UniPose+: A unified framework for 2D and 3D human pose estimation in images and videos

Bruno Artacho, *Member, IEEE,* and Andreas Savakis, *Senior Member, IEEE*

Abstract—We propose UniPose+, a unified framework for 2D and 3D human pose estimation in images and videos. The UniPose+ architecture leverages multi-scale feature representations to increase the effectiveness of backbone feature extractors, with no significant increase in network size and no postprocessing. Current pose estimation methods heavily rely on statistical postprocessing or predefined anchor poses for joint localization. The UniPose+ framework incorporates contextual information across scales and joint localization with Gaussian heatmap modulation at the decoder output to estimate 2D and 3D human pose in a single stage with state-of-the-art accuracy, without relying on predefined anchor poses. The multi-scale representations allowed by the waterfall module in the UniPose+ framework leverage the efficiency of progressive filtering in the cascade architecture, while maintaining multi-scale fields-of-view comparable to spatial pyramid configurations. Our results on multiple datasets demonstrate that UniPose+, with a HRNet, ResNet or SENet backbone and waterfall module, is a robust and efficient architecture for single person 2D and 3D pose estimation in single images and videos.

Index Terms—Human Pose Estimation, 3D Human Pose Estimation, Computer Vision, Deep Learning.

1 Introduction

Human pose estimation is an important task in computer vision that has motivated the development of several approaches, in 2D [61], [43], [59] and 3D [55], [75], [1]; on a single frame [5] or a video sequence [18]; for a single [62] or multiple subjects [10]. Pose estimation is challenging due to the large number of degrees of freedom in the human body mechanics and the frequent occurrence of joint occlusions. To deal with occlusion, many methods rely on statistical and geometric models to estimate occluded joints [46], [44]. Another approach is the utilization of a library of known poses, known as anchor poses [55], but this could limit the generalization power of the model and the ability to handle unforeseen poses.

Motivated by advances in semantic segmentation architectures [14] and [68], and expanding upon state-of-the-art results obtained by UniPose [3] on 2D pose estimation, we propose UniPose+, an expanded and improved pose estimation framework for images and videos in both 2D and 3D, consisting of only a single stage and capable of obtaining accurate results without requiring postprocessing. A main component of our architecture is the multi-scale feature representation with the Waterfall Atrous Spatial Pooling (WASP) module which combines the cascaded approach for atrous convolution with the larger Field-of-View (FOV) obtained from the parallel configuration of the Atrous Spatial Pyramid Pooling (ASPP) module [13].

Our unified approach predicts the location of joints using contextual information due to the multi-scale approach used in our network. With our contextual approach, our network includes the information of the entire frame and, therefore, does not require post analysis based on statistical or geometric methods. Examples of pose estimation obtained with our UniPose+ method for both 2D and 3D are shown in Figure 1. The main contributions of this paper are the following.

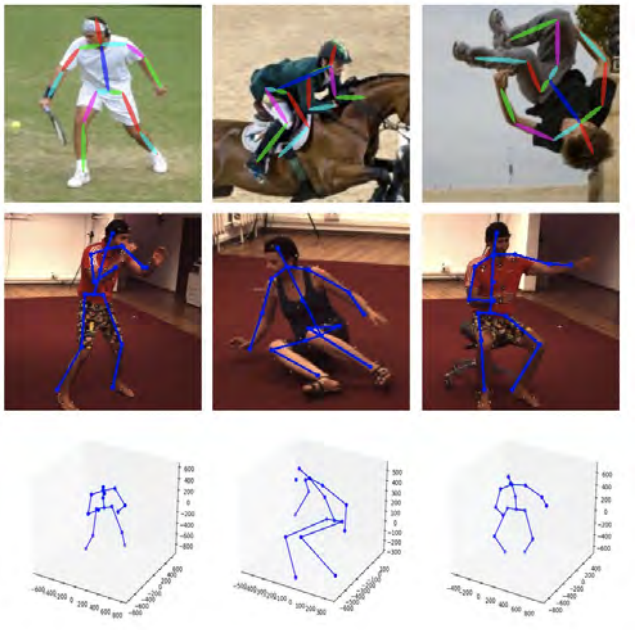

Fig. 1. 2D and 3D Pose estimation examples with our UniPose+ method.

- We propose the UniPose+ framework, a single-pass multi-scale approach that achieves state-of-the-art results for single person 2D and 3D human pose estimation in an end-to-end architecture incorporating depth regression into the pose estimation network.
- The UniPose+ framework achieves an increase in performance with Gaussian heatmap modulation of

• *Artacho and Savakis are with the Department of Computer Engineering, Rochester Institute of Technology, Rochester, NY, 14623.*

E-mail: bmartacho@mail.rit.edu; andreas.savakis@rit.edu

the multi-scale decoder for a more accurate representation of joint locations and reduction of the reprojection error during the decoder stage of the network, without requiring anchor poses or post processing.
- We extend the waterfall-based 2D UniPose+ approach to UniPose+LSTM by adopting a linear sequential Long Short-Term Memory (LSTM) configuration and obtain state-of-the-art results for temporal human pose estimation in video.

2 Related Work

Early works on human pose estimation from a single image focused on the detection of joints [51], [67]. In recent years, deep learning methods relying on Convolutional Neural Networks (CNNs) have achieved superior results [61], [62], [10], [59], [55]. The popular Convolutional Pose Machine (CPM) [62] proposed an architecture that refined joint detection via a set of stages in the network. Building upon [62], Yan et al. integrated the concept of Part Affinity Fields (PAF) resulting in the OpenPose method [10].

Multi-scale representation has been successfully used in backbone structures for pose estimation. Stacked hourglass (HG) networks [43] use cascaded structures of the hourglass method for the pose estimation task. Expanding on the hourglass structure, the multi-context approach in [19] relies on an hourglass backbone to perform pose estimation. The original backbone is augmented by the Hourglass Residual Units (HRU) with the goal of increasing the receptive FOV. Postprocessing with Conditional Random Fields (CRFs) is used to assemble the relations between detected joints. However, the drawback of CRFs is increased complexity that requires high computational power and reduces speed.

The High-Resolution Network (HRNet) [59] includes both high and low resolution representations. HRNet benefits from the larger FOV of multi resolution, a capability that we achieve in a simpler fashion with our WASP module. An analogous approach to HRNet is used by the Multi-Stage Pose Network (MSPN) [37], where the HRNet structure is combined with cross-stage feature aggregation and coarse-to-fine supervision.

More recently, the HRNet structure was combined with multi-resolution pyramids in [17] to further explore multi-scale features. The Distribution-Aware coordinate Representation of Keypoints (DARK) method [71] aims to reduce loss during the inference processing of the decoder stage when using an HRNet backbone.

Other works attempt to leverage contextual information into pose estimation. The Cascade Prediction Fusion (CPF) [72] uses graphical components in order to exploit the context for pose estimation. Similarly, the Cascade Feature Aggregation (CFA) [58] aims to use semantic information to detect pose with a cascade approach. Generative Adversarial Networks (GANs) were used in [11] to learn dependencies and contextual information for pose.

A drawback of some methods is that they require an independent branch for the detection of the bounding box of human subjects in the frame. LightTrack [45], for instance, relies on a separate YOLO [53] architecture to perform the detection of subjects prior to detecting joints. In a different feamework, LCR-Net [55] has different branches for the detection using Detectron [21] and the arrangement of joints during classification.

2.1 Depth Regression

Most 3D pose estimation methods rely on regression to generate 3D joint coordinates from 2D pose. Multi-scale approaches to depth estimation became popular for overcoming the loss of pooling [20]. Hao et al. [23] initially made use of atrous convolutions to access multiple scales for depth. Analogously, [27] implements a multi-scale approach with improved results by fusing feature scales, although it still lacks in precision for more complex objects. Other methods that use multi-scales include [15] and [65] which combines the multi-scale approach with CRFs. Several networks rely on leveraging information from the backbone to perform both 2D pose and depth tasks in multi-scale approaches [64] and [30].

2.2 Temporal Pose Estimation

For the task of pose estimation in videos, most methods do not account for the temporal component and process each frame independently. An additional challenge is the occasional blurring resulting from the movement of the humans in the video. The main incentive for developing a pose estimation method with a temporal component is to better estimate joints during blurring or occlusion conditions using information from previous frames.

Deepflow [63] used optical flow to better connect predictions between frames. Another method that utilized optical flow is Thin-Slicing [57], relying on both optical flow and a spatio-temporal model. However, the increased complexity of this model results in a significant increase in computational cost. The Chained Model [22] utilizes recurrent networks to incorporate the temporal component. A similar concept was adopted by the LSTM Pose Machine [41] approach, where the LSTM was used to augment memory in the network.

2.3 Multi-Scale Feature Representations

A challenge with CNN-based semantic segmentation and pose estimation methods is the significant reduction of resolution caused by pooling. Fully Convolutional Networks (FCN) [39] addressed this problem by deploying upsampling strategies across deconvolution layers that increase the feature maps back to the dimensions of the input image.

In semantic segmentation, dilated or atrous convolutions [13] are used to increase the size of the receptive fields in the network and avoid downsampling in a multi-scale framework. The ASPP approach assembles atrous convolutions in four parallel branches with different rates, that are combined by fast bilinear interpolation with an additional factor of eight. This configuration recovers the feature maps in the original image resolution. The increase in resolution and FOV in the ASPP network can be beneficial for a contextual detection of body parts during pose estimation.

