UniPose: Unified Human Pose Estimation in Single Images and Videos

Bruno Artacho   Andreas Savakis

Rochester Institute of Technology

Rochester, NY

bmartacho@mail.rit.edu   andreas.savakis@rit.edu

Abstract

We propose UniPose, a unified framework for human pose estimation, based on our "Waterfall" Atrous Spatial Pooling architecture, that achieves state-of-art-results on several pose estimation metrics. UniPose incorporates contextual segmentation and joint localization to estimate the human pose in a single stage, with high accuracy, without relying on statistical postprocessing methods. The Waterfall module in UniPose leverages the efficiency of progressive filtering in the cascade architecture, while maintaining multi-scale fields-of-view comparable to spatial pyramid configurations. Additionally, our method is extended to UniPose-LSTM for multi-frame processing and achieves state-of-the-art results for temporal pose estimation in Video. Our results on multiple datasets demonstrate that UniPose, with a ResNet backbone and Waterfall module, is a robust and efficient architecture for pose estimation obtaining state-of-the-art results in single person pose detection for both single images and videos.

1. Introduction

Human pose estimation is an important task in computer vision with applications in activity recognition [54], human computer interaction [44], animation [4], gaming [42], health [10], and sports [51]. The importance of pose estimation has motivated the development of several approaches, in 2D [50], [49], [31], [46] and 3D [41], [59], [1]; on a single frame [5] or a video sequence [15]; for a single [52] or multiple subjects [8].

Pose estimation is challenging due to the large number of degrees of freedom in the human body mechanics and the frequent occurrence of parts occlusion. To overcome problems with occlusion, many methods rely on statistical and geometric models to estimate occluded joints [34], [32]. Another approach is the utilization of a library of known poses, known as anchor poses [41], but this could limit the generalization power of the model and the ability to learn for unforeseen poses.
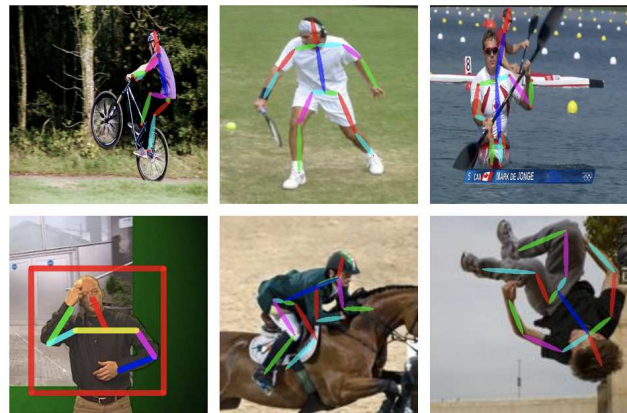

Figure 1. Pose estimation examples with our UniPose method.

Motivated by advances in semantic segmentation architectures [14], [56], [36], we propose a unified pose estimation framework, called UniPose, that consists of only one stage and obtains accurate results without postprocessing. A main component of our architecture is the Waterfall Atrous Spatial Pooling (WASP) module [3] which combines the cascaded approach for Atrous Convolution with the larger FOV obtained from parallel configuration from the Atrous Spatial Pyramid Pooling (ASPP) module [13].

Our unified approach predicts the location of joints using contextual information due to the larger Field-of-View (FOV) and 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 are shown in Figure 1. The main contributions of this paper are the following.

- We propose the UniPose framework, based on the Waterfall module for Atrous Spatial Pooling that achieves state-of-the-art results for single person human pose estimation.

- Our Waterfall module increases the receptive field of

the network by combining the benefits of cascade atrous convolutions with multiple FOV in a parallel architecture inspired by the spatial pyramid approach.

- The proposed UniPose method determines both the locations of joints and the bounding box for person detection, eliminating the need for separate branches in the network.

- We extend the Waterfall based approach to UniPose-LSTM by adopting a linear sequential LSTM configuration and obtain state-of-the-art results for temporal human pose estimation in video.

2. Related Work

Traditional methods for human pose estimation focused on the detection of joints, and consequently pose, via techniques that explored the geometry between joints in the target image [38], [55], and [48]. In recent years, methods relying on Convolutional Neural Networks (CNNs) achieved superior results [46], [8], [41]. The popular Convolutional Pose Machine (CPM) [52] took an approach that refined joint detection via a set of stages in the network. Stacked hourglass networks [31] utilized cascades of the hourglass structure for the pose estimation task. Building upon [52], Yan et al. integrated the concept of Part Affinity Fields (PAF), resulting in the OpenPose method [8]. PAF uses the detection of more significant joints to better estimate the prediction of less significant joints. This innovation allowed advances toward multi-person detection with decreased complexity and computational power.

