Survey

Outline

Method	Conf	Pose 2D	Pose 3D	Dense Pose	Parsing	Images	Videos	
MMAN [1]	ECCV2018				√	√		GAN
PGN [2]	ECCV2018				√	√		New dataset
MuLA[3]	ECCV2018	√			√	√		dynamic filters
DensePose [4]	CVPR2018			√		√		
DLCM [5]	ECCV2018	√				√		compositional model
Compositional[6]	CVPR2018	√	√			√		
PoseNet[7]	ECCV2018		√			√	√	
SMPL[8]	ECCV2016		√			√		
Wei Yang [9]	CVPR2018		√			√		GAN
Xuecheng Nie [10]	CVPR2018	√			√	√		dynamic filters
Hossain [11]	ECCV2018		√				√	2D->3D, only coordinates
Xiao Sun [12]	ECCV2018	√	√			√		
Lipeng Ke [13]	ECCV2018	√				√		structure-aware loss
Bin Xiao[14]	ECCV2018	√				√	√	

[1] Macro-Micro Adversarial Network for Human Parsing
[2] Instance-level Human Parsing via Part Grouping Network
[3] Mutual Learning to Adapt for Joint Human Parsing and Pose Estimation
[4] DensePose: Dense Human Pose Estimation In TheWild
[5] Deeply Learned Compositional Models for Human Pose Estimation
[6] Compositional Human Pose Regression
[7] Learning 3D Human Pose from Structure and Motion

[8] Keep it smpl: Automatic estimation of 3d human pose and shape from a single image
[9] 3D Human Pose Estimation in the Wild by Adversarial Learning
[10] Human Pose Estimation with Parsing Induced Learner
[11] Exploiting temporal information for 3D human pose estimation
[12] Integral Human Pose Regression
[13] Multi-Scale Structure-Aware Network for Human Pose Estimation
[14] Simple Baselines for Human Pose Estimation and Tracking

3D Human Pose

Defination

Notations

A 3D human pose P = {p1, p2, . . . , pk} is defined by the positions of k = 16 body joints in Euclidean space. These joint positions are defined relative to a root joint, which is fixed as the pelvis.

The input to the pose estimation system could be a single RGB image or a continuous stream of RGB images I = {. . . , Ii−1, Ii}. The i^{th} joint pi is the coordinate of the joint in a 3D Euclidean space i.e. p_i = (p^x_i , p^y_i , p^z_i).

An inferred joint will be denoted as ˜p and ground-truth as ˆp.

The 2D pose can be expressed with only the x,y-coordinates and denoted as pxy = (p^x, p^y); the depth-only joint location is denoted as p^z = (p^z).

The ith training data from a 3D annotated dataset consists of an image I_i and corresponding joint locations in 3D, ˆ P_i. On the other hand, the 2D data has only the 2D joint locations, ˆ P^{xy}_i .

A weakly-supervised framework

The weakly-supervised framework for joint learning from [41].
[41] X. Zhou, Q. Huang, X. Sun, X. Xue, and Y. Wei. Towards 3d human pose estimation in the wild: A weakly-supervised approach. In ICCV, 2017.

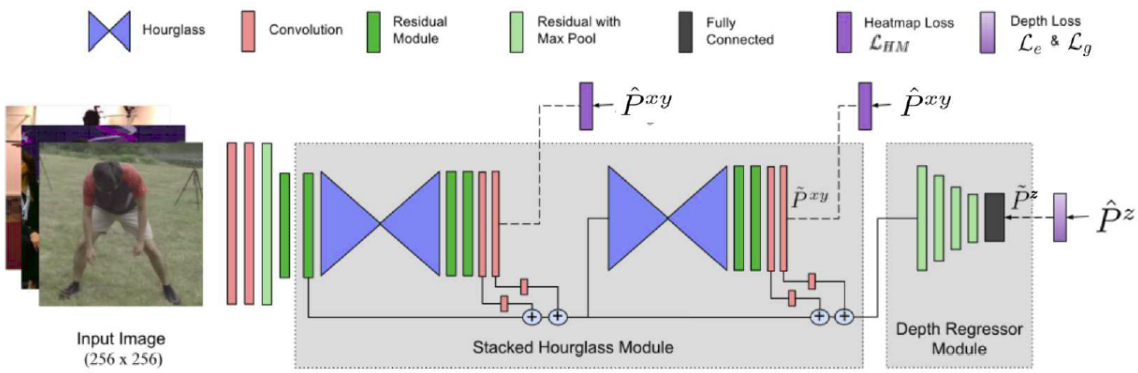


Fig. 1. A schematic of the network architecture. The stacked hourglass module is trained using the standard Euclidean loss $\mathcal{L}_{HM}$ against ground truth heatmaps. Whereas, the depth regressor module is trained on either $\mathcal{L}^z_{3D}$ or $\mathcal{L}^z_{2D}$ depending on whether the ground truth depth $\hat{P}^z$ is available or not.

The stacked hourglass architecture [25] for 2D pose estimation.

[25] A. Newell, K. Yang, and J. Deng. Stacked hourglass networks for human pose estimation. In ECCV, 2016.

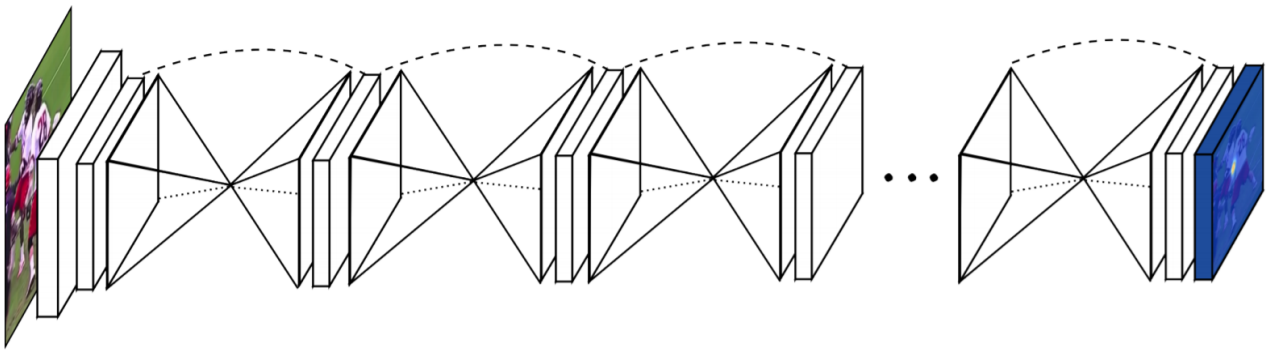


Fig. 1. Our network for pose estimation consists of multiple stacked hourglass modules which allow for repeated bottom-up, top-down inference.

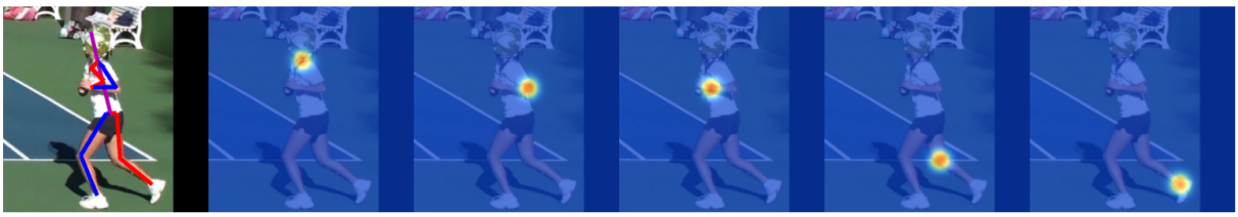


Fig. 2. Example output produced by our network. On the left we see the final pose estimate provided by the max activations across each heatmap. On the right we show sample heatmaps. (From left to right: neck, left elbow, left wrist, right knee, right ankle)

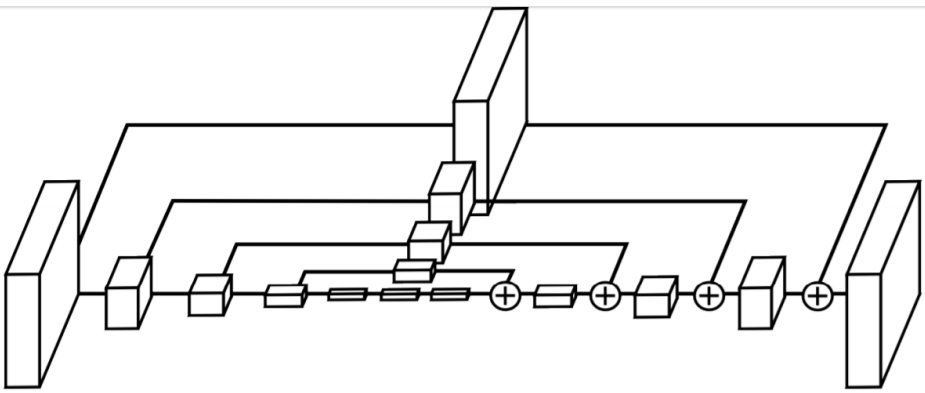


Fig. 3. An illustration of a single "hourglass" module. Each box in the figure corresponds to a residual module as seen in Figure 4. The number of features is consistent across the whole hourglass.

Related work

A detailed review of the literature [32]

[32] N. Sarafianos, B. Boteanu, B. Ionescu, and I. A. Kakadiaris. 3d human pose estimation: A review of the literature and analysis of covariates. Computer Vision and Image Understanding, 2016.

ConvNet architectures

Most existing ConvNet based approaches either directly regress 3D poses from the input image [34,17,42,43] or infer 3D from 2D pose in a twostage approach [35,41,23,24,19].

Utilizing structural information

The structure of the human skeleton is constrained by fixed bone lengths, joint angle limits, and limb interpenetration constraints. Some approaches use these constraints to infer 3D from 2D joint locations.

Sun et al. [34] re-parameterize the pose presentation to use bones instead of joints and propose a structure-aware loss.
[34] X. Sun, J. Shang, S. Liang, and Y. Wei. Compositional human pose regression. In ICCV, 2017.