The WASP module incorporates multi-scale features without immediately parallelizing the input stream [4], [3]. Instead, it creates a waterfall flow by first processing through a filter and then creating a new branch. WASP

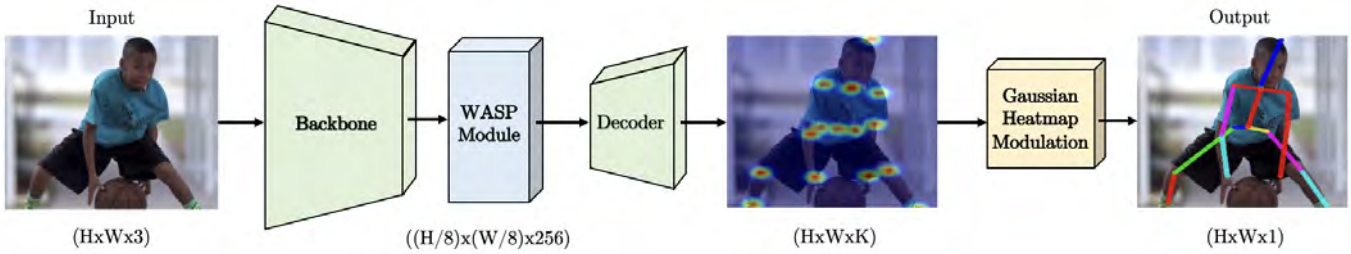

Fig. 2. UniPose+ architecture for single frame 2D pose detection. The input color image of dimensions (HxW) is fed through the backbone and WASP module to obtain 256 feature channels. The decoder module generates K heatmaps, one per joint.

also goes beyond the cascade approach by combining the streams from all its branches and average pooling of the original input to achieve a multi-scale representation.

2.4 3D Pose Estimation

The Localization, Classification, and Regression network (LCR-Net) [55] extends pose estimation to 3D space via depth regression. LCR-Net relies on a Detectron backbone [21] for the detection of human joint locations. From these locations, the method finds the best fit to predefined anchor poses for the detected human poses. Finally, LCR-Net performs a regression to estimate 3D coordinates in the image. A drawback of this method is the limited set of anchor poses available, which impose a limitation for the estimation of unforeseen poses.

Similarly, PandaNet [6] also relies on anchor poses. An expansion to the LCR-Net architecture was proposed by LCR-Net++ [56]. This version improves pose estimation performance by using additional synthetic training data.

Aiming to better associate joints into the 3D skeleton without anchor poses, [32] relies on an autoencoder to learn a latent pose representation and accounts for joint dependencies. It also applies LSTM to exploit temporal consistencies between frames.

The MonoCap method for 3D human capture [75] couples a CNN with a geometric prior in order to statistically determine the third dimension for pose using the Expectation-Maximization algorithm. Similarly, [76] applies a geometric constraint to regularize 3D predictions, exploiting the correlations between 2D and 3D poses.

The baseline for 3D pose method [42] applies a feed-forward network to overcome the errors associated with the regression from 2D to 3D. The same regression is achieved by [12] by combining state-of-the-art 2D pose estimation architecture with a MoCap library with 3D pose data. Pavlakos et al. [49] focused on refining of the coarse pose estimation data through the network in order to reduce error in the pose estimation. Further refinement of joint detections was explored by [48] with the use of a weakly supervised signal from ordinal depths to overcome the lower availability of 3D labelled images and achieve a competitive performance with CNN trained with accurate 3D joint coordinates.

Several approaches rely on the use of multiple cameras for the geometric inference of the 3D pose through triangulation. The work in [54] computes the FOV from the camera angles during training to estimate the 3D pose via a direct linear transform. Another approach to 3D pose is to rely on part-specific architectures with architecture search [16].

An approach to the reconstruction of 3D poses is to generate a mesh representation of the human body. The Skinned Multi-Person Linear Model (SMPL) [40] estimates the 3D model based on the skinning and blending of images from 3D body scans. Following SMPL, the SPIN method [35] uses a supervised network to learn the SMPL model of the human body during training iterations. The Video Inference for Body Pose (VIBE) method [34] trains the SMPL structure to learn the body model and interpret the statics, physics, and kinetics of the human body in videos. Aiming to leverage the human body mechanics, [66] uses the kinematics structure of the human body to simplify the body structure.

Similarly, Pavllo et al. [50] applies dilated temporal convolutions to correlate 2D keypoint detections between frames. The method applies the concept of back-projection to train the model with unlabeled video data in a semi-supervised fashion. Further, Cai et al. [9] combined the information of temporal consistencies with the domain knowledge of the human body using a combination of multi-scale features and a graph-based representation to estimate 3D pose in a sequence of frames.

To overcome the challenge of 3D pose estimation of multiple targets in video, [25] applied LSTM to create a sequence-to-sequence network that creates temporal constraints between frames. In another approach, [69], applies trajectory optimization and the Hungarian method to resolve the 3D temporal assignment of individual poses. MubyNet [70] aimed to optimize the occurrence of multiple people for 3D pose by fusing the extracted information from attention maps and the deep-auto-encoders for the multitask of localization and grouping of people.

3 UniPose+ Architecture

We propose the UniPose+ framework, a unified framework for human pose estimation tasks including 2D and 3D pose estimation in images or videos. Improving upon previous works, UniPose+ does not require separate branches for bounding box and joint detections, and simultaneously estimates 2D and 3D pose in an end-to-end architecture with shared backbone. The UniPose+ framework performs a unified detection of the bounding box and joints of a person, as well as regression for the 3D coordinates of the joints. Building upon the UniPose method for 2D pose [3], the

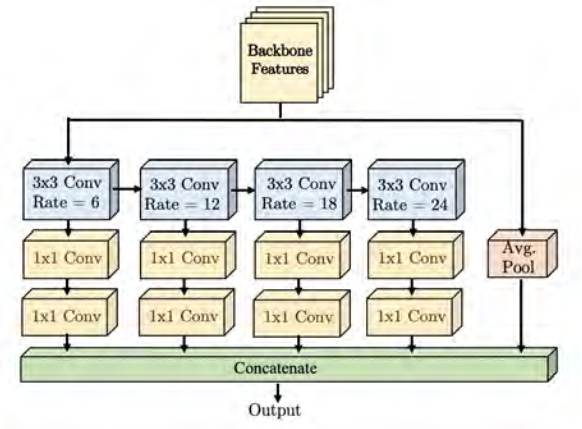

Fig. 3. Waterfall architecture in the WASP module [4]. The inputs to the WASP module are 1280 channels of ResNet features maps.

UniPose+ framework includes the multi-scale "Waterfall" configuration and further improves the efficiency of joint detection with the incorporation of Gaussian heatmap modulation at the decoder stage.

The first configuration of UniPose+ is the framework for 2D pose estimation in single images, which provides increased accuracy over UniPose [3]. The processing pipeline is shown in Figure 2. The input image is initially fed into a deep CNN backbone. Unipose+ is a backbone agnostic framework. To demonstrate its capability of improving accuracy in a wide variety of backbones, we implement the UniPose+ framework to accommodate a variety of backbones including HRNet [59], a modified ResNet-101 [24], and SENet-152 [26].

The resultant feature maps are processed by a decoder network that generates K heatmaps, one for each joint, with the corresponding probability distributions obtained from Softmax. Then the decoder performs bilinear interpolation to recover the original resolution, followed by a local max operation to localize the joints for 2D pose estimation. The decoder in our network generates detections for both visible and occluded joints. Additionally, the decoder generates a bounding box detection without the use of postprocessing or independent parallel branches.

The incorporation of the multi-level approach via the WASP module and the Gaussian heatmap modulation during interpolation allows the UniPose+ framework to more widely explore feature representations without incorporating a larger backbone, such as the deeper ResNet-152, or a heavier multi-stage architecture, such as Hourglass. The modularity of the UniPose+ framework enables easier implementation for reproducibility, and natural expansion to 3D pose estimation.

3.1 WASP Module

The WASP module generates an efficient multi-scale representation that helps UniPose+ to achieve state-of-the-art results. The WASP architecture, shown in Figure 3, is designed to leverage both the larger FOV of the ASPP configuration and the reduced size of the cascade approach.

WASP relies on atrous convolutions to maintain a large FOV, performing a cascade of atrous convolutions at increasing rates to gain efficiency. In contrast to ASPP, WASP does not immediately parallelize the input stream. Instead, it creates a waterfall flow by first processing through a filter and then creating a new branch. In addition, WASP goes beyond the cascade approach by combining the streams from all its branches and average pooling of the original input to achieve a multi-scale representation. The WASP module output f_{WASP} is defined by the equation:

$$f_{WASP} = K_1 \circledast (\sum_{i=1}^{4} K_1 \circledast (K_1 \circledast (K_{d_i} \circledast f_{i-1})) + AP(f_0)) \quad (1)$$

where $\circledast$ represents convolution, f_0 is the input feature map, f_i is the feature map resulting from the i^{th} atrous convolution, AP is the 2D global average pooling operation through the channels with filter dimension and stride of 1, K_1 and K_{d_i} represent convolutions of kernels 1×1 and 3×3 with dilations of $d_i = [6, 12, 18, 24]$, respectively. All feature maps from the 4 branches are concatenated with the 2D average pooling branch with pooling and kernel size equal to one, averaging their channel dimension, resulting in 1,280 channels. The last convolution of kernel size 1 brings the number of feature maps down to 256.