The multi-context approach in [16] 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. Post processing with Conditional Random Fields (CRF) is used to assemble the relations between detected joints. However, the drawback of CRF is increased complexity that requires high computational power and results in a reduction in speed.

The High-Resolution Network (HRNet) [46] includes both high and low resolution representations. Starting with high resolution, the method gradually adds low resolution sub-networks to form more stages, and performs multi-scale fusion between sub-networks. 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 is used by Multi-Stage Pose Network (MSPN) [26], where the HRNet structure is combined with cross stage feature aggregation and coarse-to-fine supervision.

DeepPose [50] utilizes a cascade of deep CNNs and locates body joints via regression. The method relies on iterative refinement in order to better predict symmetric and lower confidence joints.

Some recent works attempt to leverage contextual information into pose estimation. The Cascade Prediction Fusion (CPF) [57] uses graphical components in order to exploit the context for pose estimation. Similarly, the Cascade Feature Aggregation (CFA) [45] aims to use semantic information to detect pose with a cascade approach. Generative Adversarial Networks (GANs) were used in [9] to learn dependencies and contextual information for pose.

The Location, Classification, and Regression network (LCR-Net) [41] extends pose estimation to 3D space via depth regression. LCR-Net relies on a Detectron backbone [19] 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 on the network for the estimation of unforeseen poses.

In a different approach for 3D pose estimation, the MonoCap method for human capture [59] couples a CNN with a geometric prior in order to statistically determine the third dimension for the pose using the Expectation-Maximization algorithm.

A drawback of some current methods is that they require an independent branch for the detection of the bounding box of human subjects in the frame. LightTrack [33], for instance, relies on a separate YOLO [40] architecture to perform the detection of subjects prior to detecting joints. In a different feamework, LCR-Net [41] has different branches for the detection using Detectron [19] and the arrangement of joints during classification.

2.1. 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 that takes into account the temporal component is to better estimate joints during blurring or occlusion using information from previous frames.

Targeting video applications, Modeep [23] utilized color channels from adjacent frames as input attempting to merge the motion in the video. Pfister et al. [37] also proposed a similar technique to detect gestures in a video sequence.

More recently, optical flow techniques were adopted to tackle the temporal component for pose estimation. Deep-flow [53] used optical flow to better connect predictions between frames in a more continuous detection. Another method that utilized optical flow is Thin-Slicing [43], relying on both optical flow and spatial-temporal model. However, the increased complexity of this model results in a significant increase in computational cost.
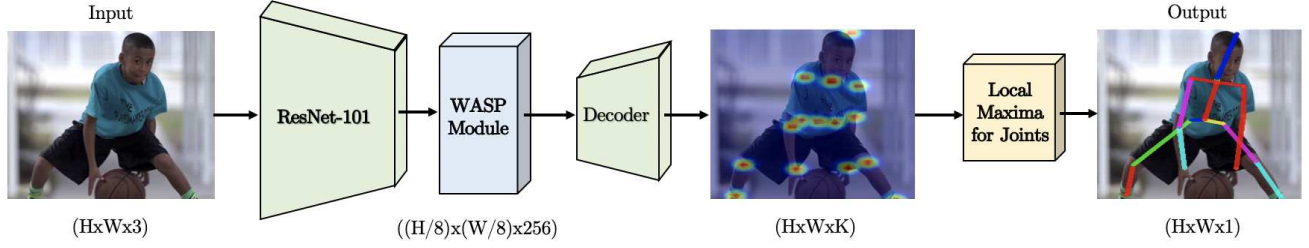
Figure 2. UniPose architecture for single frame pose detection. The input color image of dimensions (HxW) is fed through the ResNet backbone and WASP module to obtain 256 feature channels at reduced resolution by a factor of 8. The decoder module generates K heatmaps, one per joint, at the original resolution, and the locations of the joints are determined by a local max operation.

The Chained Model [20] utilizes recurrent networks to incorporate the temporal component. A similar concept was adopted by the LSTM Pose Machine [30] approach, where the LSTM was used to augment memory in the network.

Applications of LSTM aren't limited to the temporal component. Recurrent 3D Pose Sequence Machines (RSPM) [27] used LSTMs in the regression from 2D to 3D, to obtain better correspondence during the regression.