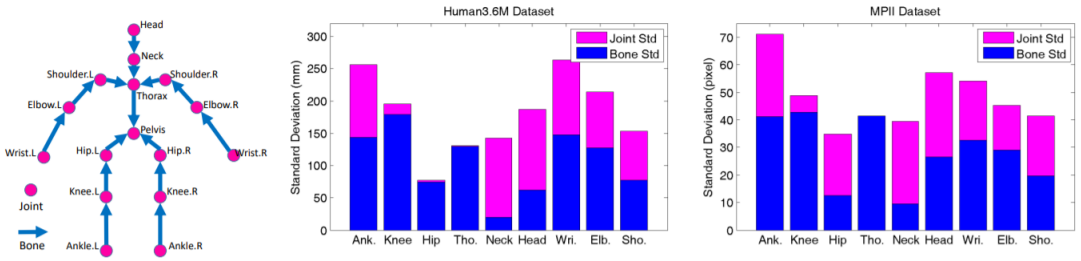


Figure 1. Left: a human pose is represented as either joints $\mathcal{J}$ or bones $\mathcal{B}$. Middle/Right: standard deviations of bones and joints for the 3D Human3.6M dataset [20] and 2D MPII dataset [3].

Zhou et al. [41] introduce a weakly-supervised framework for joint training with 2D and 3D data with the help of a geometric loss function to exploit the consistency of bone-length ratios in human body.
[41] X. Zhou, Q. Huang, X. Sun, X. Xue, and Y. Wei. Towards 3d human pose estimation in the wild: A weakly-supervised approach. In ICCV, 2017.

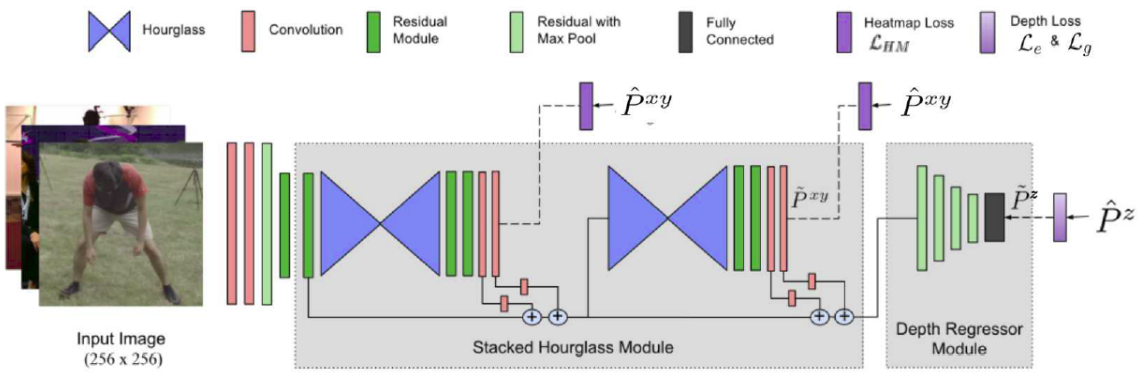


Fig. 1. A schematic of the network architecture. The stacked hourglass module is trained using the standard Euclidean loss $\mathcal{L}_{HM}$ against ground truth heatmaps. Whereas, the depth regressor module is trained on either $\mathcal{L}^z_{3D}$ or $\mathcal{L}^z_{2D}$ depending on whether the ground truth depth $\hat{P}^z$ is available or not.

Bogo et al. [4] penalize body-part interpenetration and illegal joint angles in their objective function for finding SMPL [21] based shape and pose parameters.
[4] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In ECCV, 2016.

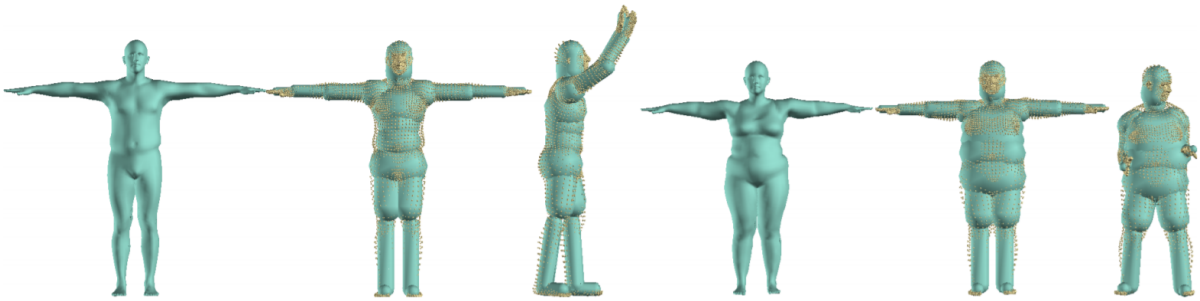


Fig. 3. Body shape approximation with capsules. Shown for two subjects. Left to right: original shape, shape approximated with capsules, capsules reposed. Yellow point clouds represent actual vertices of the model that is approximated.

To fit the 3D pose and shape to the CNN-detected 2D joints, we minimize anobjective function that is the sum of five error terms: a joint-based data term,three pose priors, and a shape prior; that is E(β, θ) =

$$E_J(\boldsymbol{\beta}, \boldsymbol{\theta}; K, J_{\text{est}}) + \lambda_\theta E_\theta(\boldsymbol{\theta}) + \lambda_a E_a(\boldsymbol{\theta}) + \lambda_{sp} E_{sp}(\boldsymbol{\theta}; \boldsymbol{\beta}) + \lambda_\beta E_\beta(\boldsymbol{\beta}) \qquad (1)$$

where K are camera parameters and λθ, λa, λsp λβ are scalar weights.

Utilizing temporal information

Direct estimation of 3D pose from disjointed images leads to temporally incoherent output with visible jitters and varying bone lengths. 3D pose estimates from a video can be improved by using simple filters or temporal priors.

Dataset

Present ablation studies, quantitative results on Human3.6M and MPI-INF-3DHP datasets and comparisons with previous art, and qualitative results on MPII 2D and MS COCO datasets.

2D dataset

MS-COCO and MPII are in-the-wild 2D pose datasets with no 3D ground truth annotations.

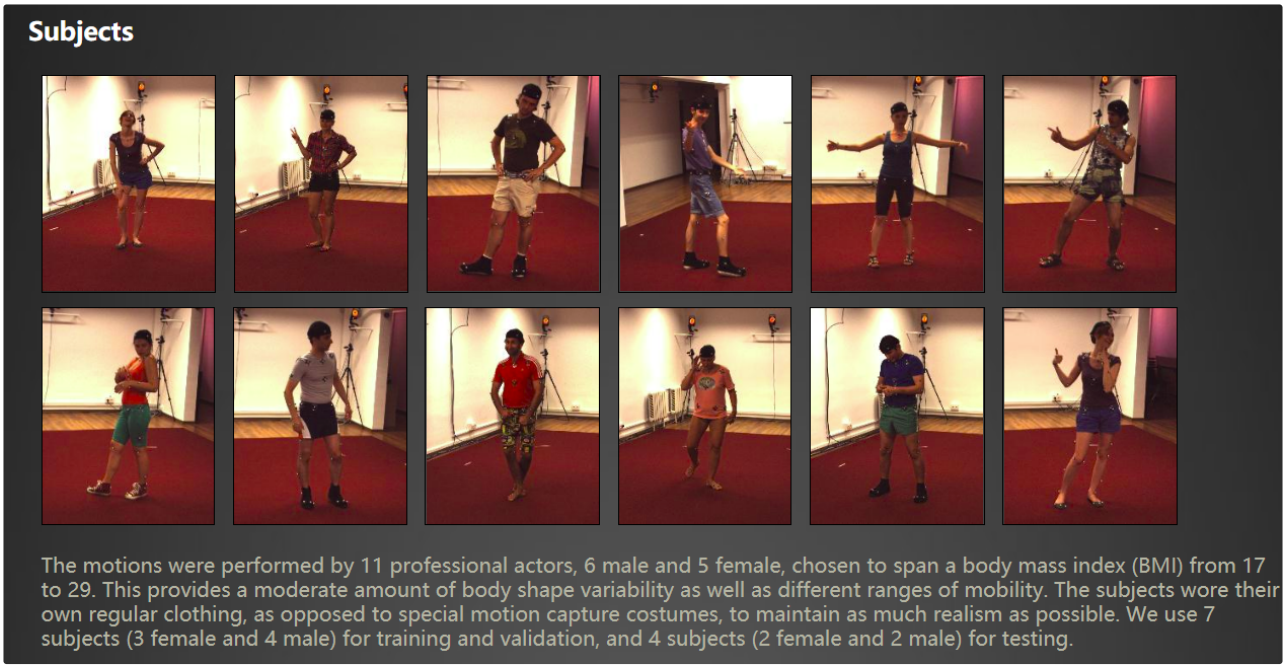
3D dataset

Human3.6M

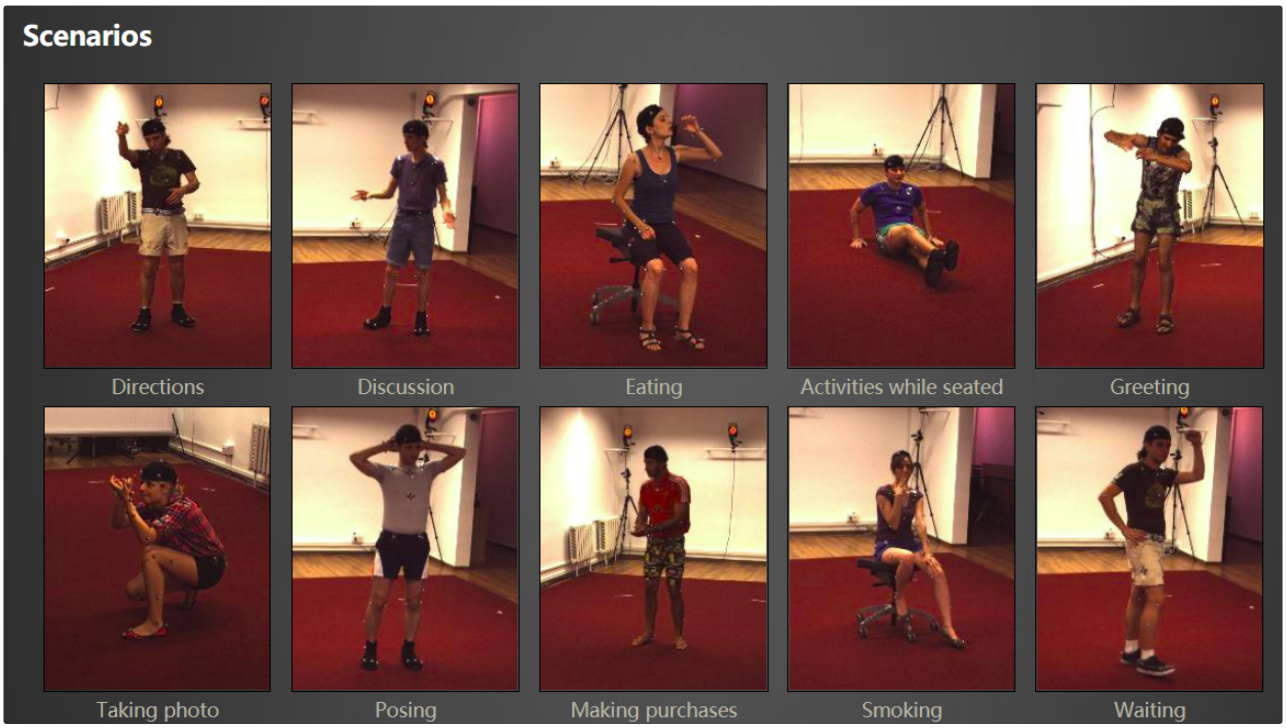
11 subjects performing different indoor actions with ground-truth annotations captured using a marker-based MoCap system.