3.2 Decoder Module for 2D pose

Our 2D decoder module converts the score maps from the WASP module to heatmaps corresponding to body joints and the bounding box. Figure 4 shows the decoder architecture for an input color image of size (1280×720). The decoder receives feature maps from WASP and low level feature maps from the first block of the backbone. After a max pooling operation to match the dimensions of the inputs, the feature maps are concatenated and processed through convolutional layers, dropout layers, bilinear interpolation to resize to the original input size and Gaussian heatmap modulation to select the peak.

3.3 Gaussian Heatmap Modulation

Conventional interpolation or upsampling methods for the decoding stage of the network result in an inevitable loss in resolution and consequently accuracy, limiting the potential of the network. Motivated by recent results with distribution aware modulation [71], we include Gaussian heatmap modulation in our decoder module for training, validation, and inference. The implementation of the Gaussian interpolation allows the network to achieve sub-pixel resolution for peak localization following the anticipated Gaussian pattern of the feature response. This method results in a smoother response and more accurate peak prediction for joints, by eliminating false positives in noisy responses during the joint detection.

We utilize a convolution operation of the interpolated features map f_D with a Gaussian kernel K, shown in Equation (2), aiming to approximate the response shape to the expected label of the dataset during training.

$$f_G = K \circledast f_D \quad (2)$$

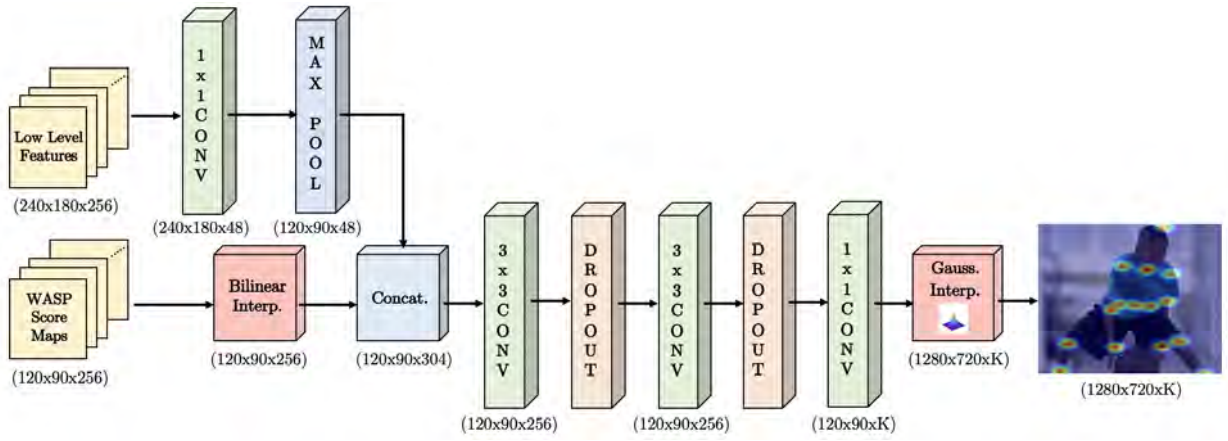

Fig. 4. Decoder module used in the UniPose+ pipeline. Assuming original image dimensions of (1280×720), the inputs to the decoder are the channels from low level features layer of the backbone and channels of the WASP feature maps. The bilinear interpolation is used to bring the high level feature dimensions to match the lower level features dimensions depending on the backbone selected. The output of the decoder is K heatmaps corresponding to K joints, shown in the image example. Additionally, the decoder outputs heatmaps for the bounding box (not shown in the image).

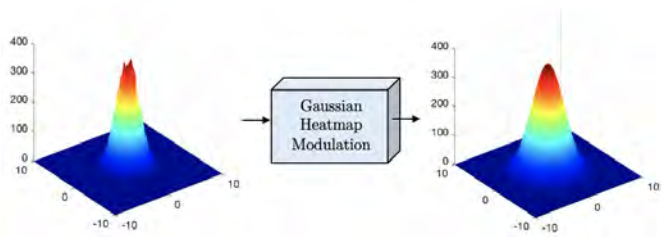

Fig. 5. Illustration of the Gaussian heatmap modulation process for feature maps following the interpolation in the decoder.

where f_G represent the feature maps after the Gaussian convolution operation. The behavior is learned and reproduced by the network during validation and inference.

Following convolution with the Gaussian kernel, the modulation of the interpolation output is scaled to f_{G_s} by mapping f_G to the range of the response of the original feature map f_D using:

$$f_{G_s} = \frac{f_G - min(f_G)}{max(f_G) - min(f_G)} * max(f_D). \quad (3)$$

Our Gaussian heatmap modulation approach allows for better localization of the coordinates during interpolation, by overcoming the quantization error inherited from the increase in resolution. Figure 5 demonstrates the modularization of a feature map response used by UniPose+.

3.4 UniPose+LSTM for Pose Estimation in Video

The UniPose architecture was modified to UniPoseLSTM for pose estimation in videos [3]. For video processing, it is useful to leverage the similarities and temporal correlations between consecutive frames. To operate in video processing mode, the UniPose+ architecture is augmented by an LSTM module that receives the final heatmaps from the previous frame along with the decoder heatmaps from the current frame. The pipeline of UniPose+LSTM is shown in Figure 6. This network includes CNN layers following the LSTM to generate the final heatmaps used for joint detection.

The UniPose+LSTM configuration allows the network to use information from the previously processed frames, without significantly increasing the total size of the network. For both the single image and video configurations, our network uses identical ResNet-101 backbone, WASP module, and decoder. We evaluated the performance benefits due to the temporal length of the memory component, when using an LSTM for several frames. It was experimentally determined that accuracy improves when incorporating up to 5 frames in the LSTM, and a plateau in accuracy was observed for additional frames.

3.5 UniPose3D for 3D Pose Estimation

We extend the UniPose+ framework to perform 3D pose estimation from monocular images. We propose UniPose3D, an end-to-end unified architecture for both 2D and 3D pose estimation that does not require anchor poses. Our 3D regression approach, inspired by [76], is based on depth regression using multi-scale representations and 2D joint coordinates.

The UniPose3D processing pipeline is shown in Figure 7. Our proposed methodology is composed of the 2D UniPose method combined with a depth regression module. After the 2D coordinates for the joint locations are determined, the additional depth dimension is estimated, resulting in a concatenated output of pixel coordinates and depth.

The input image is initially processed through the backbone to extract high-level and low-level features. The output feeds its high-level features to the WASP module, followed by concatenating the low level-features with the WASP module output. The resultant feature maps are processed by a short decoder network that generates K heatmaps for the 2D pose estimation output, one for each joint, with the corresponding probability distributions obtained from Softmax. The short 2D decoding stage is followed by a 3D regression stage that extracts the depth estimation for the joints and generates a 3D pose detection without the requirement of anchor poses.

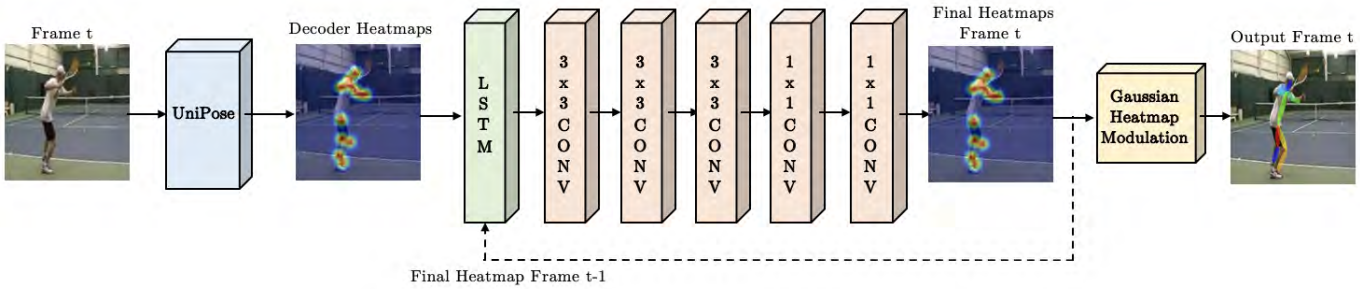

Fig. 6. UniPose+LSTM architecture for pose estimation in videos. The joint heatmaps from the UniPose+ decoder are fed into the LSTM along with the final heatmaps from the previous LSTM state. The convolutional layers following the LSTM reorganize the outputs into the final heatmaps used for joint localization.