2.2. Pose Estimation for Sign Language

Despite the efforts on generic pose estimation methods, specific applications, such as for sign language, are currently lacking in research. Charles et al. [11] estimated pose during signing in long television broadcasting videos. The method relied on an initial separation from the background by the use of semantic segmentation, followed by a random forest regression to locate the upper limbs of the signer. The work in [6] used temporal tracking in order to detect parts and estimate upper body joints in similar frames.

DeepSign [18] applied transfer learning on a pretrained CNN for joint detection during sign language. Their approach followed the work done by [50] in generic pose images and incorporated application specific transfer learning in the final architecture.

2.3. Atrous Convolution and ASPP

An important challenge with both semantic segmentation and pose estimation methods incorporating CNN layers is the significant reduction of resolution caused by pooling. Fully Convolutional Networks (FCN) [29] [29] addressed the resolution reduction problem by deploying upsampling strategies across deconvolution layers. These attempt to reverse the convolution operation and increase the feature map size back to the dimensions of the original image.

A popular technique in semantic segmentation is the use of dilated or Atrous or dilated convolutions [13]. The main objectives of Atrous convolutions are to increase the size of the receptive fields in the network, avoid downsampling, and generate a multi-scale framework for processing.

In the simpler case of a one-dimensional convolution, the output of the signal is defined as follows:

$$y[i] = \sum_{l=1}^{L} x[i + rl] \cdot w[l] \qquad (1)$$

where r is the rate of dilation, $\omega[l]$ is the filter of length L, $x[i]$ is the input, and $y[i]$ is the output. A rate value of one results in a regular convolution operation.

Motivated by the success of the Spatial Pyramids applied on pooling operations [21], the ASPP architecture was successfully incorporated in DeepLab [13] for semantic segmentation. 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. We leverage this capability in a more efficient manner with our Waterfall architecture in the UniPose framework.

3. UniPose Architecture

We propose UniPose, a unified architecture for pose estimation, that exploits the large FOV generated by atrous convolutions combined with cascade of convolutions in a "Waterfall" configuration. Our WASP module offers multi-scale representations as well as efficiency in the reduced size of the network. Improving upon previous works, Uni-Pose does not require separate branches for bounding box and joint detections. Instead, it performs a unified detection of the bounding box for the human subject and its joints.

The UniPose processing pipeline is shown in Figure 2. The input image is initially fed into a deep CNN, in this case ResNet-101, with the final layers replaced by a WASP module. The resultant feature maps are processed by a decoder network that generated K heatmaps, one for each joint, with the corresponding probability distributions obtained from Softmax. Then the decoder performs bilinear
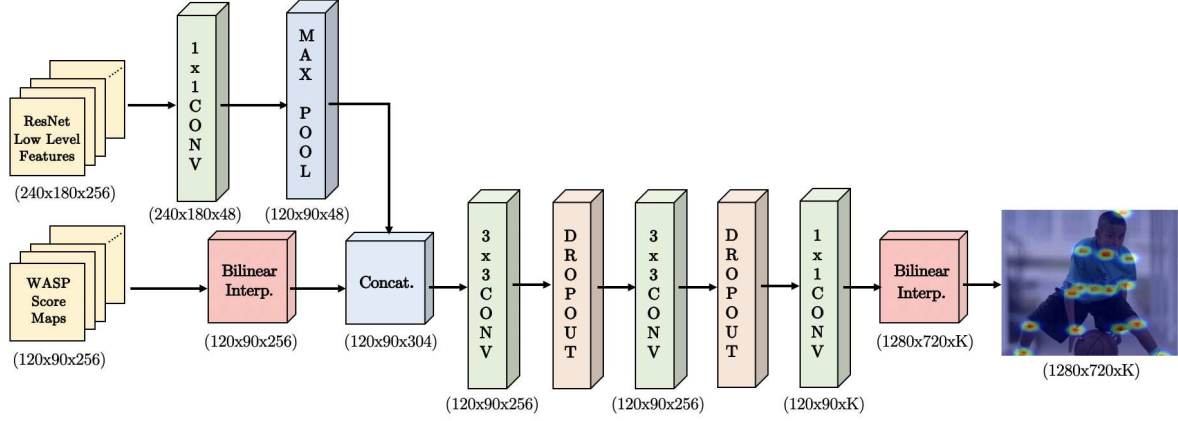
Figure 3. Decoder module used in the UniPose pipeline. The original image dimensions are (1280x720). The inputs to the decoder are 256 channels of ResNet low level features and 256 channels of the WASP feature maps. The output of the decoder is K heatmaps corresponding to K joints, shown in the image example. Additionally, the decoder outputs heatmaps for the bounding box (not shown in the image).

interpolation to recover the original resolution, followed by a local max operation to localize the joints for pose estimation. The decoder in our network generates detections of joints for both visible and occluded parts. Additionally, the decoder generates a bounding box detection without the use of post-processing or independent parallel branches.