We follow [35] and evaluate our results under 1) Protocol 1 that uses Mean Per Joint Position Error (MPJPE) as the evaluation metric w.r.t. root relative poses and 2) Protocol 2 that uses Procrustes Aligned MPJPE (PAMPJPE) which is MPJPE calculated after rigid alignment of predicted pose with the ground truth. As is common, we evaluate the results on every fifth frame.

Link



The dataset consists of 3.6 million different human poses collected with 4 digital cameras. Our data is organized into 15 training motions containing walking with many types of asymmetries (e.g. walking with a hand in a pocket, walking with a bag on the shoulder), sitting and laying down poses, various types of waiting poses and other types of poses. The actors were given detailed tasks with examples in order to help them plan a stable set of poses between repetitions for the creation of training, validation and test sets. In the execution of these tasks the actors were however given quite a bit of freedom in moving naturally over a strict, rigid interpretation of the tasks.

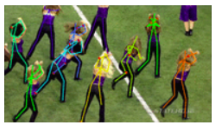


POSETRACK CHALLENGE ARTICULATED PEOPLE TRACKING IN THE WILD

CHALLENGES

1. POSETRACK CHALLENGE ON ARTICULATED HUMAN POSE ESTIMATION AND TRACKING



In this challenge it will be required to estimate and track 2D articulated poses of multiple people in real-world videos. Both single-frame pose estimation accuracy as well as articulated tracking accuracy will be evaluated and a winner will be determined in each category. The videos in this challenge will be similar to those included in the PoseTrack'17 benchmark at ICCV'17. To further improve the benchmark this year we will double the dataset size.

New: The PoseTrack18 dataset and evaluation code are now available here.

Organizers: Mykhaylo Andriluka, Umar Iqbal, Anon Milan, Eldar Insafutdinov, Christoph Lassner, Siyu Tang, Leonid Pishchulin, Juergen Gall, Bernt Schiele

Full Benchmark Release and Evaluation Server Opening: August 6th, 2018

Results Submission: August 31st, 2018

2. DENSEPOSE-POSETRACK CHALLENGE ON DENSE POSE ESTIMATION IN TIME

In this challenge the participants will be required to estimate dense correspondences between people videos and a 3D body shape model. The challenge is based on the data from PoseTrack'17 benchmark that has been annotated with dense pose correspondences. More details about the task are available at densepose.org

New: The DensePose-PoseTrack dataset and evaluation code are now available here.

Evaluation server is now open for submissions.

Organizers: Rıza Alp Güler, Natalia Neverova and Iasonas Kokkinos

Train+Val Pre-Release: July 2nd, 2018

Full Benchmark Release and Evaluation Server Opening: July 15th, 2018

Results Submission: August 18th, 2018

3. 3D HUMAN POSE ESTIMATION CHALLENGE

In this challenge the participants are required to estimate poses of people in 3D. The challenge is based on the popular Human3.6 benchmark which offers the possibility of estimating 2D and 3D skeletal joint positions, joint angles, semantic segmentation of body parts, as well as 3D human shape and depth, and will in addition provide and evaluate dense correspondences similar to the DensePose challenge.

New: The dataset for the 3D human pose estimation challenge is now available here.

Organizers: Andrei Zanfir, Elisabeta Marinoiu, Alin Popa, Mihai Zanfir, Vlad Olaru and Cristian Sminchisescu

Train+Val Pre-Release: July 9th, 2018

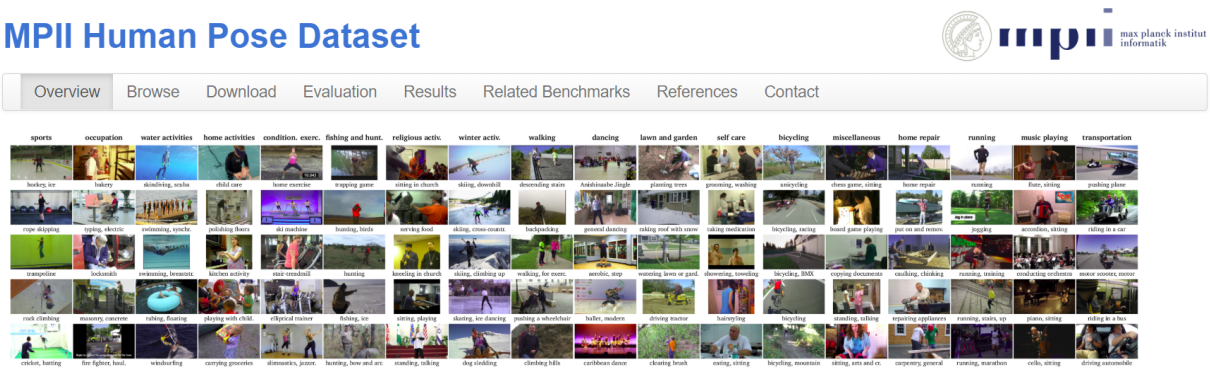
Full Benchmark Release and Evaluation Server Opening: July 22nd, 2018

Results Submission: August 18th, 2018

MPI-INF-3DHP (test) dataset

MPI-INF-3DHP (test) dataset is a recently released dataset of 6 test subjects with different indoor settings (green screen and normal background) and 2 subjects performing in-the-wild that makes it more challenging than Human3.6M, which only has a single indoor setting. We follow the evaluation metric proposed in [22] and report Percentage of Correct Keypoints (PCK) within 150mm range and Area Under Curve (AUC).

[22] D. Mehta, H. Rhodin, D. Casas, P. Fua, O. Sotnychenko,W. Xu, and C. Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In 3DV, 2017.



3D human pose Methods

Learning 3D Human Pose from Structure and Motion

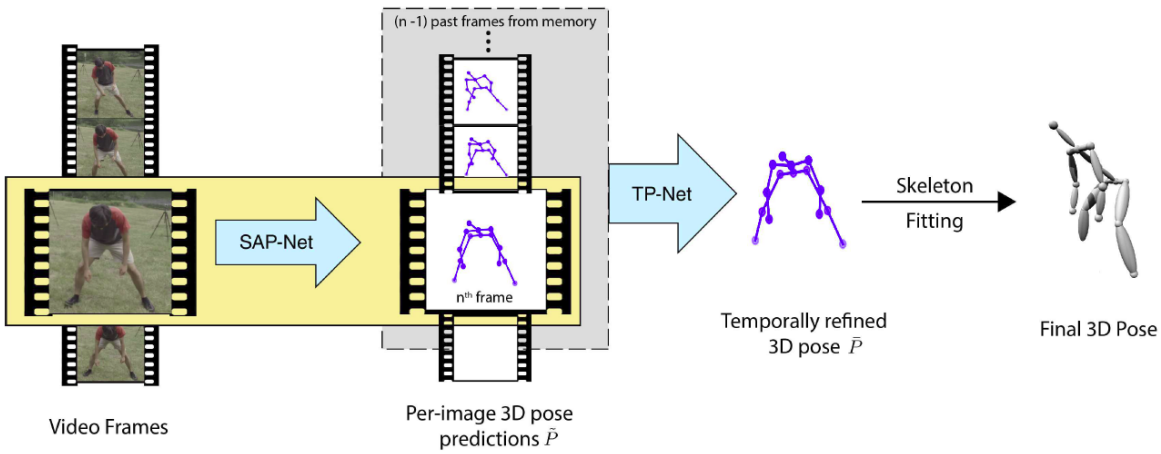


Fig. 2. Overall pipeline of our method: We sequentially pass the video frames to a ConvNet that produces 3D pose outputs (one at a time). Next, the prediction is temporally refined by passing a context of past N frames along with the current frame to a temporal model. Finally, skeleton fitting may be performed as an optional step depending upon the application requirement.

Structure-Aware PoseNet or SAP-Net

Most body joints are constrained to move within a certain angular limits only. Our illegal angle loss, La, encapsulates this constraint for the knee and elbow joints and restricts their bending beyond 180◦.

/

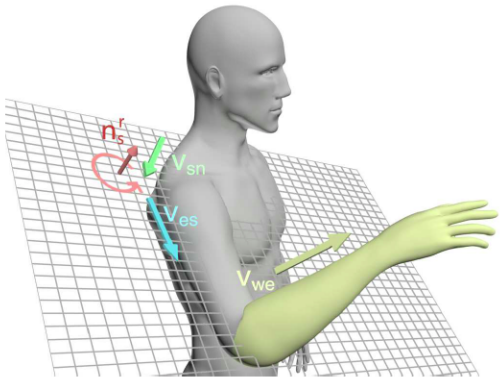


Fig. 3. Illustration of Illegal Angle loss: For the elbow joint angle to be legal, the lower-arm must project a positive component along $\mathbf{n_s^r}$ (normal to collarbone-upperarm plane) , i.e. $\mathbf{n_s^r} \cdot \mathbf{v_{we}} \geq 0$. Note that we only need 2D annotated data to train our model using this formulation.