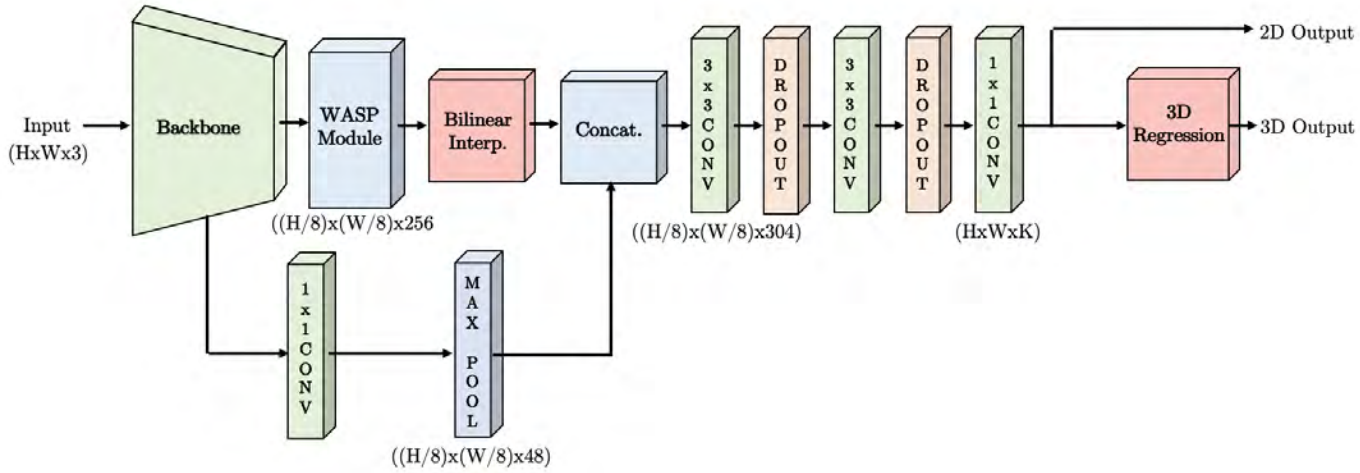

Fig. 7. UniPose3D architecture for 3D pose estimation. The input color image of size (H×W) is fed through the backbone and WASP module to obtain feature channels at reduced resolution by a factor of 8. The bilinear interpolation is used to bring the high level feature dimensions to match the lower level features dimensions depending on the backbone selected. The concatenation of the WASP output and low level features from the backbone are fed in the short decoder and 3D regression module. The decoder generates K heatmaps, one per joint for the 2D pose estimation at the original resolution. For the specific case of the Human3.6M dataset, there are K=17 joints. The 3D regression branch outputs the 3D pose estimation.

3.6 Depth Regression Module

For depth regression, we utilize a combination of heatmaps for 2D joints with lower level features extracted from the backbone and fed in our 2D decoder. The integration of the 2D joint detections with intermediate feature maps and multi-level features from the WASP module allows a more complete representation of the semantic information form the backbone. The 3D depth regression utilizes a loss, similar to [76], that is a regression with Euclidean loss for the 3D component given as follows.

$$L_{depth} = \lambda_{reg}||Y_{dep} - \hat{Y}_{dep}||^2 \quad (4)$$

where L_{depth} is the depth regression loss and λ_{reg} is the regularization term for regression loss.

4 Datasets

We performed experiments on the following datasets. Two datasets are composed of single images: Leeds Sports Pose (LSP) [31] and MPII [2]; one dataset consists of video sequences: Penn Action [73]; and the Human3.6M dataset is used for 3D pose estimation. The Leeds Sports Pose (LSP) dataset [31] was initially used for single person pose estimation. Images for LSP were collected from Flickr for a variety of individuals performing sports activities. The dataset is composed of 1,000 images for training and 1,000 images for testing with 14 labelled keypoints in the entire body. The LSP dataset includes lower variation in the data, allowing a good initial assessment of the network performance for the task of single person pose estimation.

The MPII [2] dataset contains approximately 25,000 images of annotated body joints of over 40,000 subjects. The images are collected from YouTube videos in 410 everyday human activities. The dataset contains frames with 2D and 3D joints annotations, head and torso orientations, and body part occlusions. Another feature of the MPII dataset is that it contains previous and following frames, although it lacks labelling for those frames.

The Penn Action [73] dataset contains 2,326 video sequences of 15 different activities including sports, athletic activities, and playing instruments. The dataset was used to evaluate the performance of our architecture for temporal

pose estimation and joint tracking, i.e., the estimation of pose in a frame while contextually using previous detections to refine the result.

Human3.6M dataset [28] is a large scale dataset for 2D and 3D pose estimation training and testing. The dataset consists of 3.6 million human poses captured in a controlled laboratory environment with 11 actors performing a set of 17 different everyday actions. The images are extracted from videos captured from 4 cameras in different positions from the front and back of the individuals using a MoCap system.

In order to better train our network for joint detection, ideal Gaussian maps were generated at the joint locations in the ground truth. These maps are more effective for training than single points at the joint locations, and they train our network to generate Gaussian heatmaps corresponding to the location of each joint. Gaussians with different σ values were considered and a value of $\sigma = 3$ was adopted, resulting in a well defined Gaussian curve for both the ground truth and predicted outputs. This value of σ also allows enough separation between joints in the image.

5 Experiments

The training, validation and testing of UniPose+ was based on the procedures and metrics outlined in each dataset. For LSP and MPII datasets, data augmentation during training included horizontal flip, rotation, and random crop.

For 3D pose estimation, similarly to [76], utilizing a network pre-trained on 2D pose estimation was found to be more effective compared to random initialization for training. We performed pre-training of the UniPose3D method for the specific task of 2D pose using the MPII dataset. Using the weights trained for the 2D task, we then incorporate the 3D regression module to the architecture resulting in a closer representation of the 3D pose estimation.

The training procedure for UniPose3D consists of initially training the network on 140 epochs for 2D pose estimation on the MPII dataset, following the first step of the protocol proposed by [76]. We then trained UniPose+ for 60 epochs on the Human3.6M dataset without the use of any geometric constraints.

The training process in [76] continues training the network for additional 15 epochs with the introduction of the geometric constraint induced loss for regularization of depth prediction. The geometric constraint loss is calculated for the length l_i of each limb i. The limb lengths are normalized in the Human3.6M dataset, obtaining the normalized value from $\overline{l}_i$.

$$L_{geo} = \sum_i \frac{1}{|l_i|} \sum_{j \in l_i} (\frac{l_j}{\overline{l}_j} - \overline{r}_j)^2 \tag{5}$$

where the average normalized length of the limb is given as

$$\overline{r}_i = \frac{1}{|l_i|} \sum_{j \in l_i} \frac{l_j}{\overline{l}_j}. \tag{6}$$

The new depth regression loss L_{depth} is then given by the following equation.

$$L_{depth} = \begin{cases} \lambda_{reg} ||Y_{dep} - \hat{Y}_{dep}||^2, & \text{for 3D regression} \\ \lambda_{geo} L_{geo}, & \text{for 2D joints} \end{cases} \tag{7}$$

where λ_{reg} is the regularization term for regression loss.

During the last stage of training, we did not observe an accuracy increase, and UniPose+ was able to achieve the best results without geometric constraints. We attribute this success to the effectiveness of the WASP module that is able to more accurately estimate 3D pose in a more robust multi-scale structure.

5.1 Metrics

For the evaluation of UniPose+, various metrics were used depending on previously reported results and the available ground truth for each dataset. The first metric used is the Percentage of Correct Keypoints (PCK). This metric considers the prediction of a keypoint correct when a joint detection lies within a certain threshold distance of the ground truth. Two commonly used thresholds were adopted. The first is PCK@0.2, which refers to a threshold of 20% of the torso diameter, and the second is PCKh@0.5, which refers to a threshold of 50% of the head diameter.

To compare our results with other methods for the Human3.6M dataset, we employed a downsampling protocol used by [76] for both training and testing from 50 fps to 10 fps, reducing the redundancy of the high frame video. In the evaluation approach used by [36], [77], [74], and [76], we set the subjects S1, S5, S6, S7, and S8 for training, and subjects S9 and S11 for testing.

The error is measured in mm by Mean per Joint Position Error (MPJPE) for the aligned joints. The 2D and 3D coordinates are aligned to the root joint (pelvis) by the conversion to the canonical skeleton follows procedures used by [49], [75], and [76].

$$\hat{Y} = (Y_{out} - Y_{out}^{root}) \left(\frac{Ave(Sum(l_{sk})}{l_{sk}} \right) + Y_{GT}^{root} \tag{8}$$

where Y_{out} is the aggregate of 2D and 3D joints, $\overline{l_{sk}}$ is the summation of the skeleton length, l_{sk} is the average total skeleton length of all subjects in the dataset, and Y_{GT}^{root} the ground-truth for the root joint.

5.2 Parameter Selection

We process the input image at its native resolution without resizing, in order to train the network with the most detail possible. For that reason, the batch size varied depending on the size of the dataset images. We considered different rates of dilation on the WASP module and found that larger rates result in better prediction. A set of dilation rates of $r = \{6, 12, 18, 24\}$ was selected for the WASP module.

The training procedure for UniPose3D adopts the pre-trained weights from MPII as the starting weights and follows both 2D and 3D detections for the Human3.6M dataset annotations for backpropagation.

For all datasets, we calculate the learning rate based on the step method, where the learning rate started at 10^{-4} and was reduced progressively by an order of magnitude at each step [38]. All experiments were performed using PyTorch 1.8 on Ubuntu 18.04. The workstation has an Intel i5-2650 2.20GHz CPU with 16GB of RAM and an NVIDIA Tesla V100 GPU.

Fig. 8. Pose estimation examples from the LSP dataset.

