented in the same image plane. However, merely performing convolution over the large background region is computation-

[2] Similar to CondInst [42], there are centerness and box heads in our instance branch, but omitted in the figure for simplicity.

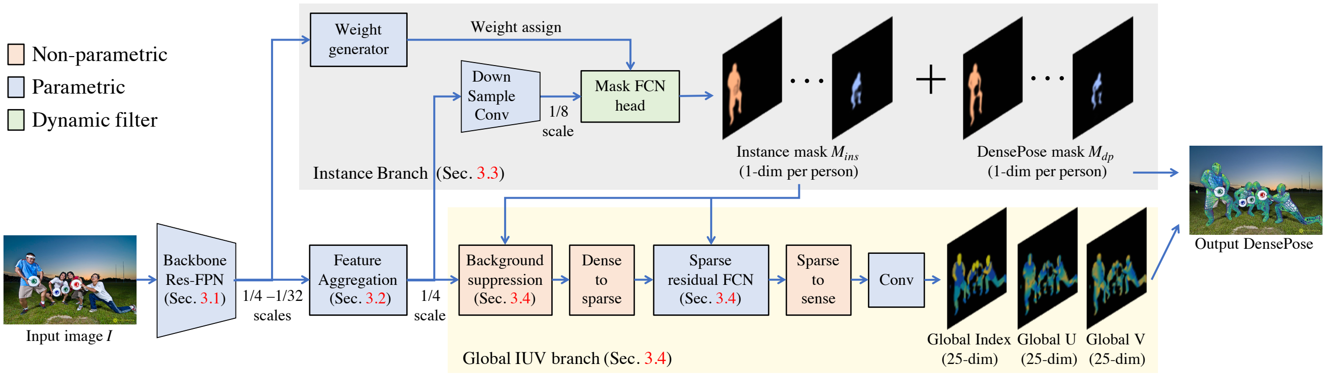

Non-parametric
Parametric
Dynamic filter
Weight generator
Weight assign
Down Sample Conv
1/8 scale
Mask FCN head
Instance mask M_{ins} (1-dim per person)
DensePose mask M_{dp} (1-dim per person)
Instance Branch (Sec. 3.3)
Output DensePose
Input image I
Backbone Res-FPN (Sec. 3.1)
1/4 –1/32 scales
Feature Aggregation (Sec. 3.2)
1/4 scale
Background suppression (Sec. 3.4)
Dense to sparse
Sparse residual FCN (Sec. 3.4)
Sparse to sense
Conv
Global Index (25-dim)
Global U (25-dim)
Global V (25-dim)
Global IUV branch (Sec. 3.4)

Figure 2: Framework overview. The ResNet-FPN based backbone is first used to extract a feature pyramid. Then, the feature aggregation module is applied to aggregate the feature pyramid into a global feature representation. Such a global feature is then fed into the instance branch and global IUV branch, respectively, to estimate the instance-level masks and the global IUV representation. Note that, for each instance, the Mask FCN weights are generated dynamically via a weight generator.

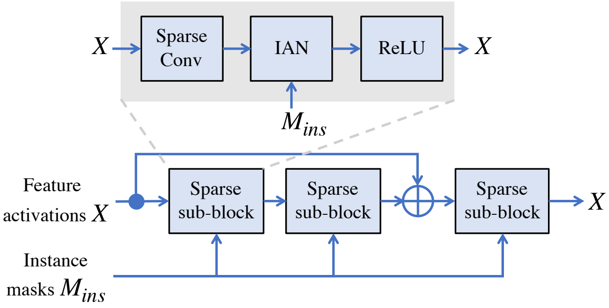


Figure 3: The sparse residual block contains three sparse sub-blocks. Each sparse sub-block consists of a sparse convolution layer [10], an instance-aware normalization (IAN) layer, and a ReLU layer.

ally quite expensive and wasteful. Therefore, we propose a background suppression operation followed by a sparse residual FCN module to only deal with the person region of interest. We also propose an instance-aware normalization (IAN) technique that performs better than normalizing all the instances together as done in [1].