We exponentiate E to strongly penalize large deviations beyond legality. La can now be defined as:

$$\mathcal{L}_a = -E_e^r e^{-E_e^r} + E_e^l e^{E_e^l} + E_k^r e^{E_k^r} - E_k^l e^{-E_k^l} \tag{1}$$

Symmetry Loss

It is simple yet heavily constrains the joint depths, especially when the inferred depth is ambiguous due to occlusions. Ls is defined as the difference in lengths of left/right bone pairs.

Let B be the set of all the bones on right half of the body except torso and head bones. Also, let BLb represent the bone-length of bone b. We define Ls as

$$\mathcal{L}_s = \sum_{b \in \mathcal{B}} ||BL_b - BL_{C(b)}||_2 \tag{2}$$

where C(.) indicates the corresponding left side bone.

Finally, our structure-aware loss

$$\mathcal{L}_{SA}^z(\tilde{P}^z, \hat{P}^{xy}) = \lambda_a \mathcal{L}_a(\tilde{P}^z, \hat{P}^{xy}) + \lambda_s \mathcal{L}_s(\tilde{P}^z, \hat{P}^{xy}) + \lambda_g \mathcal{L}_g(\tilde{P}^z, \hat{P}^{xy}) \tag{3}$$

Loss Surface Visualization:

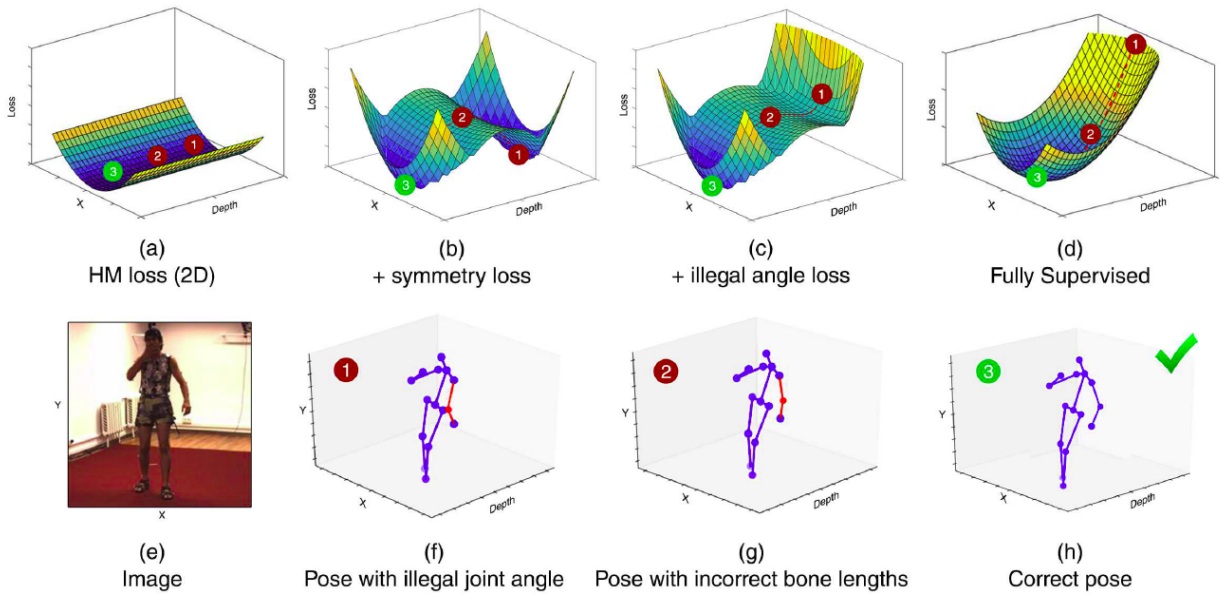


Fig. 4. Loss Surface Evolution Plots (a) to (d) show the local loss surfaces for (a) 2D-location loss. (b) 2D-location+symmetry loss (c) 2D-location+symmetry+illegal angle loss and (d) full 3D-annotation Euclidean loss. The points (1), (2) and (3) highlighted on the plots are the corresponding 3D poses shown in (f), (g) and (h), with (3) being the ground-truth depth. The illegal angle penalty increases the loss for pose (1), which has the elbow bent backwards. Pose (2) has a legal joint angle, but the symmetry is lost. Pose (3) is correct. We can see that without the angle loss, the loss at (1) and (3) are equal and we cannot discern between the two points.

Temporal PoseNet or TP-Net

In this section we propose to learn a temporal pose model, referred as Temporal PoseNet, to exploit the temporal consistency and motion cues present in video sequences. Given independent pose estimates from SAP-Net, we seek to exploit the information from a set of adjacent pose-estimates Padj to improve the inference for the required pose P.

We propose to use a simple two-layer, 4096 hidden neurons, fully-connected network with ReLU non-linearity that takes a fixed number, N = 20, of adjacent poses as inputs and outputs the required pose ¯ P. The adjacent pose vectors are simply flattened and concatenated in order to make a single vector that goes into the TP-Net and it is trained using standard L2 loss from the ground-truth pose.

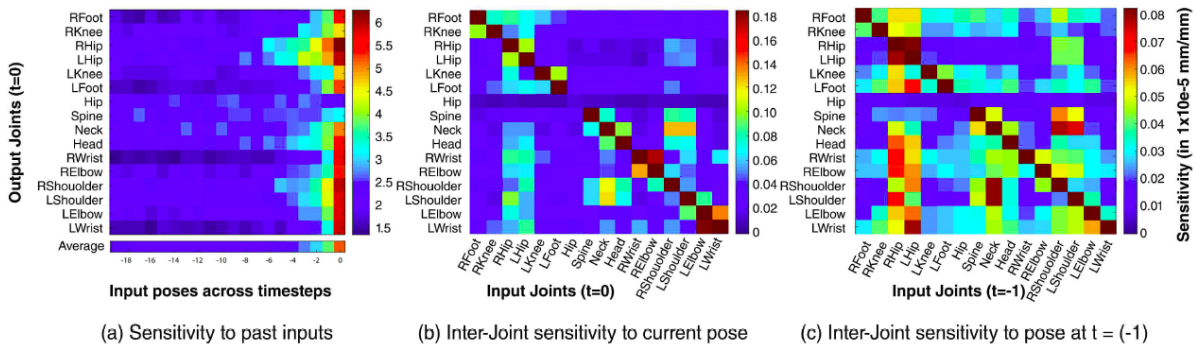


Fig. 5. (a) The variation of sensitivity in output pose w.r.t to the perturbations in input poses of TP-Net for from t=0 to t=-19. (b) Strong structural correlations are learned from the pose input at t=0 frame. (c) Past frames show smaller but more complex structural correlations. The self correlations (diagonal elements) are an order of magnitude larger and the colormap range has been capped to better display.

Unsupervised Geometry-Aware Representation for 3D Human Pose Estimation

In this paper, we propose to overcome this problem by learning a geometry-aware body representation from **multi-view** images without annotations. To this end, we use an encoder-decoder that predicts an image from one viewpoint given an image from another viewpoint. Because this representation encodes 3D geometry, using it in a semi-supervised setting makes it easier to learn a mapping from it to 3D human pose. As evidenced by our experiments, our approach significantly outperforms fully-supervised methods given the same amount of labeled data, and improves over other semi-supervised methods while using as little as 1% of the labeled data.

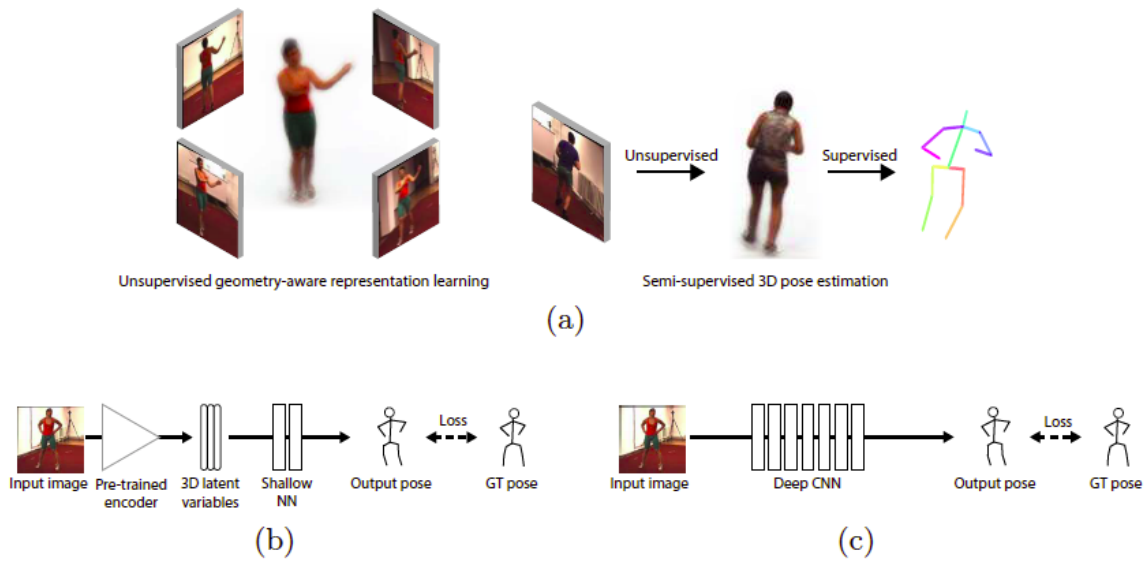


Fig. 1: Approach. (a) During training, we first learn a geometry-aware representation using unlabeled multi-view images. We then use a small amount of supervision to learn a mapping from our representation to actual 3D poses, which only requires a shallow network and therefore a limited amount of supervision. (b) At run-time, we compute the latent representation of the test image and feed it to the shallow network to compute the pose. (c) By contrast, most state-of-the-art approaches train a network to regress directly from the images to the 3D poses, which requires a much deeper network and therefore more training data.