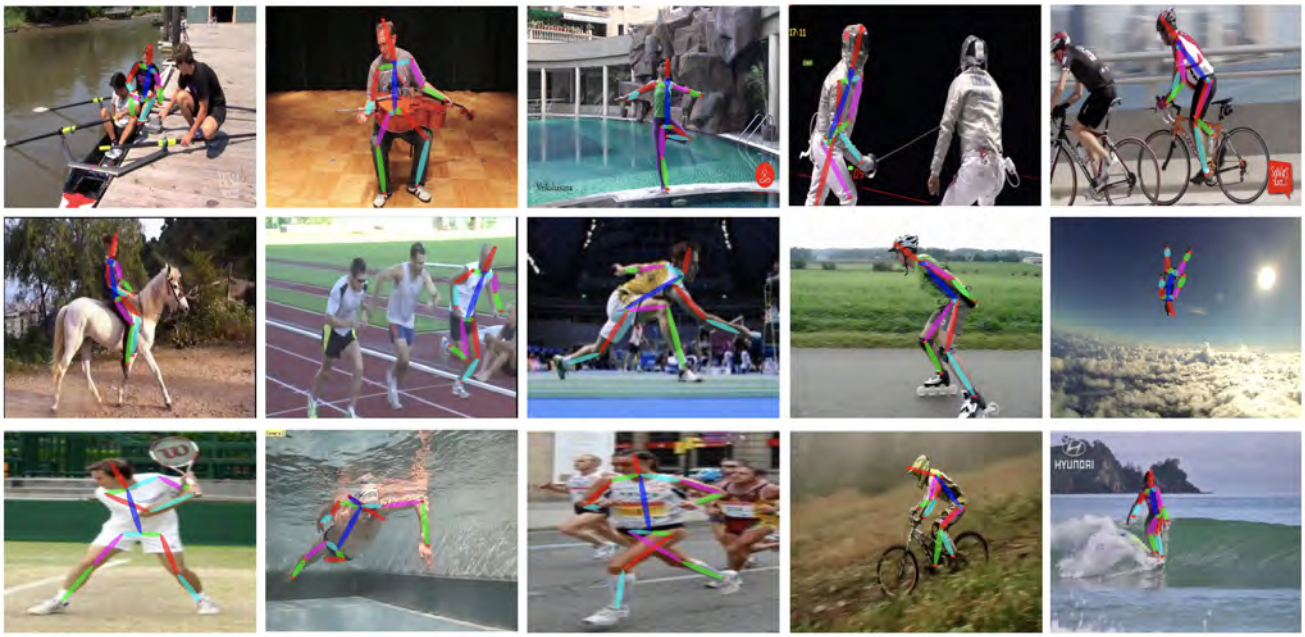

Fig. 9. Pose estimation examples from the MPII dataset.

5.3 Backbone Selection

The backbone agnostic UniPose+ framework performs pose estimation with high accuracy by enhancing the features through the waterfall module, which leverages information from the first and last blocks of the backbone. The inclusion of the WASP module in the framework improves the feature representations of the backbone without requiring significant computational effort and achieves increases in accuracy with low overhead during implementation.

We demonstrate the UniPose+ framework's robustness and flexibility due to its modular nature by considering three different backbones for feature extraction: ResNet [24], SENet [26], and HRNet [59]. Comparisons with these three backbones are provided in the next section, including ablation studies and analyses of the number of parameters and GFLOPs for each configuration. Our results show that the use of different backbones significantly impacts the accuracy and computational cost of the network. The most significant increases in performance are achieved when adopting the HRNet backbone, as demonstrated next by our results.

6 Results

6.1 Single Image 2D Pose Estimation

Initially we performed a series of ablation studies to investigate individual and combined improvements of components used in the UniPose framework. Table 1 demonstrates the results for the inclusion of the Gaussian deconvolution modulation (GDM) during interpolation, and the use of the WASP module [4] for multi-scale feature extraction. All results for Table 1 utilize ResNet-101 as its decoder.

Method	Params (M)	GFLOPs	GDM	WASP	PCK @0.2
UniPose [3]	44.3	27.8			93.4%
UniPose [3]	48.3	29.5		✓	94.5%
UniPose+	48.3	29.5	✓	✓	94.8%

TABLE 1
Results for the LSP dataset using different configurations of UniPose with ResNet-101 backbone. GDM represents the use of Gaussian Deconvolution Modulation and WASP indicates the use of the waterfall module in the network.

Our UniPose method progressively increases its performance with the addition of innovations, resulting in 1.5% improvement over the use of ResNet as a feature extractor, followed by the UniPose decoder. Most significantly, the use of the WASP module further improves the extraction of feature maps at different scales, increasing the accuracy of keypoints detection, particularly for occluded joints.

The UniPose+ framework was tested on the LSP dataset and compared with other methods, as shown in Table 2. UniPose+ achieved a PCK@0.2 of 94.8% when applying a ResNet backbone, and further increases its performance to a PCK@0.2 of 99.6% when applying when applying a more efficient multi-scale backbone, HRNet. UniPose+ shows gains in comparison to other approaches, including the original UniPose framework [3] and its versatility to be implemented with different backbones.

Differently than methods such as CPM, [62], UniPose+ is able to detect the body joints with high confidence in a single pass, instead of going through several stages of refinement in the network. Examples of pose estimation results for the LSP dataset are shown in Figure 8. It is noticeable from these examples that our method identifies the location of symmetric body joints with high precision. Challenging conditions include the detection of joints when limbs are not sufficiently separated or occlude each other, or for unusual upside down orientations. Most instances of error occur with the incorrect association of another person's joint, crossing detection between left and right joints (i.e. ankles) when there is occlusion, specifically for harder joints to detect, as ankles and wrists.

We next perform training and testing in the larger MPII dataset [2], focusing on single person detection. Since the MPII images may contain multiple people, we selected the center map of the main person to detect the pose of the correct individual. We used the WASPnet method [4] implementation for segmentation and detection of all the individuals, followed by the UniPose method to detect pose of the selected individual.

Table 3 shows the results for the MPII testing dataset. UniPose+ achieves a PCKh detection rate of 96.4% with an HRNet backbone and 92.9% with a ResNet-101 backbone, outperforming other methods for single person pose estimation in both configurations and surpassing previous results by UniPose [3].

Method	Backbone	Params (M)	GFLOPs	Extra Data	PCK @0.2
UniPose+	HRNet	50.4	**27.7**		**99.6%**
UniPose+	ResNet	48.3	29.5		**94.8%**
Gated Skip [7]	HG	26.0	33.5	✓	94.8%
UniPose+	SENet	55.6	118.8		**94.5%**
UniPose [3]	ResNet	48.3	29.5		94.5%
SAGAN [11]	HG (x2)	25.5	-	✓	94.3%
8-Stack HG [72]	HG	23.7	41.4		94.0%
PHR [8]	ResNet	-	-		90.7%
CPM [62]	-	31.4	163.7		90.5%
DeepCut [52]	VGG	-	-		87.1%
Recurrent [5]	-	15.4	-		85.2%

TABLE 2
Results for 2D pose estimation and comparison with other methods for the LSP dataset. The backbones used for different versions of UniPose+ and other methods are shown in the second column.

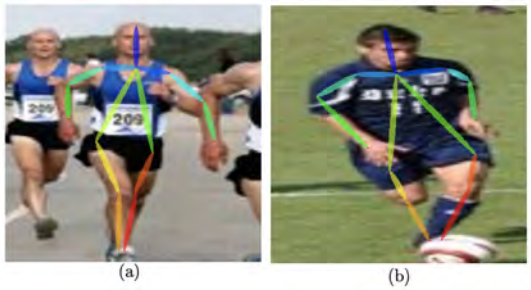

Fig. 10. Examples of fail cases for images in the LSP datasetp: (a) wrist location is misplaced due to multiple individuals; and (b) ankle location is misplaced due to occlusion.

The MPII dataset generally presents more common poses of people in everyday activities that mostly take place outdoors. The main difficulty with MPII is the presence of multiple people. Instances where there is not enough separation between the main person and other people resulted less accurate detections. Figure 9 demonstrates successful detections on the main person in MPII images. These examples illustrate that UniPose+ deals effectively with occlusion, e.g. in the case of the horse rider.

Method	Backbone	Params (M)	GFLOPs	PCKh @0.5
UniPose+	HRNet	50.4	27.7	**96.4%**
UniPose+	SENet	55.6	118.8	**94.3%**
UniPose+	ResNet	48.3	29.5	**92.9%**
UniPose [3]	ResNet	48.3	29.5	92.7%
MSPN [37]	ResNet (x4)	-	-	92.6%
8-Stack HG [72]	HG	23.7	41.4	92.5%
Tang et al. [60]	-	15.5	33.6	92.3%
SAGAN [11]	HG (x2)	25.5	-	92.3%
Structure-Aware [33]	-	-	-	92.0%
CFA [58]	HG/ResNet	-	-	90.0%
CPM [62]	-	31.4	163.7	88.5%

TABLE 3
Results for 2D pose estimation and comparison with other methods for the MPII dataset. The backbones used for different versions of UniPose+ and other methods are shown in the second column.