We next provide the motivation for the development of the WASP module [3] and contrast it with traditional deconvolutions in [29] and the ASPP architecture in [13].
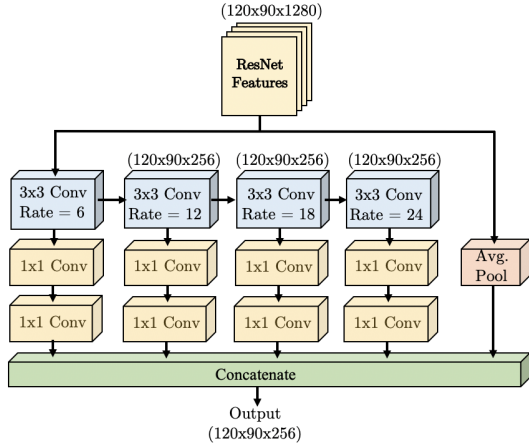

Figure 4. Waterfall architecture in the WASP module [3]. The original image dimensions are (1280x720). The inputs to the WASP module are 1280 channels of ResNet features maps.

3.1. WASP Module

The WASP module generates an efficient multi-scale representation that helps UniPose achieve state-of-the-art results. The WASP architecture, shown in Figure 4, is designed to leverage both the larger FOV of the ASPP configuration and the reduced size of the cascade approach. Ini-

tially applied by [3] for semantic segmentation, the inspiration for WASP was to combine the benefits of the ASPP [13], Cascade [14], and Res2Net [17] modules.

WASP relies on atrous convolutions, which are fundamental to ASPP, to maintain a large FOV. It also performs a cascade of atrous convolutions at increasing rates to gain efficiency, a concept motivated by the cascade approach. Furthermore, WASP incorporates multi-scale features inspired by the Res2Net architecture and other multi-scale approaches. In contrast to ASPP and Res2Net, 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. WASP 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.

WASP is designed with the goal of reducing the number of parameters in order to deal with memory constraints and overcome the main limitation of atrous convolutions. The four branches in WASP have different FOV and are arranged in a waterfall-like fashion. The atrous convolutions in WASP start with a small rate of 6, which consistently increases in subsequent branches (rates of 6,12,18,24). This configuration gains efficiency due to the smaller filter sizes, and creates multi-scale features with each branch that are combined to obtain a richer representation. The WASP module is utilized in the UniPose architecture of Figure 2 for pose estimation.

3.2. Decoder Module

Our decoder module converts the score maps resulting from the WASP module to heatmaps corresponding to body joints and the bounding box. Figure 3 shows the decoder architecture for an input color image of size (1280×720). The decoder receives 256 feature maps from WASP and 256 low
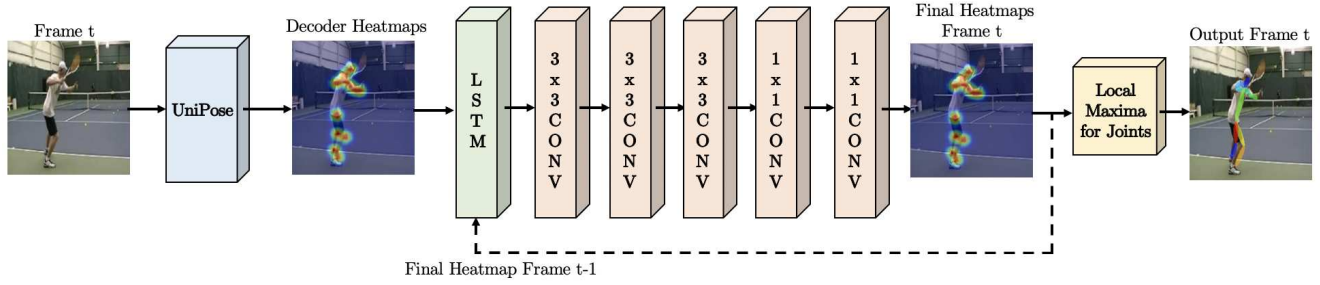
Figure 5. UniPose-LSTM architecture for pose estimation in videos. The joint heatmaps from the decoder of UniPose are fed into the LSTM along with the final heatmaps from the previous LSTM state. The convolutional layers following the LSTM reorganize the outputs into the final heatmaps used for joint localization.

level feature maps from the first block of the ResNet 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, and a final bilinear interpolation to resize to the original input size. The output consists of K heatmaps corresponding to K joints that are used for joint localization after a local max operation. Additionally, the decoder outputs heatmaps for the bounding box without requiring an additional branch.