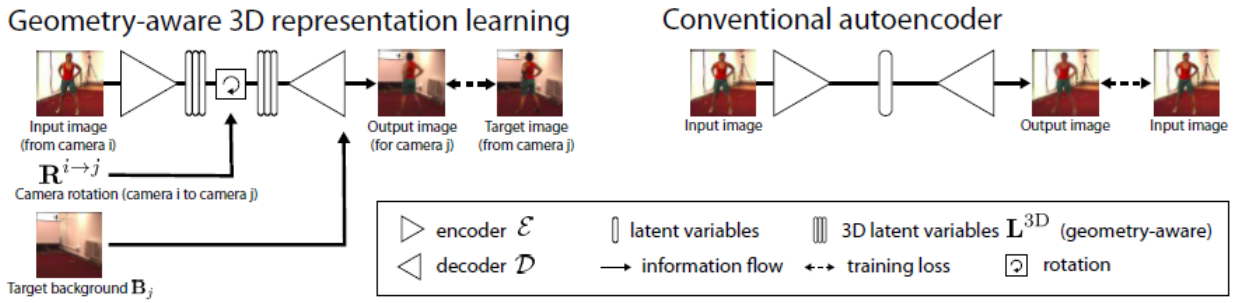


Fig. 2: Representation learning. We learn a representation that encodes geometry and thereby 3D pose information in an unsupervised manner. Our method (Left) extends a conventional auto encoder (Right) with a 3D latent space, rotation operation, and background fusion module. The 3D rotation enforces explicit encoding of 3D information. The background fusion enables application to natural images.

解决的问题是多个视角下的3D pose预测，本文的两点在于如何学习一个通用的高效的中间特征表达。
和目前做的setting不是很相同。

Deeply Learned Compositional Models for Human Pose Estimation

Compositional Human Pose Regression

Keep it SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image

Dense Pose

Defination

Dense pose estimation aims at mapping all human pixels of an RGB image to the 3D surface of the human body.

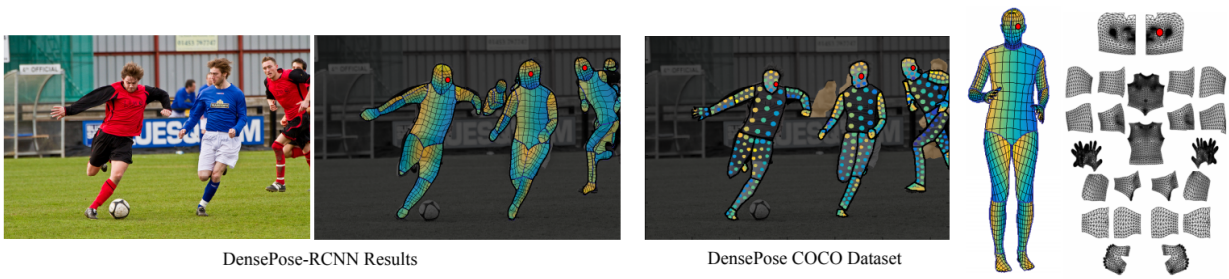


Figure 1: Dense pose estimation aims at mapping all human pixels of an RGB image to the 3D surface of the human body. We introduce DensePose-COCO, a large-scale ground-truth dataset with image-to-surface correspondences manually annotated on 50K COCO images and train DensePose-RCNN, to densely regress part-specific UV coordinates within every human region at multiple frames per second. *Left:* The image and the regressed correspondence by DensePose-RCNN, *Middle:* DensePose COCO Dataset annotations, *Right:* Partitioning and UV parametrization of the body surface.

Dataset

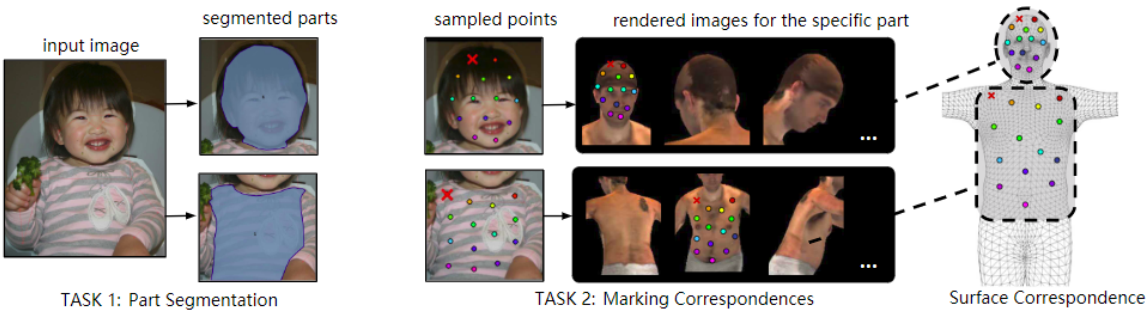
We involve human annotators to establish dense correspondences from 2D images to surface-based representations of the human body. If done naively, this would require by manipulating a surface through rotations - which can be

frustratingly inefficient. Instead, we construct a two-stage annotation pipeline to efficiently gather annotations for image-to-surface correspondence.

As shown below, in the first stage we ask annotators to delineate regions corresponding to visible, semantically defined body parts. We instruct the annotators to estimate the body part behind the clothes, so that for instance wearing a large skirt would not complicate the subsequent annotation of correspondences.

In the second stage we sample every part region with a set of roughly equidistant points and request the annotators to bring these points in correspondence with the surface. In order to simplify this task we `unfold' the part surface by providing six pre-rendered views of the same body part and allow the user to place landmarks on any of them. This allows the annotator to choose the most convenient point of view by selecting one among six options instead of manually rotating the surface.
We use the SMPL model and SURREAL textures in the data gathering procedure.

We annotate dense correspondence between images and a 3D surface model by asking the annotators to segmentthe image into semantic regions and to then localize the corresponding surface point for each of the sampled points on anyof the rendered part images. The red cross indicates the currently annotated point. The surface coordinates of the renderedviews localize the collected 2D points on the 3D model.

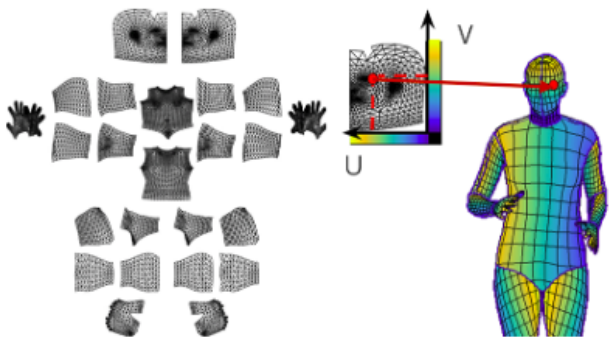


DensePose-RCNN System

Similar to DenseReg, our strategy to find dense correspondence by partitioning the surface. And for every pixel, determine:

- which surface part it belons to,
- where on the 2D paremeterization of the part it corresponds to.

On the right, partitioning of the surface and "correspondence to a point on a part" is demonstrated.

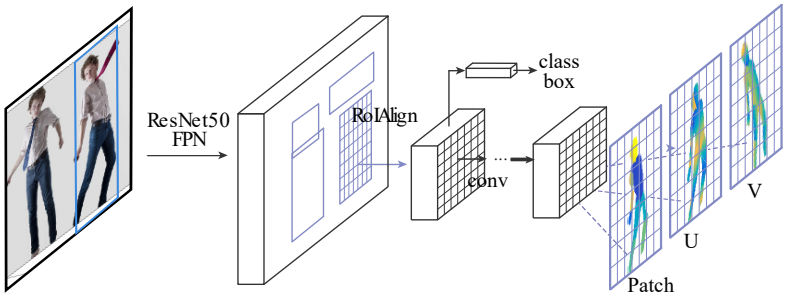


We adopt the architecture of Mask-RCNN with the Feature Pyramid Network (FPN) features, and ROI-Align pooling so as to obtain dense part labels and coordinates within each of the selected regions.

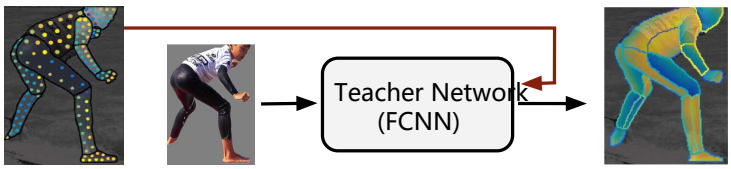
As shown below, we introduce a fully-convolutional network on top of the ROI-pooling that is entirely devoted to two tasks:

- Generating per-pixel classification results for selection of surface part.
- For each part regressing local coordinates within part.

During inference, our system operates at 25fps on 320x240 images and 4-5fps on 800x1100 images using a GTX1080 graphics card.



The DensePose-RCNN system can be trained directly using the annotated points as supervision. However, we obtain substantially better results by ``inpainting'' the values of the supervision signal on positions that are not originally annotated. To achieve this, we adopt a learning-based approach where we firstly train a ``teacher'' network: A fully-convolutional neural network(depicted below) that reconstructs the ground-truth values given images scale-normalized images and the segmentation masks.



We further improve the performance of our system using cascading strategies. Via cascadning, we exploit information from related tasks, such as keypoint estimation and instance segmentation, which have successfully been addressed by the Mask-RCNN architecture. This allows us to exploit task synergies and the complementary merits of different sources of supervision.

Human Parsing

Defination

Multi-Human Parsing refers to partitioning a crowd scene image into semantically consistent regions belonging to the body parts or clothes items while differentiating different identities, such that each pixel in the image is assigned a semantic part label, as well as the identity it belongs to. A lot of higher-level applications can be founded upon Multi-Human Parsing, such as group behavior analysis, person re-identification, image editing, video surveillance, autonomous driving and virtual reality.