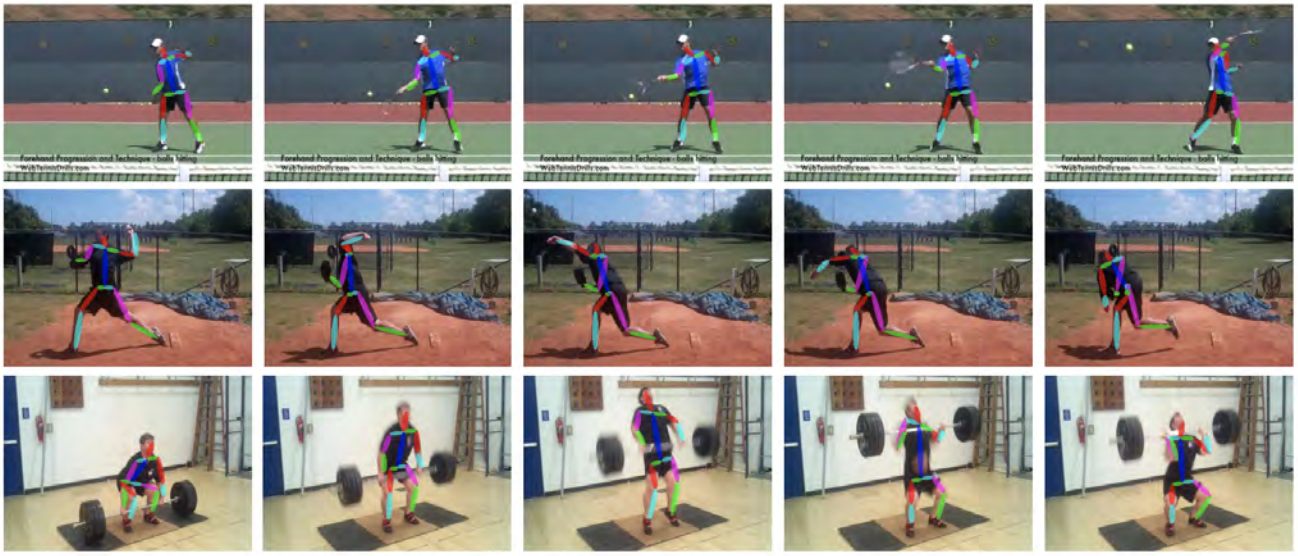

Fig. 11. Pose estimation examples from the Penn Action dataset for a sequence of frames.

Representative examples of fail cases are shown in Figure 10. Most of the joint detection errors were due to the presence of multiple people next to each other without sufficient separation of their limbs. Other types of errors were due to occlusion or when opposing limbs were very close to each other resulting in cross detection between the left and right joints of legs.

6.2 Multi-frame Pose Estimation

Table 4 shows the results for UniPose+LSTM in the Penn Action dataset [73]. Our results show a significant improvement over previous state-of-the-art methods by the application of UniPose+LSTM in the temporal mode with 5 consecutive frames. For this dataset, the results are reported as a correct detection when the predicted joint location lies within the provided bounding box, following the same procedure proposed by [67] and applied by [41]. Our method results in a 99.4% detection rate, a significant improvement of 1.6% over the next best result.

Method	PCK for Penn Action
UniPose+LSTM (ours)	**99.4%**
UniPoseLSTM [3]	99.3%
LSTM-PM [41]	97.7%
CPM [62]	97.1%
Thin-Slicing Network [57]	96.5%
N-best [47]	91.8%
ACPS [29]	81.1%

TABLE 4
Results for 2D pose estimation in a sequence of frames and comparisons with other methods for the Penn Action dataset.

Our UniPose+ network leverages the memory capability of the LSTM by incorporating 5 consecutive frames, found optimal in previous experiments [3]. This feature enables a higher detection rate and consequently a more robust architecture against motion blur and occlusions in the image. The PennAction dataset shows signs of saturation in performance, with UniPose+ achieving high accuracy scores. Due to its saturation, different backbone configurations for UniPose+ do not present a significant variation in performance, achieving the state-of-the-art PCK of 99.4% for both the HRNet and ResNet configurations of UniPose+.

Examples of detections for the Penn Action dataset [73] are shown in Figure 11. The examples selected are for fast motion scenarios showing every other frame in sequence, so that significant differences are observed between frames.

6.3 3D Pose Estimation

We performed training and testing of UniPose3D on the Human3.6M dataset [28] using monocular images. The network learned to infer depth for the human body, and obtained estimates of the 2D locations for joints. The final 3D pose estimation was obtained by the association of the 2D coordinates and depth, using intrinsic information from the cameras used for the dataset capture.

Analogously to the 2D experiments, we performed a series of ablation studies to investigate individual and combined improvements of components used in the UniPose+ framework for 3D pose estimation. Table 5 demonstrates the results for the inclusion of the GDM during interpolation, and the use of the WASP module [4] for multi-scale feature extraction. Our UniPose3D method progressively increases the performance as the innovations are included in the model, resulting in a significant total reduction of 16.77mm in error for the ResNet backbone and 16.34m for the HRNet backbone.

In contrast to 2D pose estimation methods, 3D pose estimation publications do not report values for the number of parameters used in their architecture or the total computational cost associate with processing. In order to better assess the computational cost and memory required in various methods, Table 6 shows the number of parameters and GFLOPs for the backbones used in 3D pose estimation.

We tested UniPose3D on the Human3.6M dataset using different backbone configurations. The results and com-

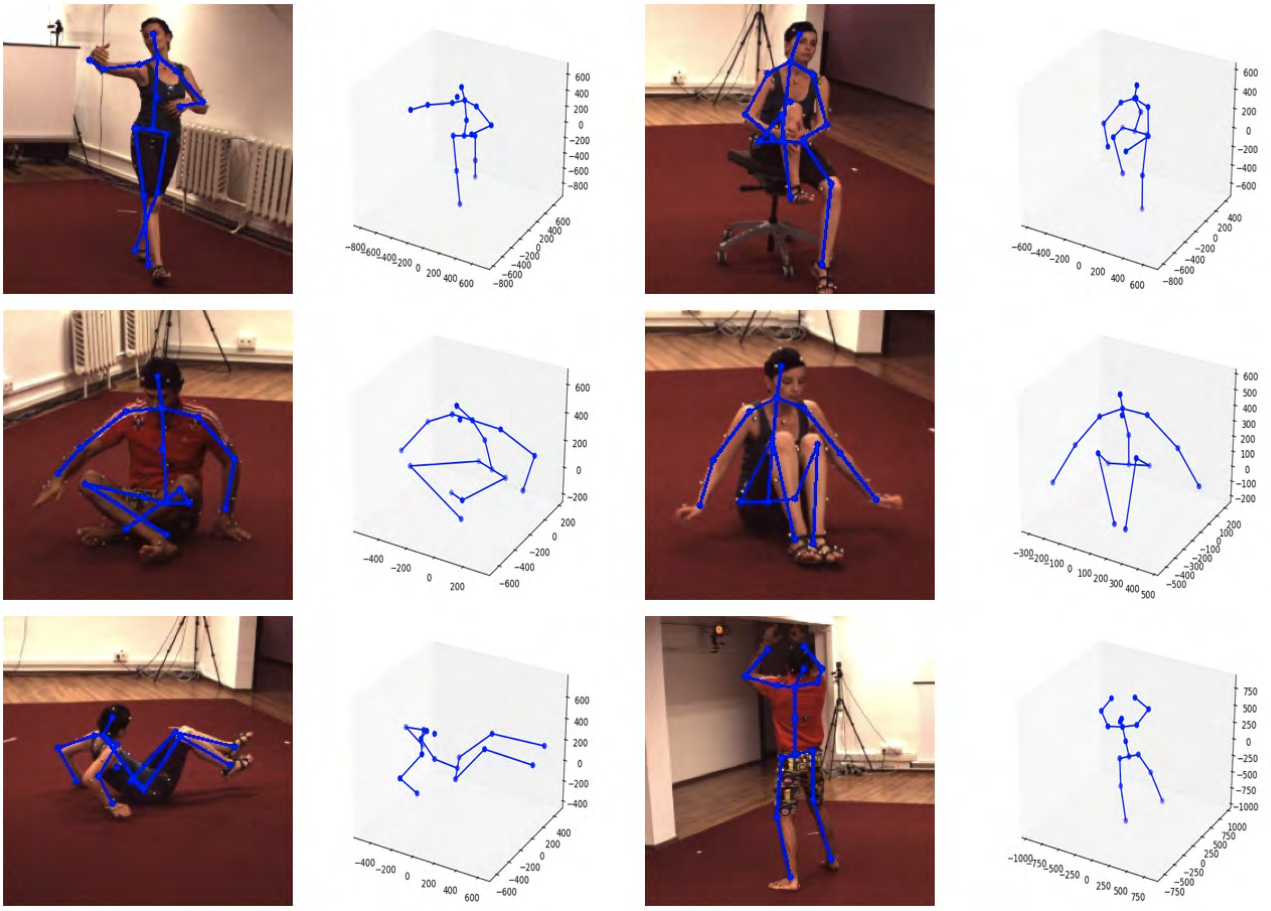

Fig. 12. Examples of 2D and 3D Pose estimation with UniPose3D on the Human3.6M dataset.

Method	Backbone	GDM	WASP	MPJPE
UniPose	ResNet-101			79.04
UniPose	ResNet-101		✓	65.86
UniPose+	ResNet-101	✓	✓	62.27
UniPose	HRNet			74.30
UniPose	HRNet		✓	61.92
UniPose+	HRNet	✓	✓	57.96

TABLE 5
Results for the Human3.6M dataset using different configurations of UniPose with ResNet backbone. GDM represents the use of Gaussian Deconvolution Modulation and WASP indicates the use of the waterfall module in the network.

Backbone	Params (M)	GFLOPs
ResNet	42.5	12.07
CPN	46.4	13.58
HRNet	68.1	22.49
SENet	113.2	26.88
2 Stack-HG	102.1	126.20
8 Stack-HG	395.3	445.44

TABLE 6
Comparison of parameters (in Millions) and floating point operations (GFLOPs) for backbones used for 3D pose estimation. All backbone measurements are reported for an input image of size 256×256×3.

parisons with state-of-the-art methods are shown in Table 7. UniPose3D achieved its best performance using HRNet as backbone, resulting in a MPJPE of 57.96mm. Using the SENet backbone, UniPose3D achieved a MPJPE of 61.66 mm when processing 256 features maps through the WASP module. This configuration corresponds to convolutions $f_i = 256$ in Equation (1) and is denoted as SENet-256 in Table 7. Increasing the number of feature maps to $f_i = 1280$ in the WASP module (SENet-1280 in Table 7) results in a significant performance boost, at the expense of computational demands, reducing the MPJPE to 59.81mm. Finally, when applying the same ResNet backbone used for UniPose [3], UniPose3D achieved a MPJPE of 62.27mm with a reduced size compared other configurations. Examples of 2D pose estimation and 3D pose regression for various poses are shown in Figure 12.