Background suppression. As for the human dense pose estimation task, we are only interested in the people region, which usually occupies a small part of the whole image. For example, the sparsity (*i.e.* ratio of the people region over the whole image) on the DensePose-COCO dataset [1] is on average 23.7% and ranges from 1.7% to 90.9%[3]. Some sparsity examples are illustrated in Fig. 1. These large background regions will waste a large amount of computation and interfere with the learning procedure as well. Previous top-down multi-person dense pose estimation methods [1, 13, 33, 48, 57, 51, 32] alleviate this issue by cropping the feature maps according to the detected bounding boxes and process them separately in a single person style. Such a cropping-based approach is sub-optimal as it still introduces some background or content from other instances. Besides, this suffers from the early commitment issue. In contrast, we propose a background suppression strategy to suppress the background interference explicitly. Specifically, we first resize the estimated instance masks $\{\boldsymbol{M}^i_{ins}\}_{i=1,...,N}$ to the resolution of features $\boldsymbol{X}_{agg}$. The resized masks $\{\tilde{\boldsymbol{M}}^i_{ins}\}_{i=1,...,N}$ are then combined into a foreground mask $\boldsymbol{M}_{fg}$ which is applied to mask the features in a point-wise manner as follows,

$$\boldsymbol{X}_{fg} = \boldsymbol{M}_{fg} \odot \boldsymbol{X}_{agg}, \text{ where } \boldsymbol{M}_{fg} = \bigcup_{i=1}^{N} \tilde{\boldsymbol{M}}^i_{ins}, \qquad (1)$$

where $\boldsymbol{X}_{fg}$ denote the masked foreground features.

Sparse residual FCN. As mentioned above, computing the features on background regions is wasteful, especially considering that we need to maintain high-resolution features to achieve dense global IUV prediction. Therefore, we perform convolution in a sparse manner to reduce the computation and memory cost. Specifically, the dense features are first transformed into a sparse format and then processed by our sparse residual FCN module and finally transformed back to a dense format. The sparse residual FCN module consists of three sparse residual blocks. As illustrated in Fig. 3, each sparse residual block contains two sparse sub-blocks with a residual skip connection followed by a third sparse sub-block (except for the last sparse residual block). Each sparse sub-block includes a sparse convolution layer, an instance-aware normalization (IAN) layer, and a ReLU layer. Regarding the sparse convolution layer, we adopt the submanifold sparse convolution (SSC) layer [10],

[3] Calculated on the instance annotations of DensePose-COCO minival split.

which can fix the location of active sites and thus maintain the same level of sparsity throughout the network. The instance-aware normalization (IAN) layer is described in the next paragraph.

Instance-aware normalization. We propose the instance-aware normalization to perform feature normalization for each instance separately. In particular, we integrate the resized instance masks $\{\tilde{\boldsymbol{M}}_{ins}\}_{i=1,...,N}$ into our normalization operation as follows,

$$\mu_{nci} = \frac{1}{\text{Numel}(\tilde{\boldsymbol{M}}^i_{ins})} \sum_{h,w \in \tilde{\boldsymbol{M}}^i_{ins}} x_{nchw}, \tag{2}$$

$$\sigma^2_{nci} = \frac{1}{\text{Numel}(\tilde{\boldsymbol{M}}^i_{ins})} \sum_{h,w \in \tilde{\boldsymbol{M}}^i_{ins}} (x_{nchw} - \mu_{nci})^2, \tag{3}$$

$$\hat{x} = \bigcup_{i=1}^{N} \bigcup_{h,w \in \tilde{\boldsymbol{M}}^i_{ins}} \Big(\frac{x_{nchw} - \mu_{nci}}{\sqrt{\sigma^2_{nci} + \epsilon}} \gamma + \beta \Big), \tag{4}$$

where x_{nchw} denotes the feature point at location (h,w) exists in $\tilde{\boldsymbol{M}}^i_{ins}$, *i.e.* the region of the i-th instance. Numel$(\cdot)$ means the point number in the region of the i-th instance. We calculate the mean μ_{nci} and variance σ^2_{nci} for each instance mask $\tilde{\boldsymbol{M}}^i_{ins}$ individually, and then apply instance normalization [44] due to its better performance over other normalization techniques in our pilot experiment. Note that, all the instances share the same learnable affine parameters γ and β as they all belong to the person category.

4. 2D temporal-smoothing scheme

Previous dense pose estimation methods are designed for image-based prediction because it is expensive to collect accurate dense pose annotations for video data. However, directly applying image-based methods to video data leads to the undesirable flickering issue, since the temporal information is not considered. To address this, we introduce a simple and effective 2D temporal-smoothing scheme, which fits well our global IUV representation. The main idea is to use the temporal constraint from the original RGB video. Specifically, given the present RGB frame I_t and its temporally adjacent frames $\{I_{t+j}\}_{j=-r,\cdots,-1,1,\cdots,r}$, we adopt an optical flow estimation model (*e.g.* RAFT [40]) to predict the optical flows $\{f_{t\to t+j}\}_{j=-r,\cdots,-1,1,\cdots,r}$, which are used to warp the global IUV dense pose representation $\boldsymbol{C}$ of adjacent frames to one $\boldsymbol{C}_{temp}$ for the present frame using the following weighted sum:

$$\boldsymbol{C}_{t+j\to t} = \text{Warp}(\boldsymbol{C}_{t+j}, f_{t\to t+j}), \tag{5}$$

$$\boldsymbol{C}_{temp} = \sum_{j=-r}^{r} \alpha_j \boldsymbol{C}_{t+j\to t}, \tag{6}$$

where r is the temporal window interval and α_j is the sum weight. We set $r = 2$ and $\{\alpha_j\} = [0.2, 0.2, 0.2, 0.2, 0.2]$ by default. Here we perform the warping and weighted sum operations on the continuous logit-level, *i.e.* $\boldsymbol{C}$ is a 75-dim logit representation.

5. Learning

As mentioned before, we predict instance mask $\boldsymbol{M}_{ins}$ and dense pose mask $\boldsymbol{M}_{dp}$ jointly in our instance branch (Sec. 3.3). We take this joint approach because of the limited dense pose annotation and the limited body coverage. Regarding the annotation, only part of human instances was selected for dense pose annotation on the DensePose-COCO dataset. Training with annotated dense pose masks only will lead to inferior instance prediction performance. Furthermore, the dense pose masks do not cover the whole person, leading to important information missing in our background suppression. (Sec. 3.4). Therefore, we jointly regress both instance masks and dense pose masks in our instance branch to improve the learning with rich instance annotations and suppress the background with instance masks. Besides, we apply the ground truth instance masks for background suppression in our global IUV branch during training, considering that the estimated instance masks contain many errors in the early training stage. Such errors will break the learning of the global IUV branch.

Losses. The overall loss function of DDP is formulated as,

$$L_{all} = L_{mask} + L_{IUV}, \tag{7}$$

$$L_{mask} = L_{fcos} + \lambda_1(L_{M_{ins}} + L_{M_{dp}}) \tag{8}$$

$$L_{IUV} = L_I + \lambda_2 L_{UV} + \lambda_3 L_s \tag{9}$$

where L_{mask} and L_{IUV} denote the loss for instance branch and global IUV branch, respectively. $\lambda_1 = 5$, $\lambda_2 = 10$, $\lambda_3 = 1$ are used to balance the losses. Similar to CondInst, our instance prediction loss L_{mask} includes L_{fcos} and two mask losses $L_{M_{ins}}$, $L_{M_{dp}}$. We refer the reader to FCOS [43] for the details of L_{fcos}. $L_{M_{ins}}$ and $L_{M_{dp}}$ are defined as,

$$L_{M_{ins}} = \frac{1}{N_{pos}} \sum_{h,w} \mathbb{1}_{\{c^*_{h,w}>0\}} L_{dice}(\boldsymbol{M}^{h,w}_{ins}, \boldsymbol{M}^{h,w*}_{ins}), \tag{10}$$

$$L_{M_{dp}} = \frac{1}{N_{pos}} \sum_{h,w} \mathbb{1}_{\{c^*_{h,w}>0\}} L_{dice}(\boldsymbol{M}^{h,w}_{dp}, \boldsymbol{M}^{h,w*}_{dp}), \tag{11}$$

where $[\boldsymbol{M}^{h,w}_{ins}, \boldsymbol{M}^{h,w}_{dp}] = \text{MaskHead}(\tilde{\boldsymbol{X}}_D; \theta_{h,w})$ are the estimated instance and dense pose masks, and $\boldsymbol{M}^{h,w*}_{ins}, \boldsymbol{M}^{h,w*}_{dp}$ are the corresponding ground truth. The $c^*_{h,w}$ is the classification label of location (h, w), which is the class of the instance associated with the location or 0 (*i.e.* background) if the location is not associated with any instance. N_{pos} is the number of locations where $c^*_{h,w} > 0$. Our global dense pose IUV prediction loss L_{IUV} contains