Dataset

Multi-Human Parsing

The MHP v1.0 dataset contains 4,980 images, each with at least two persons (average is 3). We randomly choose 980 images and their corresponding annotations as the testing set. The rest form a training set of 3,000 images and a validation set of 1,000 images. For each instance, 18 semantic categories are defined and annotated except for the "background" category

The MHP v2.0 dataset contains 25,403 images, each with at least two persons (average is 3). We randomly choose 5,000 images and their corresponding annotations as the testing set. The rest form a training set of 15,403 images and a validation set of 5,000 images. For each instance, 58 semantic categories are defined and annotated except for the "background" category,

Multi-Human Parsing Track

https://lv-mhp.github.io/human_parsing_task

Multi-Human Pose Estimation Track

https://lv-mhp.github.io/pose_estimation_task

Related work

Macro-Micro Adversarial Network for Human Parsing

motivation

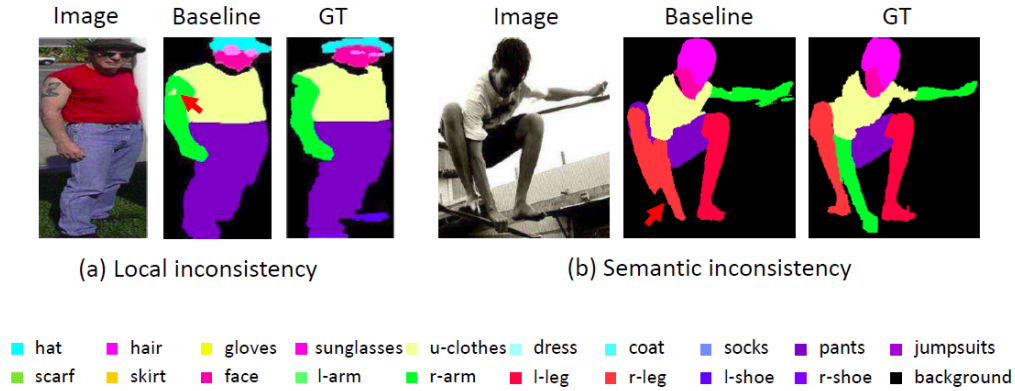


Fig. 1: Drawbacks of the pixel-wise classification loss. (a) Local inconsistency, which leads to a hole on the arm. (b) Semantic inconsistency, which causes unreasonable human poses. The inconsistencies are indicated by red arrows.

method

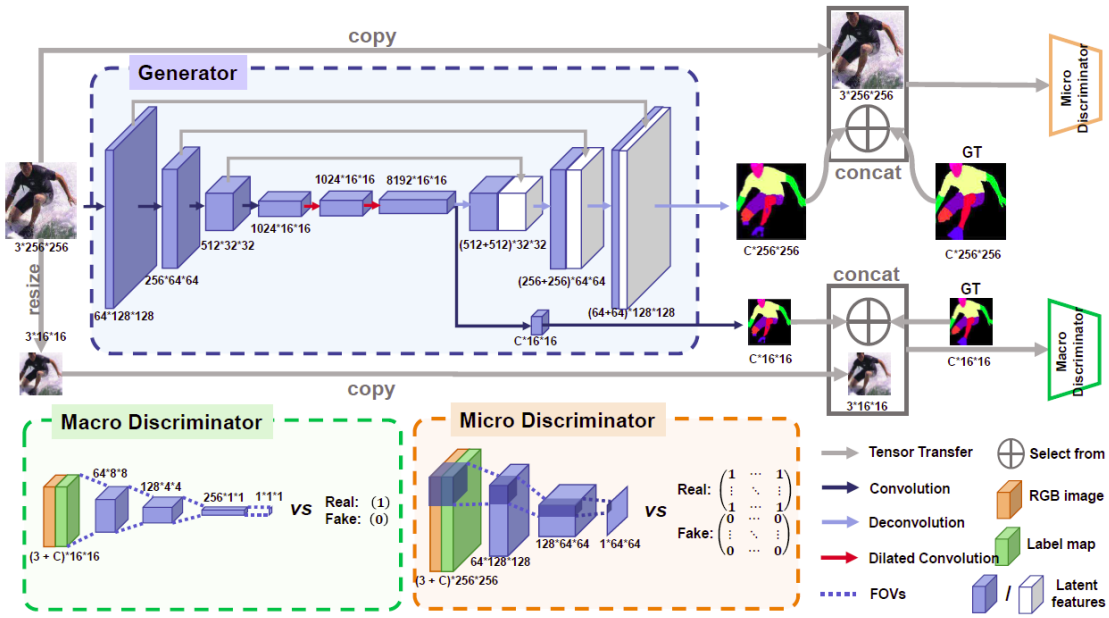
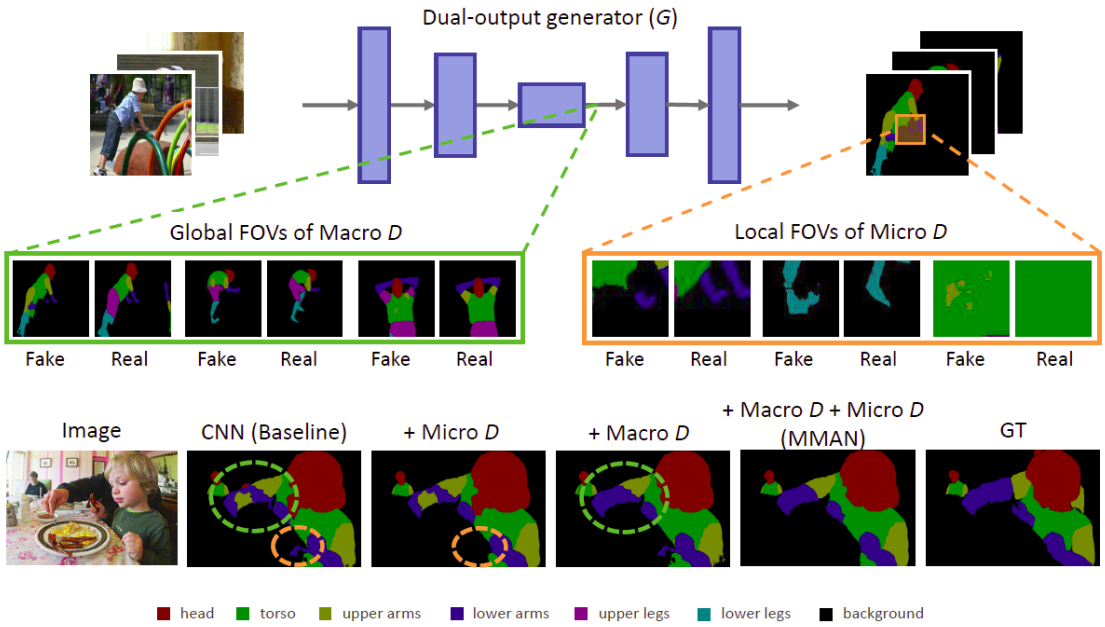


Fig. 4: MMAN has three components: a dual-output generator (blue dashed box), a Macro discriminator (green dashed box) and a Micro discriminator (orange dashed box). Given an input image of size $3 \times 256 \times 256$, the generator G first produces a low-resolution ($8192 \times 16 \times 16$) tensor, from which a low-resolution label map ($C \times 16 \times 16$) and a high-resolution label map ($C \times 256 \times 256$) are generated, where C is the number of classes. Finally, for the each label map (sized $C \times 16 \times 16$, for example), we concatenate it with an RGB image (sized $3 \times 16 \times 16$) along the 1st axis (number of channels), which is fed into the corresponding discriminator.

visualization



experiment

Table 2: Performance comparison in terms of per-class IoU with five state-of-the-art methods on the PASCAL-Person-Part test set.

Method	head	torso	u-arms	l-arms	u-legs	l-legs	bkg	avg
Deeplab-ASPP [2]	81.33	60.06	41.16	40.95	37.49	32.56	92.81	55.19
HAZN [33]	80.79	59.11	43.05	42.76	38.99	34.46	93.59	56.11
Attention [3]	81.47	59.06	44.15	42.50	38.28	35.62	93.65	56.39
LG-LSTM [20]	82.72	60.99	45.40	47.76	42.33	37.96	88.63	57.97
Attention + SSL [10]	83.26	62.40	47.80	45.58	42.32	39.48	94.68	59.36
Do-Deeplab-ASPP	81.82	59.53	44.80	42.79	38.32	36.38	93.91	56.79
Macro AN	82.01	61.19	45.24	44.30	39.73	36.75	93.89	57.58
Micro AN	82.44	61.35	44.79	43.68	38.41	36.05	93.93	57.23
MMAN	82.46	61.41	46.05	45.17	40.93	38.83	94.30	58.45
Attention + MMAN	82.58	62.83	48.49	47.37	42.80	40.40	94.92	59.91

Table 3: Comparison of human parsing accuracy on the PPSS dataset [25]. Best performance is highlighted in blue.

Method	head	face	up-cloth	arms	lo-cloth	legs	bkg	avg
DL [25]	22.0	29.1	57.3	10.6	46.1	12.9	68.6	35.2
DDN [25]	35.5	44.1	68.4	17.0	61.7	23.8	80.0	47.2
ASN [24]	51.7	51.0	65.9	29.5	52.8	20.3	83.8	50.7
MMAN	53.1	50.2	69.0	29.4	55.9	21.4	85.7	52.1

Instance-level Human Parsing via Part Grouping Network

A novel Part Grouping Network (PGN) is proposed to solve multi-person human parsing in a unified network at once by reformulating it as two twinned grouping tasks that can be mutually refined: semantic part segmentation and instance-aware edge detection.

In this section, we begin by presenting a general pipeline of our approach (see **Fig. 4**) and then describe each component in detail. The proposed Part Grouping Network (PGN) jointly train and refine the semantic part segmentation and instance-aware edge detection in a unified network. Technically, these two sub-tasks are both pixel-wise

classification problem, on which Fully Convolutional Networks (FCNs) [29] perform well. Our PGN is thus constructed based on FCNs structure, which **first learns common representation using shared intermediate layers and then appends two parallel branches with respect to semantic part segmentation and edge detection.** To explore and take advantage of the semantic correlation of these two tasks, **a refinement branch is further incorporated to make two targets mutually beneficial for each other by exploiting complementary contextual information**. Finally, an efficient **partition process with a heuristic grouping algorithm can be used to generate instance-level human parsing results** using a breadth-first search over line segments obtained by jointly scanning the generated semantic part segmentation maps and instance-aware edge maps.