3.3. UniPose-LSTM for Pose Estimation in Video

The UniPose architecture was modified to UniPose-LSTM for pose estimation in video. 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 5. 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.

4. Datasets

We performed experiments on four datasets. Two of the datasets are composed of single images: Leeds Sports Pose (LSP) [24] and MPII [2]; and two datasets are composed of video sequences: Penn Action [58] and BBC Pose [12]. A brief description of these datasets is provided below.

The Leeds Sports Pose (LSP) dataset [24] 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.

Penn Action [58] dataset contains 2,326 video sequences of 15 different activities including different 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.

The BBC Pose dataset [12] consists of 20 videos from the British Broadcasting Corporation (BBC) with the presence of a British Sign Language (BSL) signer. The BBC Pose dataset was utilized for the specialized application of human pose for sign language. The dataset includes of 610,115 labelled images for training, 309,171 for validation, and 309,260 for testing. As a limitation of the dataset, the labels consist of only 7 keypoints in the human upper body including head, shoulders, elbows, and wrists.

4.1. Data Pre-Processing

In order to train our network for joint detection, a preprocessing step was performed. Ideal Gaussian maps were

generated at the locations of joints in the ground truth labels. These maps are more effective for training than single points at the joint locations, and they are used to train our UniPose network to generate Gaussian heatmaps corresponding to the location of each joint in the frame.

Gaussians with different σ values were considered in the training of the network to evaluate their effectiveness. For the presented results and final analysis, 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

We performed training, validation and testing of UniPose based on the procedures and metrics described in this section. Compared to state of the art, our methods achieved superior performance in several datasets, for both single frame pose estimation with UniPose and video pose estimation with UniPose-LSTM, including the specific task of pose estimation on sign language videos.

5.1. Metrics

For the evaluation of UniPose, various datasets and metrics were used, depending on previously reported results and the available ground truth for each dataset. Some datasets, such as LSP [24], report and compare accuracy in Percentage of Correct Parts (PCP), where a limb is considered detected if the distance of its two predicted joints is below a threshold. In this paper, we adopted a threshold of half the distance of the ground truth limb, commonly referred as PCP@0.5. The PCP method introduces a bias due to the stronger penalization for the detection of smaller limbs (i.e. arm in comparison to torso), since they naturally have a shorter distance, and consequently a smaller threshold for detection.

Another 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.

5.2. Simulation Parameters

We input the native resolution of the input image without resizing, in order to train the network with the most detail possible through our dense and large FOV network. For that reason, the batch size utilized varied from high amounts for lower resolution datasets (e.g. LSP) to smaller batches of 4 for datasets such as the BBC Pose [12].

We experimented with different rates of dilation on the WASP module. We 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.

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 [28]. All experiments were performed using PyTorch 1.0 on Ubuntu 16.04. The workstation has an Intel i5-2650 2.20GHz CPU with 16GB of RAM and an NVIDIA Tesla V100 GPU.

6. Results

We initially tested our network on the LSP dataset and compared the results with other methods, as shown in Table 1. UniPose achieved a PCP of 72.8% and a PCK@0.2 of 94.5%, showing significant gains in comparison to other approaches in both metrics. Differently than methods such as CPM [52], UniPose is able to detect the body joints with high confidence in a single iteration, instead of going through several stages or iterations in the network.

Examples of pose estimation for subjects from LSP dataset are shown in Figure 6. 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 in limbs that are not sufficiently separated and occlude each other.

Method	PCP for LSP	PCK@0.2 for LSP
UniPose (ours)	**72.8%**	**94.5%**
Cao et al. [9]	-	94.3%
8-Stack HG [57]	-	94.0%
Part Regression [7]	-	90.7%
CPM [52]	-	90.5%
DeepCut [39]	-	87.1%
Recurrent [5]	-	85.2%
DeepPose [50]	61%	-
Poselet [38]	56%	-
Tian et al.[48]	56%	-

Table 1. Pose estimation results and comparison with other methods for the LSP dataset.