Method	AP	AP$_{50}$	AP$_{75}$	AP$_M$	AP$_L$	AR	AR$_{50}$	AR$_{75}$	AR$_M$	AR$_L$	time (s)
Top-down methods											
DP-cascade (Res50) [1]	51.6	83.9	55.2	41.9	53.4	60.4	88.9	65.3	43.3	61.6	0.583
DP-cascade (Res50)+masks [1]	52.8	85.5	56.1	40.3	54.6	62.0	89.7	67.0	42.4	63.3	-
DP-cascade (Res50)+keypoints [1]	55.8	87.5	61.2	48.4	57.1	63.9	91.0	69.7	50.3	64.8	-
Parsing (Res50) [54]	55.0	87.6	59.8	50.6	56.6	-	-	-	-	-	0.392
Parsing (Res50)+keypoints [54]	58.3	90.1	66.9	51.8	61.9	-	-	-	-	-	-
Parsing (ResNeXt101) [54]	59.1	91.0	66.9	51.8	61.9	-	-	-	-	-	-
Parsing (ResNeXt101)+keypoints [54]	61.6	91.6	72.3	54.8	64.8	-	-	-	-	-	-
SimPose (Res50)* [57]	57.3	88.4	67.3	60.1	59.3	66.4	95.1	77.8	62.4	66.7	-
AMA-Net (Res50) [13]	64.1	91.4	72.9	59.3	65.3	71.6	94.7	79.8	61.3	72.3	-
KTN (Res50) [48]	66.5	91.5	75.5	61.9	68.0	74.2	95.2	82.3	**64.2**	**74.9**	0.518
DP R-CNN DeepLab (Res50) [32]	66.8	92.8	**79.7**	60.7	68.0	72.1	95.8	82.9	62.2	72.4	0.242
DP R-CNN DeepLab (Res101) [32]	**67.7**	**93.5**	**79.7**	**62.6**	**69.1**	**73.6**	**96.5**	**84.7**	**64.2**	74.2	0.382
Direct methods											
Ours (Res50)	64.0	92.4	76.0	57.2	65.7	70.9	96.4	82.4	59.9	71.6	**0.209**

Table 1: The quantitative results on DensePose-COCO minival split. * Models are trained with a simulated dataset.

one cross-entropy loss L_I for body parts classification, one smooth L1-loss L_{UV} for local UV coordinates regression, and one smoothing loss L_s. The smoothing loss L_s is used to encourage the model to produce less noisy IUV representation, since the dense pose annotation is a set of sparse points. In particular, we adopt an edge-aware smoothness regularization [9] as follows,

$$L_s = \frac{1}{N}\sum_{i=1}^{N} |\nabla_h \boldsymbol{C}| e^{-|\nabla_h \boldsymbol{M}_{ins}^i|} + |\nabla_w \boldsymbol{C}| e^{-|\nabla_w \boldsymbol{M}_{ins}^i|}, \tag{12}$$

where $\boldsymbol{C}$ is the predicted global IUV representation.

6. Experiments

For evaluation, we present qualitative and quantitative results, a runtime analysis, and user study results.

Implementation details. Unless specified otherwise, we use the following implementation details. We use a ResNet-50 [15] architecture as the backbone, followed by a 4-level FPN (*i.e*. 1/4, 1/8, 1/16, 1/32 levels). The ResNet weights are initialized by the pre-trained keypoints estimation models from COCO [27]. Our models are trained with stochastic gradient descent (SGD) for 130K iterations with an initial learning rate of 0.01 and a mini-batch of 8 images. The learning rate is reduced by a factor of 10 at iteration 100K and 120K, respectively. Weight decay and momentum are set as 0.0001 and 0.9, respectively. Following [1], two quantitative metrics are used for evaluation, *i.e*. Average Precision (AP) and Average Recall (AR). Both metrics are calculated at a number of geodesic point similarity (GPS) ranging from 0.5 to 0.95. In addition, other evaluation metrics, *i.e*. AP$_M$ and AR$_M$ for medium people and AP$_L$ and AR$_L$ for large people, are also reported. Our method is implemented based on the Detectron2 framework [51]. No data augmentation is used during training or testing. Inference times are measurd on a single V100 GPU with one image per batch.

Dataset. Our method is evaluated on DensePose-COCO dataset [1], which has manually annotated correspondences on a subset of the COCO dataset [27]. There are about 50K labeled human instances each of which is annotated with 100 points on average. In total, there are about 5 million manually annotated correspondences. The dataset is split into a training set and a validation set with 32K images and 1.5k images, respectively.

6.1. Comparisons

We evaluate the proposed end-to-end DDP framework on DensePose-COCO minival split and compare it with the SOTA top-down methods in Table 1. Our method achieves 64.0% AP and 70.9% AR with ResNet-50. The performance of our model is better than the strong baselines DP-cascade [1] (64.0% vs. 55.8% AP) and Parsing [54] (64.0% vs. 61.6% AP), and is comparable to AMA-Net [13] (64.0% vs. 64.1% AP). Our model is still behind the top-performing top-down methods KTN [48] and DP R-CNN DeepLab [32]. Nevertheless, it is worth noting that the top-down strategy suffers from the early commitment issue, overlap ambiguities as shown in Fig. 1. Besides, top-down methods run slower in larger multi-person scenes and compress image resolution heavily.