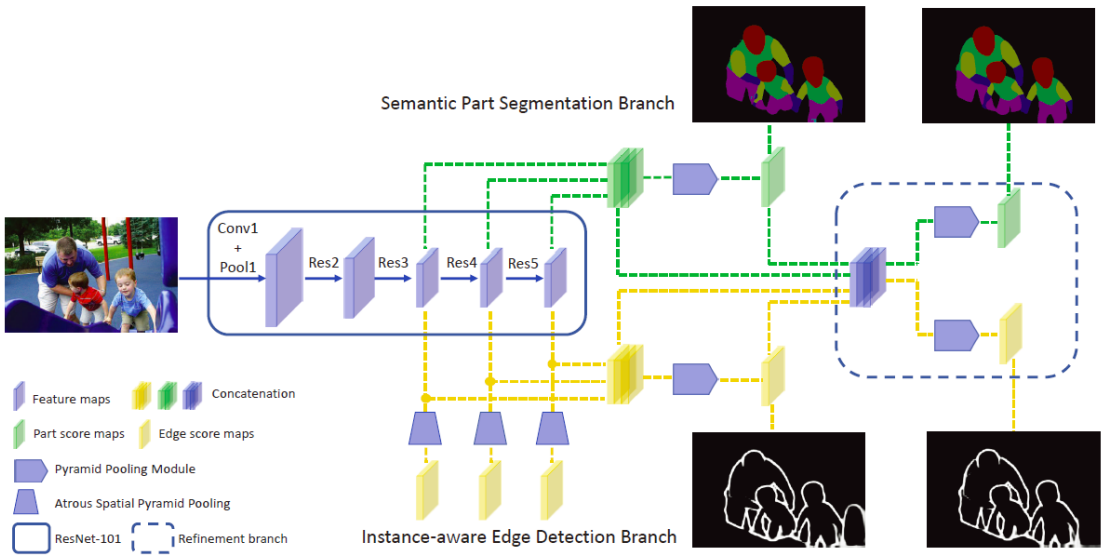


Fig. 4: Illustration of our Part Grouping Network (PGN). Given an input image, we use ResNet-101 to extract the shared feature maps. Then, two branches are appended to capture part context and human boundary context while simultaneously generating part score maps and edge score maps. Finally, a refinement branch is performed to refine both predicted segmentation maps and edge maps by integrating part segmentation and human boundary contexts.

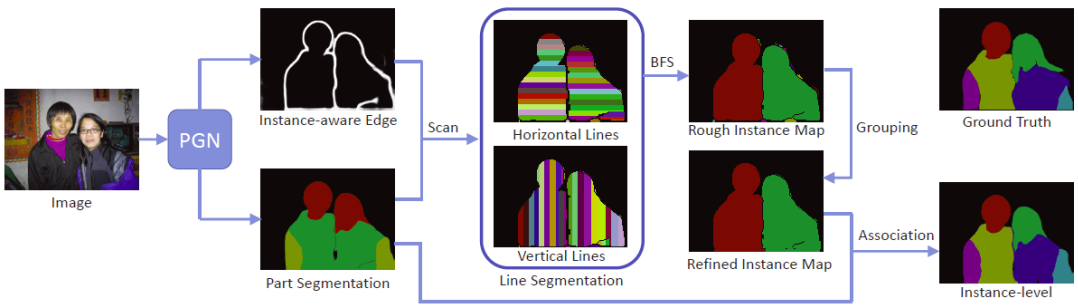


Fig. 5: The whole pipeline of our approach to tackle instance-level human parsing. Generated from the PGN, the part segmentation maps and edge maps are scanned simultaneously to create horizontal and vertical segmented lines. Just like a connected graph problem, the breadth-first search can be applied to group segmented lines into regions. Furthermore, the small regions near the instance boundary are merged into their neighbor regions that cover larger areas and several part labels. Associating the instance maps and part segmentation maps, the pipeline finally outputs a well-predicted instance-level human parsing result without any proposals from object detection.

In summary, the whole learning objective of PGN can be written as:

$$L = \alpha \cdot (L_{\text{seg}} + L'_{\text{seg}}) + \beta \cdot (L_{\text{edge}} + L'_{\text{edge}} + \sum_{n=1}^{N} L^n_{\text{side}}). \quad (1)$$

Mutual Learning to Adapt for Joint Human Parsing and Pose Estimation

MuLA predicts dynamic task-specific model parameters via recurrently leveraging guidance information from its parallel tasks. Thus MuLA can fast adapt parsing and pose models to provide more powerful representations by incorporating information from their counterparts, giving more robust and accurate results. MuLA is imple-mented with convolutional neural networks and end-to-end trainable.

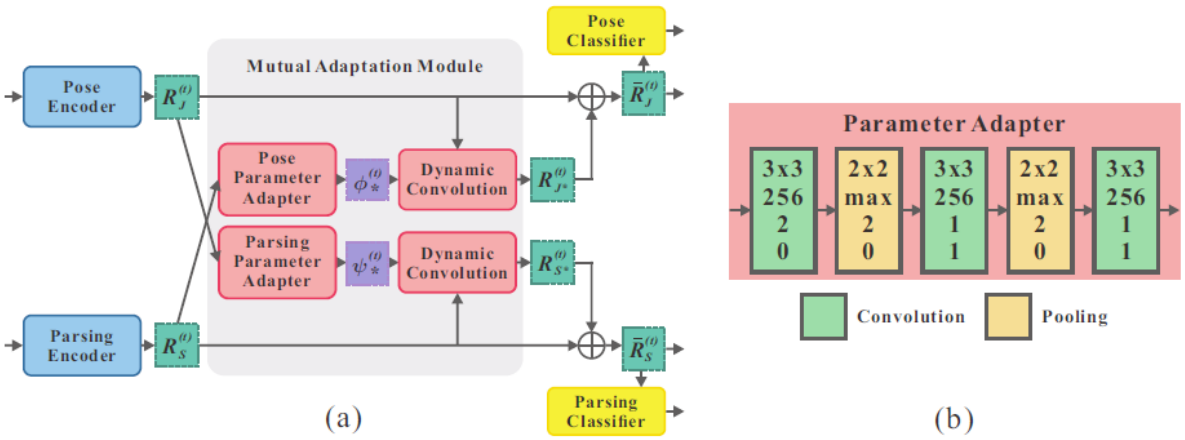


Fig. 3. (a) The CNN implementation of MuLA for one stage. Given inputs $F_S^{(t)}$ and $F_J^{(t)}$ at stage t, the parsing and pose encoders generate preliminary representations $R_S^{(t)}$ and $R_J^{(t)}$. Then, the parameter adapters predict dynamic parameters $\psi_*^{(t)}$ and $\phi_*^{(t)}$ for learning complementary representations $R_{S*}^{(t)}$ and $R_{J*}^{(t)}$ via dynamic convolutions, which are exploited to tailor preliminary representations via addition in a residual manner for producing refined representations $\bar{R}_S^{(t)}$ and $\bar{R}_J^{(t)}$. Finally, MuLA feeds $\bar{R}_S^{(t)}$ and $\bar{R}_J^{(t)}$ to classifiers for parsing and pose estimation, respectively. (b) The network architecture of parameter adapter, consisting of three convolution and two pooling layers. For each layer, the kernel size, the number of channel/pooling types, stride and padding size are specified from top to bottom

Mutual Adaptation Module

However, it is not feasible to directly predict all the convolution kernels due to their large scale. To reduce the number of kernels to predict by adapters $A_{\psi_a^{(t)}}(\cdot)$ and $A_{\phi_a^{(t)}}(\cdot)$, we follow [2] to use a way analogous to SVD for decomposing parameters $\psi_*^{(t)}$ and $\phi_*^{(t)}$ via

$$\psi_*^{(t)} = U_S^{(t)} \otimes \widetilde{\psi}_*^{(t)} \otimes_c V_S^{(t)} \text{ and } \phi_*^{(t)} = U_J^{(t)} \otimes \widetilde{\phi}_*^{(t)} \otimes_c V_J^{(t)}, \quad (5)$$

where $\otimes$ denotes convolution operation, $\otimes_c$ denotes channel-wise convolution operation, $U_S^{(t)}/U_J^{(t)}$ and $V_S^{(t)}/V_J^{(t)}$ are auxiliary parameters and can be viewed as parameter bases, and $\widetilde{\psi}_*^{(t)} \in \mathbb{R}^{h \times h \times c_i}$ and $\widetilde{\phi}_*^{(t)} \in \mathbb{R}^{h \times h \times c_i}$ are the actual parameters to predict by $A_{\phi_a^{(t)}}(\cdot)$ and $A_{\psi_a^{(t)}}(\cdot)$. In this way, the number of predicted parameters can be reduced by an order of magnitude.

[2] Bertinetto, L., Henriques, J.F., Valmadre, J., Torr, P., Vedaldi, A.: Learning feed-forward one-shot learners. In: NIPS (2016)

Human Pose Estimation with Parsing Induced Learner

In this work, we propose to leverage human parsing informationmore effectively and efficiently for learning betterpose estimation models and improving their performance.

Targeting at the above limitations of existing works, we makefollowing observations. First, the parsing representationsshould be learned towards being beneficial to pose estimation,instead of solely learned from the parsing supervision. Second, the learned parsing representations should be effectivelytransferable to the pose estimation domain. Third, the pose estimation model should be dynamic andcan fast adapt to various testing samples of different characteristics,relying on the transferred parsing information.

motivation

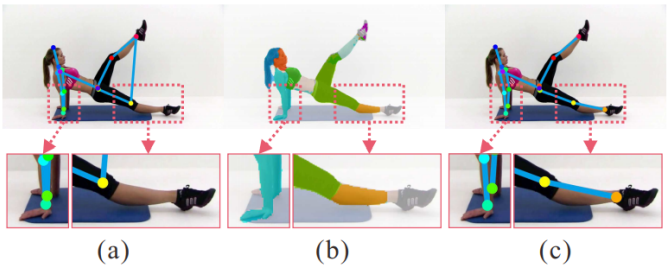


Figure 1. Illustration of our motivation for the proposed Parsing Induced Learner. (a) Pose estimation result without exploiting parsing information. (b) Parsing information generated from the proposed PIL. (c) Pose estimation result with the proposed PIL. The proposed PIL effectively leverages parsing information to refine the inaccurate locations and correct false categorizations for the highlighted body joints.

we design a novelParsing Induced Learner (PIL) that learns to fast adapt thepose estimation model conditioned on the parsing informationextracted from a specific sample, and therefore effectivelyimproves both performance and flexibility of themodel.

PIL consists of two components: an encoderthat encodes an input image into high-level parsing representations,and an adapter that learns to adapt parameters of thepose model by leveraging parsing representations.