Differently than some of the comparison methods presented in Table 7, UniPose3D does not rely on the use of intermediate supervision during training or on the generation of additional depth data for further training. In addition, UniPose3D does not use information from multiple frames for its 3D pose estimation. The inclusion of either or both of these techniques modifies the comparison between methods, as the training and evaluation take place in different settings. Methods that rely on intermediate supervision during training and/or multi-frame information are illustrated in separate columns in Table 7.

Method	Backbone	Sup.	Multi-Frame	MPJPE
Xu et al. [66]	CPN	✓	T=1	49.20
Pavllo et al. [50]	CPN	✓		51.80
Cai et al. [9]	CPN		T=1	50.60
Hossain et al. [25]	HG		T=5	51.90
Pavlakos et al. [48]	HG	✓		56.20
UniPose3D	**HRNet**			**57.96**
UniPose3D	**SENet-1280**			**59.81**
LCR-Net+ [56]	LCR-Net			61.20
UniPose3D	**SENet-256**			**61.66**
UniPose3D	**ResNet-101**			**62.27**
Martinez et al. [42]	HG			62.90
LCR-Net++ [56]	LCR-Net			63.50
Zhou et al. [76]	HG	✓		64.90
Katircioglu et al. [32]	HG	✓		65.40
LCR-Net [55]	LCR-Net			65.40
VIBE [34]	ResNet-50	✓	T=16	65.60
Chen et al. [12]	CPM			66.92
Pavlakos et al. [49]	HG	✓		71.90

TABLE 7
Results for 3D Pose estimation and comparisons with other methods for the Human3.6M dataset with resolution of 256×256. "Sup." represents the use of intermediate supervision during training and "Multi-Frame" indicates that the model uses information from T=N frames or incorporates temporal components for training with a modified procedure for the dataset.

7 Conclusion

We presented the UniPose+ framework for 2D and 3D pose estimation in single images and videos. The UniPose+ pipeline utilizes a multi-scale features extractor and the WASP module that creates a waterfall flow with a cascade of atrous convolutions and multi-scale representations. The UniPose+ framework presents improved performance, with a more accurate response to the expected Gaussian response, with the introduction of the Gaussian heatmap modulation in the interpolation module.

The results of the UniPose+ framework demonstrated state-of-the-art performance on several datasets using various metrics. UniPose3D achieves state-of-the-art results for 3D pose estimation in an end-to-end architecture and simultaneously performs 2D pose extraction, without the requirement of anchor poses or postprocessing.

Our modular framework shows promise for further use in a broader range of applications, including multi-person 2D and 3D pose estimation.

Acknowledgments

This research was funded in part by National Science Foundation grant ♯1749376.

References

[1] R. Alp Guler, N. Neverova, and I. Kokkinos. "Densepose: Dense human pose estimation in the wild". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 7297–7306.

[2] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. "2D Human Pose Estimation: New Benchmark and State of the Art Analysis". In: *IEEE International Conference on Computer Vision*. 2014.

[3] B. Artacho and A. Savakis. "UniPose: Unified Human Pose Estimation in Single Images and Videos". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020.

[4] B. Artacho and A. Savakis. "Waterfall Atrous Spatial Pooling Architecture for Efficient Semantic Segmentation". In: *Sensors* 19.24 (2019), p. 5361. DOI: "https://doi.org/10.3390/s19245361".

[5] V. Belagiannis and A. Zisserman. "Recurrent human pose estimation". In: *IEEE International Conference on Automatic Face & Gesture Recognition*. 2017, pp. 468–475.

[6] A. Benzine, F. Chabot, B. Luvison, Q. C. Pham, and C. Achard. "PandaNet: Anchor-Based Single-Shot Multi-Person 3D Pose Estimation". In: *IEEE Conference on Computer Vision and Pattern Recognition*. June 2020.

[7] A. Bulat, J. Kossaifi, G. Tzimiropoulos, and M. Pantic. "Toward fast and accurate human pose estimation via soft-gated skip connections". In: *IEEE International Conference on Automatic Face and Gesture Recognition*. 2020.

[8] A. Bulat and G. Tzimiropoulos. "Human pose estimation via convolutional part heatmap regression". In: *European Conference on Computer Vision*. Springer. 2016, pp. 717–732.

[9] Y. Cai, L. Ge, J. Liu, J. Cai, T.-J. Cham, J. Yuan, and N. M. Thalmann. "Exploiting Spatial-Temporal Relationships for 3D Pose Estimation via Graph Convolutional Networks". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Oct. 2019.

[10] Z. Cao, T. Simon, S. .-.-E. Wei, and Y. Sheikh. "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2017.

[11] Z. Cao, R. Wang, X. Wang, Z. Liu, and X. Zhu. "Improving Human Pose Estimation With Self-Attention Generative Adversarial Networks". In: *IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. 2019, pp. 567–572.

[12] C.-H. Chen and D. Ramanan. "3d human pose estimation= 2d pose estimation+ matching". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 7035–7043.

[13] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and L. Yuille. "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution and Fully Connected CFRs". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.4 (2018), pp. 834–845.

[14] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. "Rethinking Atrous Convolution for Semantic Image Segmentation". In: *http://arxiv.org/abs/1602.06541* abs/1706.05587 (2017).

[15] X. Chen, X. Chen, and Z.-J. Zha. "Structure-aware residual pyramid network for monocular depth estimation". In: *CoRR arXiv:1907.06023*. 2019.

[16] Z. Chen, Y. Huang, H. Yu, B. Xue, K. Han, Y. Guo, and L. Wang. "Towards Part-aware Monocular 3D Human Pose Estimation: An Architecture Search Approach".

In: *European Conference on Computer Vision*. Springer, 2020.

[17] B. Cheng, B. Xiao, J. Wang, H. Shi, T. S. Huang, and L. Zhang. "HigherHRNet: Scale-Aware Representation Learning for Bottom-Up Human Pose Estimation". In: *IEEE Conference on Computer Vision and Pattern Recognition*. June 2020.

[18] Y. Cheng, B. Yang, B. Wang, W. Yan, and R. T. Tan. "Occlusion-Aware Networks for 3D Human Pose Estimation in Video". In: *IEEE International Conference on Computer Vision (ICCV)*. 2019.

[19] X. Chu, W. Yang, W. Ouyang, C. Ma, A. L. Yuille, and X. Wang. "Multi-Context Attention for Human Pose Estimation". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 1831–1840.

[20] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao. "Deep ordinal regression network for monocular depth estimation". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 2002–2011.

[21] R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár, and K. He. "Detectron". In: *https://github.com/facebookresearch/Detectron* (2018).

[22] G. Gkioxari, A. Toshev, and N. Jaitly. "Chained Predictions Using Convolutional Neural Networks". In: *CoRR arXiv:1605.02346*. 2016.

[23] Z. Hao, Y. Li, S. You, and F. Lu. "Detail preserving depth estimation from a single image using attention guided networks". In: *IEEE International Conference on 3D Vision (3DV)*. 2018, pp. 304–313.

[24] K. He, X. Zhang, S. Ren, and J. Sun. "Deep residual learning for image recognition". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 770–778.

[25] M. R. I. Hossain and J. J. Little. "Exploiting temporal information for 3d human pose estimation". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 68–84.

[26] J. Hu, L. Shen, and G. Sun. "Squeeze-and-excitation networks". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 7132–7141.

[27] J. Hu, M. Ozay, Y. Zhang, and T. Okatani. "Revisiting single image depth estimation: Toward higher resolution maps with accurate object boundaries". In: *IEEE Winter Conference on Applications of Computer Vision (WACV)*. 2019, pp. 1043–1051.

[28] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu. "Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36.7 (2014), pp. 1325–1339.

[29] U. Iqbal, M. Garbade, and J. Gall. "Pose for Action - Action for Pose". In: *IEEE Conference on Automatic Face and Gesture Recognition*. 2017.

[30] J. Jiao, Y. Cao, Y. Song, and R. Lau. "Look deeper into depth: Monocular depth estimation with semantic booster and attention-driven loss". In: *European Conference on Computer Vision (ECCV)*. 2018, pp. 53–69.

[31] S. Johnson and M. Everingham. "Clustered Pose and Nonlinear Appearance Models for Human Pose Estimation". In: *British Machine Vision Conference*. 2010.

[32] I. Katircioglu, B. Tekin, M. Salzmann, V. Lepetit, and P. Fua. "Learning latent representations of 3d human pose with deep neural networks". In: *International Journal of Computer Vision* 126.12 (2018), pp. 1326–1341.

[33] L. Ke, M.-C. Chang, H. Qi, and S. Lyu. "Multi-Scale Structure-Aware Network for Human Pose Estimation". In: *European Conference on Computer Vision (ECCV)*. 2018.

[34] M. Kocabas, N. Athanasiou, and M. J. Black. "Vibe: Video inference for human body pose and shape estimation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 5253–5263.