Method	AP	AP$_{50}$	AP$_{75}$	AP$_M$	AP$_L$	AR	AR$_{50}$	AR$_{75}$	AR$_M$	AR$_L$	time (s)
Ours (Res50) w/o L_s, w/o IAN, w/o Sparse	62.2	91.5	73.4	55.9	64.0	69.3	95.9	80.3	58.2	70.1	0.380
Ours (Res50) w/o L_s, w/o IAN	62.2	92.3	73.2	55.0	63.9	69.3	96.2	80.4	58.9	70.0	0.234
Ours (Res50) w/o L_s	63.8	91.8	75.9	57.1	**65.8**	**71.2**	96.2	**83.4**	59.0	**71.9**	0.211
Ours (Res50)	**64.0**	**92.4**	**76.0**	**57.2**	65.7	70.9	**96.4**	82.4	**59.9**	71.6	**0.209**

Table 2: Ablation study and average inference time on DensePose-COCO minival split.

Method	Human preference
DP R-CNN DeepLab (Res50) [32]	0.076
Ours (Res50) w/ temporal smoothing	0.924
Ours (Res50) w/o temporal smoothing	0.016
Ours (Res50) w/ temporal smoothing	0.984

Table 3: User study for temporal smoothing.

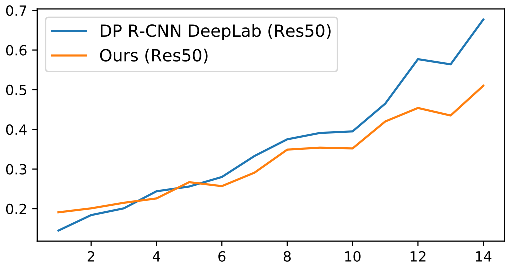
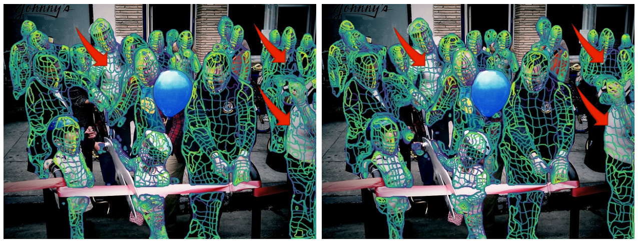


Figure 4: Inference time. X-axis: number of person instances on each image. Y-axis: average inference time in seconds for each image.



Ours (Res50) w/o L_s

Ours (Res50)

Figure 5: Effectiveness of smoothing loss L_s.

6.2. Ablation study and inference time

We perform an ablation study for each component of DDP presented in Table 2. The ablative settings are as follows. **w/o** L_{smooth}: Removing the smoothing loss. **w/o IAN**: Removing the instance-aware normalization. **w/o Sparse**: We use full dense feature maps instead of sparse feature maps, *i.e.* $\boldsymbol{M_{fg}}$ is a tensor with all elements set to one. We observe that the proposed IAN technique improves the results, and the sparse technique aids to significantly reduce the computation time. Regarding the smoothing loss L_s, the quantitative results are almost the same, while the qualitative results are spatially smoothed. As shown in the isocontour visualization in Fig. 5, the isocontour becomes less noisy when applying the smoothing loss L_s. Such smoothness is desirable because of the continuity nature of human body surface. Furthermore, we also compare the inference time on images of different instance numbers with the state-of-the-art top-down method DP R-CNN DeepLab [32] as shown in Fig. 4. We observe that our method's inference time increases more slowly with the number of people in the scene, because our approach's inference time mainly depends on the image's sparsity.

6.3. Temporal smoothing evaluation

User study. To demonstrate the effectiveness of our 2D temporal-smoothing scheme, we perform a user study on 11 YouTube video sequences of 20 seconds each. As shown in Table 3, we ask 30 users to assess the temporal smoothing of the results of two comparisons: (1) our method versus the top-performing top-down method - DP R-CNN DeepLab (Res50) [32] (row 1-2), and (2) our method versus our non-smoothing setting (row 3-4). For each video sequence, given two randomly ordered video results of dense pose isocontour visualization from two methods, users are asked to pick one that looks temporally smoother than the other. The results in Table 3 illustrate that our method is obviously preferred over the alternative.