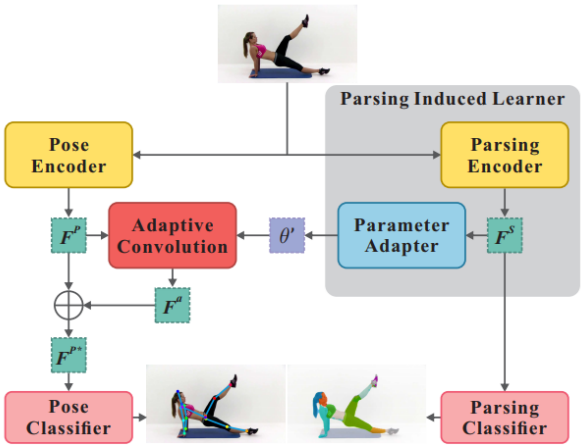


Figure 2. Overall architecture of our model. Given an input image, our model first utilizes a pose encoder to extract pose features F^P and the proposed PIL to predict dynamic parameters θ' through a parameter adapter taking in parsing features F^S from a parsing encoder. Then, our model feeds F^P and θ' to an adaptive convolution to extract parsing induced features F^a for fast adaption of the pose model. Our model regards F^a as residual information and fuses it with F^P via addition, leading to the refined features F^{P*} for body joint localization. Finally, our model inputs F^{P*} and F^S to pose and parsing classifiers, respectively, to produce pose estimation and parsing prediction (ignored during testing).

Parameter Adapter The parameter adapter Kφ(·) is theother component of the PIL model, which is a one-shotlearner to predict the dynamic parameters θ′ via taking inthe output FS of I from the parsing encoder network.

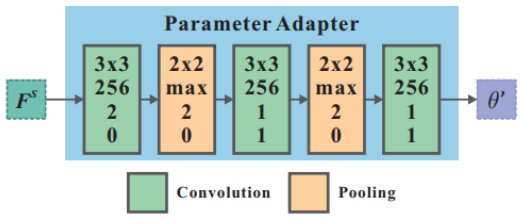


Figure 3. The architecture of the parameter adapter in PIL. The parameter adapter takes the features F^S from the parsing encoder as input and outputs the adaptive convolution parameters θ'. It is composed by stacking three convolution layers and two pooling layers. For each layer, the kernel size, the number of channels/pooling type, stride and padding size are specified from top to bottom.

Weakly and Semi Supervised Human Body Part Parsing via Pose-Guided Knowledge Transfer

3D Human Pose Estimation in the Wild by Adversarial Learning

In this paper, we propose an adversarial learningframework, which distills the 3D human pose structureslearned from the fully annotated dataset to in-the-wild imageswith only 2D pose annotations. Instead of defininghard-coded rules to constrain the pose estimation results,we design a novel multi-source discriminator to distinguishthe predicted 3D poses from the ground-truth, which helpsto enforce the pose estimator to generate anthropometricallyvalid poses even with images in the wild. Thus,we design a geometric descriptor, which computes the pairwiserelative locations and distances between body joints,as a new information source for the discriminator

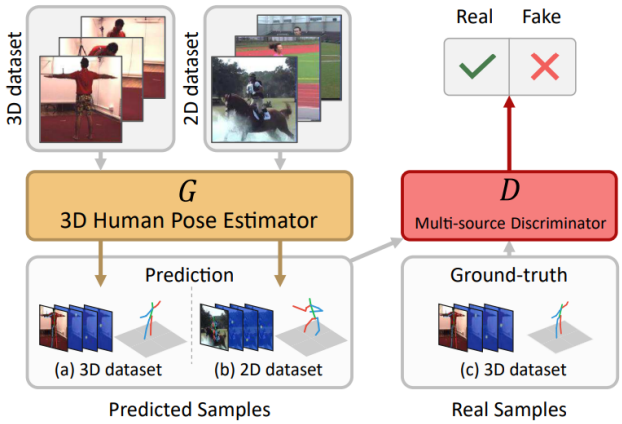


Figure 1. Given a monocular image and its predicted 3D pose, the human can easily tell whether the prediction is anthropometrically plausible or not (as shown in b) based on the perception of image-pose correspondence and the possible human poses constrained by articulation. We simulate this human perception by proposing an adversarial learning framework, where the discriminator is learned to distinguish ground-truth poses (c) from the predicted poses generated by the pose estimator (a, b), which in turn is enforced to generate plausible poses even on unannotated in-the-wild data.

As illustrated in Figure 1, our proposed frameworkcan be formulated as the Generative Adversarial Networks(GANs), which consist of two networks: a generator and adiscriminator. The generator is trained to generate samplesin a way that confuses the discriminator, which in turn triesto distinguish them from real samples. In our framework,the generator G is a 3D pose estimator, which tries to predictaccurate 3D poses to fool the discriminator. The discriminatorD distinguishes the ground-truth 3D poses from thepredicted ones.

The generator can be viewed as a two-stage pose estimator.We adopt the state-of-the-art architecture [57] as ourbackbone network for 3D human pose estimation.
The first stage is the 2D pose estimation module, whichis the stacked hourglass network [29]. The second stage is a depth regression module, whichconsists of several residual modules taking the 2D bodyjoint heatmaps and intermediate image features generatedfrom the first stage as input. The output is a P × 1 vectordenoting the estimated depth for each body joint.

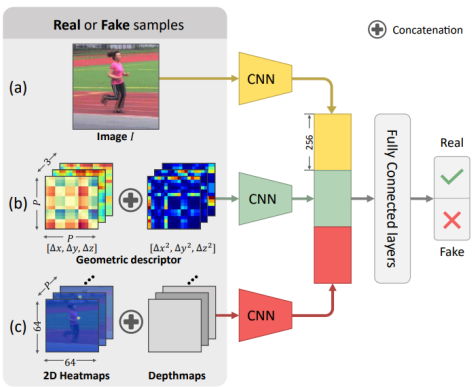


Figure 2. The multi-source architecture. It contains three information sources, image, geometric descriptor, as well as the heatmaps and depth maps. The three information sources are separately embedded and then concatenated for deciding if the input is the ground-truth pose or the estimated pose.

In the discriminator, there are three information sources: 1)the original image, 2) the pairwise relative locations and distances,and 3) the heatmaps of 2D locations and the depthsof body joints.

To learn the body articulation constraints, we designa geometric descriptor as the second information source(Figure 2(b)), which is motivated by traditional approachesbased on pictorial structures.

Our design of the geometric descriptor is motivated by thequadratic deformation constraints widely used in pictorialstructures [54, 32, 5] for 2D human pose estimation. It encodesthe spatial relationships, limbs length and symmetryof body parts. By extending it from 2D to 3D space, wedefine the 3D geometric descriptor d(·, ·) between pairs ofbody joints as a 6D vector:

image.png 0.13 MB 上传图片失败，请重试

Ordinal Depth Supervision for 3D Human Pose Estimation

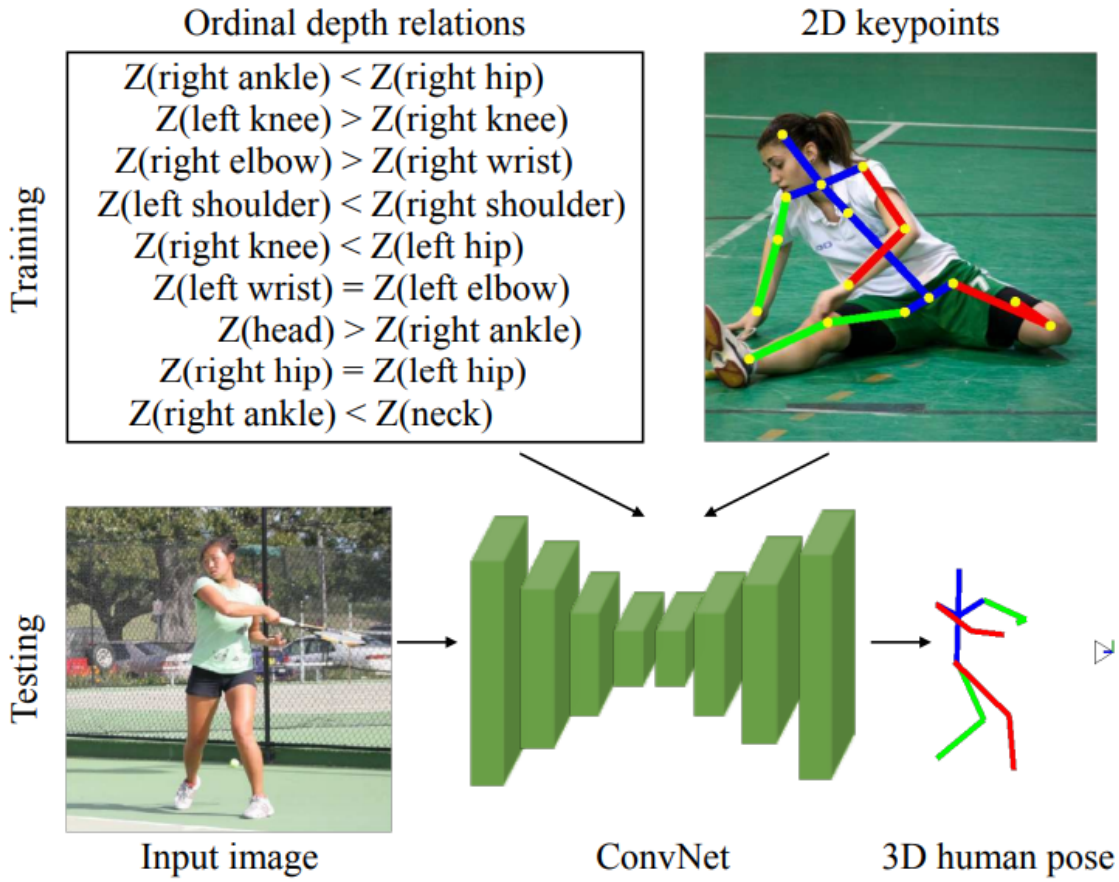


Figure 1: Summary of our approach. In the absence of accurate 3D ground truth we propose the use of ordinal depth relations (closer-farther) of the human body joints for end-to-end training of 3D human pose estimation systems.