[35] N. Kolotouros, G. Pavlakos, M. J. Black, and K. Daniilidis. "Learning to reconstruct 3D human pose and shape via model-fitting in the loop". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 2252–2261.

[36] S. Li and A. B. Chan. "3d human pose estimation from monocular images with deep convolutional neural network". In: *Asian Conference on Computer Vision*. Springer. 2014, pp. 332–347.

[37] W. Li, Z. Wang, B. Yin, Q. Peng, Y. Du, T. Xiao, G. Yu, H. Lu, Y. Wei, and J. Sun. "Rethinking on Multi-Stage Networks for Human Pose Estimation". In: *CoRR arXiv:1901.00148* (2019).

[38] W. Liu, A. Rabinovich, and A. C. Berg. "ParseNet: Looking Wider to See Better". In: *CoRR*. Vol. abs/1506.04579. 2015.

[39] J. Long, E. Shelhamer, and T. Darrel. "Fully Convolutional Networks for Semantic Segmentation". In: *IEEE International Conference on Computer Vision* (2015).

[40] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. "SMPL: A Skinned Multi-Person Linear Model". In: *ACM Trans. Graphics (Proc. SIGGRAPH Asia)* 34.6 (Oct. 2015), 248:1–248:16.

[41] Y. Luo, J. S. J. Ren, Z. Wang, W. Sun, J. Pan, J. Liu, J. Pang, and L. Lin. "LSTM Pose Machines". In: *IEEE International Conference on Computer Vision*. 2018.

[42] J. Martinez, R. Hossain, J. Romero, and J. J. Little. "A simple yet effective baseline for 3d human pose estimation". In: *IEEE International Conference on Computer Vision*. 2017, pp. 2640–2649.

[43] A. Newell, K. Yang, and J. Deng. "Stacked Hourglass Networks for Human Pose Estimation". In: *European Conference on Computer Vision*. Springer. 2016, pp. 483–499.

[44] A. Nibali, Z. He, S. Morgan, and L. Prendergast. "3d Human Pose Estimation with 2D Marginal Heatmaps". In: *IEEE Winter Conference on Applications of Computer Vision (WACV)*. 2019, pp. 1477–1485.

[45] G. Ning and H. Huang. "LightTrack: A Generic Framework for Online Top-Down Human Pose Tracking". In: *CoRR arXiv:1905.02822* (2019).

[46] G. Papandreou, T. Zhu, L.-C. Chen, S. Gidaris, J. Tompson, and K. Murphy. "PersonLab: Person Pose Estimation and Instance Segmentation with a Bottom-Up, Part-Based, Geometric Embedding Model". In: *European Conference on Computer Vision (ECCV)*. 2018.

[47] D. Park and D. Ramanan. "N-best maximal decoders for part models". In: *IEEE International Conference on Computer Vision*. 2011, pp. 2627–2634.

[48] G. Pavlakos, X. Zhou, and K. Daniilidis. "Ordinal Depth Supervision for 3D Human Pose Estimation". In: *Computer Vision and Pattern Recognition (CVPR)*. 2018.

[49] G. Pavlakos, X. Zhou, K. G. Derpanis, and K. Daniilidis. "Coarse-to-fine volumetric prediction for single-image 3D human pose". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 7025–7034.

[50] D. Pavllo, C. Feichtenhofer, D. Grangier, and M. Auli. "3d human pose estimation in video with temporal convolutions and semi-supervised training". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 7753–7762.

[51] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele. "Poselet Conditioned Pictorial Structures". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2013, pp. 588–595.

[52] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, and B. Schiele. "Deepcut: Joint subset partition and labeling for multi person pose estimation". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 4929–4937.

[53] J. Redmon and A. Farhadi. "YOLOv3: An Incremental Improvement". In: *CoRR arxiv:1804.02767* (2018).

[54] E. Remelli, S. Han, S. Honari, P. Fua, and R. Wang. "Lightweight Multi-View 3D Pose Estimation Through Camera-Disentangled Representation". In: *IEEE Conference on Computer Vision and Pattern Recognition*. June 2020.

[55] G. Rogez, P. Weinzaepfel, and C. Schmid. "LCR-Net: Localization-Classification-Regression for Human Pose". In: *IEEE Conference on Computer Vision and Pattern Recognition*. Honolulu, HI, USA, 2017.

[56] G. Rogez, P. Weinzaepfel, and C. Schmid. "LCR-Net++: Multi-person 2d and 3d pose detection in natural images". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2019).

[57] J. Song, L. Wang, L. V. Gool, and O. Hilliges. "Thin-Slicing Network: A Deep Structured Model for Pose Estimation in Videos". In: *IEEE International Conference on Computer Vision* (2017).

[58] Z. Su, M. Ye, G. Zhang, L. Dai, and J. Sheng. "Cascade Feature Aggregation for Human Pose Estimation". In: *CoRR arXiv:1902.07837* (2019).

[59] K. Sun, B. Xiao, D. Liu, and J. Wang. "Deep High-Resolution Representation Learning for Human Pose Estimation". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2019.

[60] W. Tang, P. Yu, and Y. Wu. "Deeply Learned Compositional Models for Human Pose Estimation". In: *European Conference on Computer Vision (ECCV)*. 2018.

[61] A. Toshev and C. Szegedy. "Deeppose: Human Pose Estimation via Deep Neural Networks". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 1653–1660.

[62] S. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. "Convolutional Pose Machines". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2016.

[63] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. "DeepFlow: Large Displacement Optical Flow with Deep Matching". In: *IEEE International Conference on Computer Vision*. 2013, pp. 1385–1392.

[64] D. Xu, W. Ouyang, X. Wang, and N. Sebe. "Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 675–684.

[65] D. Xu, E. Ricci, W. Ouyang, X. Wang, and N. Sebe. "Multi-scale continuous crfs as sequential deep networks for monocular depth estimation". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 5354–5362.

[66] J. Xu, Z. Yu, B. Ni, J. Yang, X. Yang, and W. Zhang. "Deep kinematics analysis for monocular 3d human pose estimation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 899–908.

[67] Y. Yang and D. Ramanan. "Articulated Human Detection with Flexible Mixtures of Parts". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.12 (2012), pp. 2878–2890.

[68] F. Yu and V. Koltun. "Multi-Scale Context Aggregation by Dilated Convolutions". In: *International Conference on Learning Representations*. 2016.

[69] A. Zanfir, E. Marinoiu, and C. Sminchisescu. "Monocular 3d pose and shape estimation of multiple people in natural scenes-the importance of multiple scene constraints". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 2148–2157.

[70] A. Zanfir, E. Marinoiu, M. Zanfir, A.-I. Popa, and C. Sminchisescu. "Deep network for the integrated 3d sensing of multiple people in natural images". In: *Advances in Neural Information Processing Systems* 31 (2018), pp. 8410–8419.

[71] F. Zhang, X. Zhu, H. Dai, M. Ye, and C. Zhu. "Distribution-Aware Coordinate Representation for Human Pose Estimation". In: *IEEE Conference on Computer Vision and Pattern Recognition*. June 2020.

[72] H. Zhang, H. Ouyang, S. Liu, X. Qi, X. Shen, R. Yang, and J. Jia. "Human Pose Estimation with Spatial Contextual Information". In: *CoRR arXiv:1901.01760* (2019).

[73] W. Zhang, M. Zhu, and K. G. Derpanis. "From actemes to action: A strongly-supervised representation for detailed action understanding". In: *IEEE International Conference on Computer Vision*. 2013, pp. 2248–2255.

[74] X. Zhou, M. Zhu, S. Leonardos, K. G. Derpanis, and K. Daniilidis. "Sparseness meets deepness: 3D human pose estimation from monocular video". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 4966–4975.

[75] X. Zhou, M. Zhu, G. Pavlakos, S. Leonardos, K. G. Derpanis, and K. Daniilidis. "MonoCap: Monocular human motion capture using a CNN coupled with a

geometric prior". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41.4 (2018), pp. 901–914.

[76] X. Zhou, Q. Huang, X. Sun, X. Xue, and Y. Wei. "Towards 3d human pose estimation in the wild: a weakly-supervised approach". In: *IEEE International Conference on Computer Vision*. 2017, pp. 398–407.

[77] X. Zhou, X. Sun, W. Zhang, S. Liang, and Y. Wei. "Deep kinematic pose regression". In: *European Conference on Computer Vision*. Springer. 2016, pp. 186–201.

Bruno Artacho received his B.Eng from Sao Paulo State University (2015) and his M.Eng. from Memorial University of Newfoundland (2017), both in Electrical Engineering. Artacho worked at Google and Amazon developing methods and applying his research on Pose Estimation and Segmentation for a diverse set of products; and Transport Canada as part of the Unmanned Aerial System Task Force to assess risk and update the Canadian Air Traffic Policy. He is currently pursuing his Ph.D. in Engineering at the Rochester Institute of Technology in Rochester, NY. His research interests include Computer Vision, Machine Learning, and Human Pose Estimation.

Andreas Savakis is Professor of Computer Engineering and Director of the Center for Human-aware Artificial Intelligence (CHAI) at Rochester Institute of Technology (RIT). He received his Ph.D. in Electrical and Computer Engineering from North Carolina State University. Prior to joining RIT, he was Senior Research Scientist at Kodak Research Labs. His research interests include computer vision, deep learning, machine learning, domain adaptation, object tracking, human pose estimation, and scene analysis. Dr. Savakis has coauthored over 120 publications and is co-inventor on 12 U.S. patents.