Quantitative evaluation. Furthermore, we also perform a quantitative evaluation measured with the most widely used metrics in the video stabilization field: Interframe Transformation Fidelity (ITF) index [31] and Interframe Similarity Index (ISI) [12], which are based on the video interframe PSNR and SSIM scores, respectively. We report the average scores (see Tab. 4) of 11 isocontour visualization videos used in the user study. We can see that our temporal smoothing strategy improves the temporal smoothness on both evaluation metrics.

Image-to-image translation. We also apply our method on the human pose transfer task, *i.e.* translate the dense pose IUV representation input to an RGB image. We adopt the popular pix2pixHD [47] as our translation model and gener-

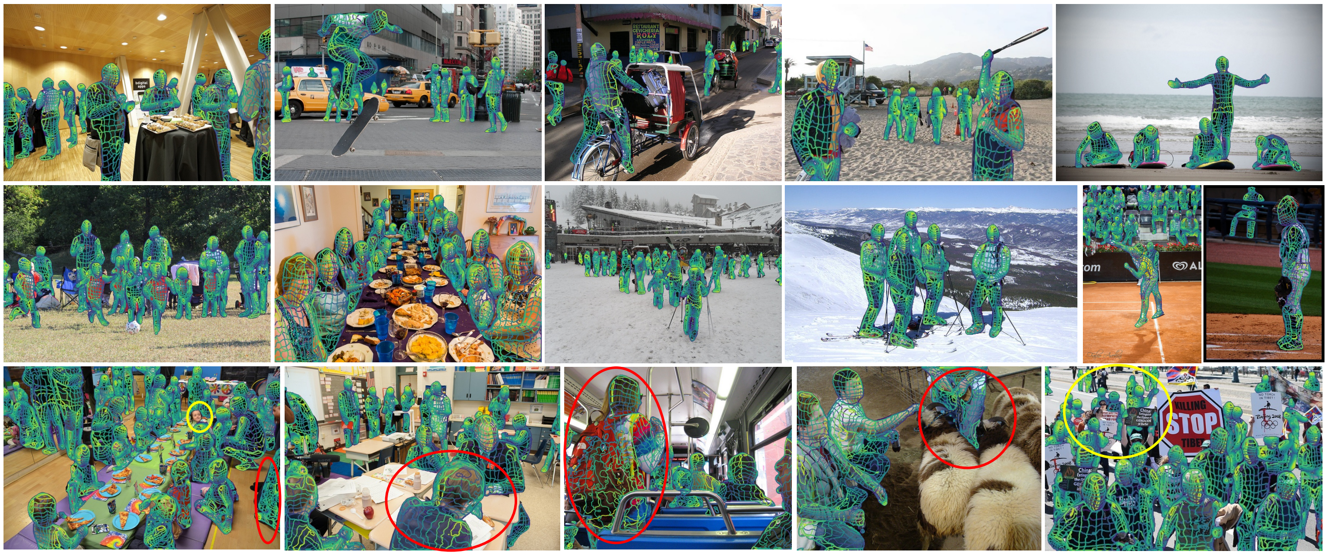


Figure 6: Qualitative results on DensePose-COCO minival split. The red and yellow circles spot the failure cases (see Sec. 7).

Model	ITF	ISI
DP R-CNN DeepLab (Res50) [32]	35.04	0.9202
Ours (Res50) w/o temporal smoothing	35.24	0.9253
Ours (Res50) w/ temporal smoothing	**35.68**	**0.9294**

Table 4: Quantitative evaluation for temporal smoothing.

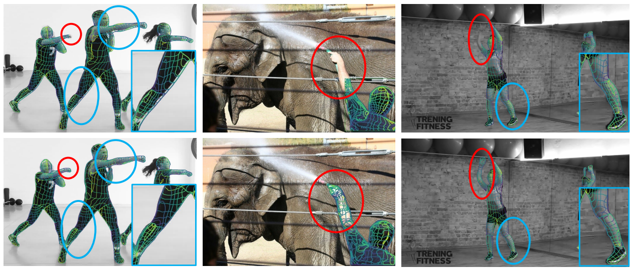


Figure 7: Results of [32] (top) and Ours (bottom) with Res50. The red and blue circles spot our advantages.

ate the image frame-by-frame. We observe that our temporally smooth dense pose can help alleviate video flickering issue and generate stable video results. Please refer to our supplementary video for the visualization results.

7. Qualitative results and discussion

In Fig. 7, we show the advantages of our method compared to the top-down methods [32] of avoiding early commitment (red circles) and better boundaries (blue circles). In Fig. 6, we illustrate more isocontour visualization results. We observe that our DDP can predict smooth and accurate 3D body correspondences in diverse real-world scenarios exhibiting different challenges, *e.g*. illumination variations (indoor and outdoor), occlusions (self occlusion, inter occlusion, and background occlusion), diverse body poses and views. Although our method achieves comparable performance to SOTA methods, it may produce noisy results for large-scale or occluded people (*e.g*. the red circle in Fig. 6) and fail to detect some occluded small people (*e.g*. the yellow circle in Fig. 6). As for our 2D temporal-smoothing scheme, it can alleviate the temporal flickering for most cases, but may fail in textureless region or varying illumination where the optical flow estimation is not robust.

8. Conclusion

We present a direct human dense pose estimation method DDP that deals with the multi-person dense pose estimation via two inter-related sub-tasks: global IUV estimation and instance segmentation. The proposed method achieves comparable results to strong top-down benchmark methods, and avoids issues like early commitment and overlap ambiguity. Our method also runs in weak instance number dependent time. Furthermore, we introduce a temporal smoothing pose-processing which naturally fits the global IUV representation and enables temporal smoothing dense pose estimation. We believe such a temporal smoothing ability can benefit human analysis and synthesis tasks.

Acknowledgements. Liqian Ma and Luc Val Gool were supported by Toyota Motors Europe. Christian Theobalt was supported by ERC Consolidator Grant 4DReply (770784). Lingjie Liu was supported by Lise Meitner Postdoctoral Fellowship.

References

[1] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *CVPR*, 2018. 4321, 4322, 4323, 4324, 4326

[2] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *ECCV*, 2016. 4322

[3] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: Real-time instance segmentation. In *ICCV*, 2019. 4322

[4] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017. 4322

[5] Xinlei Chen, Ross Girshick, Kaiming He, and Piotr Dollár. Tensormask: A foundation for dense object segmentation. In *CVPR*, 2019. 4322

[6] Bowen Cheng, Bin Xiao, Jingdong Wang, Honghui Shi, Thomas S Huang, and Lei Zhang. Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation. In *CVPR*, 2020. 4322

[7] Bert De Brabandere, Davy Neven, and Luc Van Gool. Semantic instance segmentation with a discriminative loss function. In *CVPRw*, 2017. 4322

[8] Oran Gafni, Lior Wolf, and Yaniv Taigman. Vid2game: Controllable characters extracted from real-world videos. In *ICLR*, 2020. 4321

[9] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *CVPR*, 2017. 4326

[10] Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *CVPR*, 2018. 4324

[11] Artur Grigorev, Artem Sevastopolsky, Alexander Vakhitov, and Victor Lempitsky. Coordinate-based texture inpainting for pose-guided human image generation. In *CVPR*, 2019. 4321

[12] Wilko Guilluy, Azeddine Beghdadi, and Laurent Oudre. A performance evaluation framework for video stabilization methods. In *European Workshop on Visual Information Processing (EWVIP)*, 2018. 4327

[13] Yuyu Guo, Lianli Gao, Jingkuan Song, Peng Wang, Wuyuan Xie, and Heng Tao Shen. Adaptive multi-path aggregation for human densepose estimation in the wild. In *ACM MM*, 2019. 4321, 4323, 4324, 4326

[14] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 4321, 4322, 4323

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4326

[16] Junjie Huang, Zheng Zhu, Feng Guo, and Guan Huang. The devil is in the details: Delving into unbiased data processing for human pose estimation. In *CVPR*, 2020. 4322

[17] Shaoli Huang, Mingming Gong, and Dacheng Tao. A coarse-fine network for keypoint localization. In *ICCV*, 2017. 4322

[18] Zhaojin Huang, Lichao Huang, Yongchao Gong, Chang Huang, and Xinggang Wang. Mask scoring r-cnn. In *CVPR*, 2019. 4322

[19] 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. 4322

[20] Eldar Insafutdinov, Leonid Pishchulin, Bjoern Andres, Mykhaylo Andriluka, and Bernt Schiele. Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In *ECCV*, 2016. 4322

[21] Sheng Jin, Wentao Liu, Enze Xie, Wenhai Wang, Chen Qian, Wanli Ouyang, and Ping Luo. Differentiable hierarchical graph grouping for multi-person pose estimation. In *ECCV*, 2020. 4322

[22] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, 2018. 4322

[23] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *CVPR*, 2019. 4323

[24] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. Pifpaf: Composite fields for human pose estimation. In *CVPR*, 2019. 4322

[25] Yi Li, Haozhi Qi, Jifeng Dai, Xiangyang Ji, and Yichen Wei. Fully convolutional instance-aware semantic segmentation. In *CVPR*, 2017. 4322

[26] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 4323

[27] 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 *ECCV*, 2014. 4321, 4326

[28] Shu Liu, Jiaya Jia, Sanja Fidler, and Raquel Urtasun. Sgn: Sequential grouping networks for instance segmentation. In *ICCV*, 2017. 4322

[29] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *CVPR*, 2018. 4322

[30] Liqian Ma, Zhe Lin, Connelly Barnes, Alexei A Efros, and Jingwan Lu. Unselfie: Translating selfies to neutral-pose portraits in the wild. In *ECCV*, 2020. 4321

[31] Lucio Marcenaro, Gianni Vernazza, and Carlo S Regazzoni. Image stabilization algorithms for video-surveillance applications. In *ICIP*, 2001. 4327

[32] Natalia Neverova, David Novotny, Vasil Khalidov, Marc Szafraniec, Patrick Labatut, and Andrea Vedaldi. Continuous surface embeddings. In *NeurIPS*, 2020. 4321, 4322, 4323, 4324, 4326, 4327, 4328

[33] Natalia Neverova, David Novotny, and Andrea Vedaldi. Correlated uncertainty for learning dense correspondences from noisy labels. In *NeurIPS*, 2019. 4321, 4324

[34] Natalia Neverova, James Thewlis, Riza Alp Guler, Iasonas Kokkinos, and Andrea Vedaldi. Slim densepose: Thrifty learning from sparse annotations and motion cues. In *CVPR*, 2019. 4323

[35] Alejandro Newell, Zhiao Huang, and Jia Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *NeurIPS*, 2017. 4322

[36] George Papandreou, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, and Kevin Murphy. Towards accurate multi-person pose estimation in the wild. In *CVPR*, 2017. 4322

[37] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE TPAMI*, 39(6):1137–1149, 2017. 4321

[38] Kripasindhu Sarkar, Dushyant Mehta, Weipeng Xu, Vladislav Golyanik, and Christian Theobalt. Neural rerendering of humans from a single image. In *ECCV*, 2020. 4321

[39] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, 2019. 4322

[40] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 2020. 4325

[41] Zhi Tian, Hao Chen, and Chunhua Shen. Directpose: Direct end-to-end multi-person pose estimation. *arXiv preprint arXiv:1911.07451*, 2019. 4322

[42] Zhi Tian, Chunhua Shen, and Hao Chen. Conditional convolutions for instance segmentation. In *ECCV*, 2020. 4321, 4322, 4323

[43] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *ICCV*, 2019. 4322, 4325

[44] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016. 4325

[45] Jian Wang, Xiang Long, Yuan Gao, Errui Ding, and Shilei Wen. Graph-pcnn: Two stage human pose estimation with graph pose refinement. In *ECCV*, 2020. 4322

[46] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. In *NeurIPS*, 2018. 4321

[47] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *CVPR*, 2018. 4327

[48] Xuanhan Wang, Lianli Gao, Jingkuan Song, and Heng Tao Shen. Ktn: Knowledge transfer network for multi-person densepose estimation. In *ACM MM*, 2020. 4321, 4323, 4324, 4326

[49] Xinlong Wang, Tao Kong, Chunhua Shen, Yuning Jiang, and Lei Li. Solo: Segmenting objects by locations. In *ECCV*, 2020. 4321, 4322, 4323

[50] Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li, and Chunhua Shen. Solov2: Dynamic and fast instance segmentation. In *NeurIPS*, 2020. 4322, 4323

[51] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019. 4324, 4326

[52] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *ECCV*, 2018. 4322

[53] Yuanlu Xu, Song-Chun Zhu, and Tony Tung. Denserac: Joint 3d pose and shape estimation by dense render-and-compare. In *ICCV*, 2019. 4321

[54] Lu Yang, Qing Song, Zhihui Wang, and Ming Jiang. Parsing r-cnn for instance-level human analysis. In *CVPR*, 2019. 4326

[55] Feng Zhang, Xiatian Zhu, Hanbin Dai, Mao Ye, and Ce Zhu. Distribution-aware coordinate representation for human pose estimation. In *CVPR*, 2020. 4322

[56] Hongwen Zhang, Jie Cao, Guo Lu, Wanli Ouyang, and Zhenan Sun. Learning 3d human shape and pose from dense body parts. *IEEE TPAMI*, 2020. 4321

[57] Tyler Zhu, Per Karlsson, and Christoph Bregler. Simpose: Effectively learning densepose and surface normals of people from simulated data. In *ECCV*, 2020. 4321, 4323, 4324, 4326