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 used the center map of the main person in order to detect the pose of the correct individual. We used "Detectron2" [19] for segmentation and detection of all the individuals, followed by the UniPose method to detect pose of the selected individual.

Table 2 shows the results for the MPII testing dataset. UniPose achieves a PCKh detection rate of 92.7% and outperformed other methods for single person pose estimation.

Examples of pose estimation with UniPose in the MPII dataset are shown in Figure 7. These examples illustrate
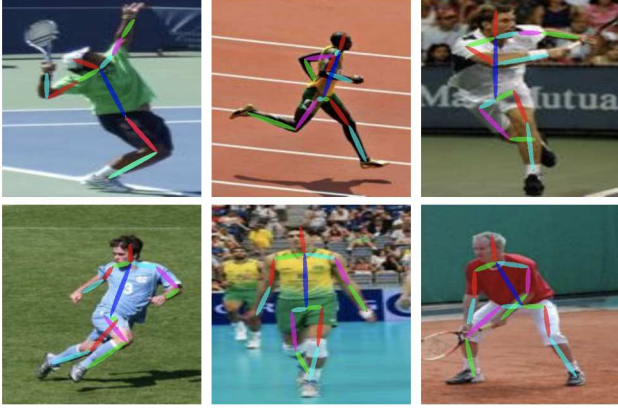
Figure 6. Pose estimation examples from the LSP dataset

that UniPose deals effectively with occlusion, e.g. in the case of the horse rider.

Method	PCKh@0.5 for MPII
UniPose (ours)	**92.7%**
MSPN [26]	92.6%
8-Stack HG [57]	92.5%
Deeply-Learned Models [47]	92.3%
Cao et al. [9]	92.3%
Structure-Aware [25]	92.0%
Improvement Multi-Stage [45]	90.1%
CFA [45]	90.0%
CPM [52]	88.5%

Table 2. Pose estimation results and comparisons with other methods for the MPII dataset.
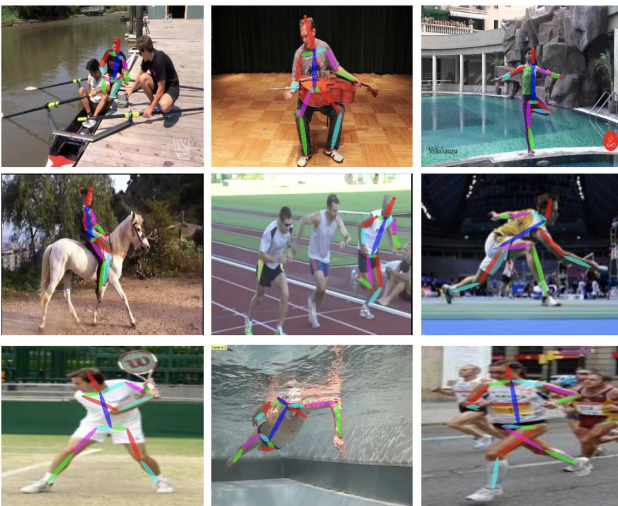

Figure 7. Pose estimation examples from the MPII dataset

Table 3 shows the results for our UniPose-LSTM in the Penn Action dataset [58]. 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 [55] and applied by [30]. Our method results in a 99.3% detection rate, an improvement of 1.6% over the next best result.

Method	PCK for Penn Action
UniPose-LSTM (ours)	**99.3%**
LSTM-PM [30]	97.7%
CPM [52]	97.1%
Thin-Slicing [43]	96.5%
N-best [35]	91.8%
Iqbal [22]	81.1%

Table 3. Pose estimation results 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. This feature enables a higher detection rate and consequently a more robust architecture against motion blur and occlusions in the image.

We experimented with different numbers of frames to evaluate the memory capability associated with the use of the LSTM. Table 4 shows the accuracy gains observed from implementing LSTM for a number of frames ranging from 1 to 6. It is noticeable that the accuracy gains obtained by the LSTM plateaus as the number of frames reaches values of 5 or larger.

Examples of detections for the Penn Action dataset [58] are shown in Figure 8. The examples selected are of fast motion scenarios, showing every other frame in sequence, so that significant differences are observed between frames.

Number of frames in LSTM	PCK for Penn Action
1	98.4%
2	98.6%
3	98.8%
4	99.1%
5	99.3%
6	99.3%

Table 4. UniPose-LSTM results for the Penn Action dataset for different number of frames used by the LSTM.

Table 5 shows results for the BBC Pose dataset, where pose is detected specifically for sign language. UniPose-
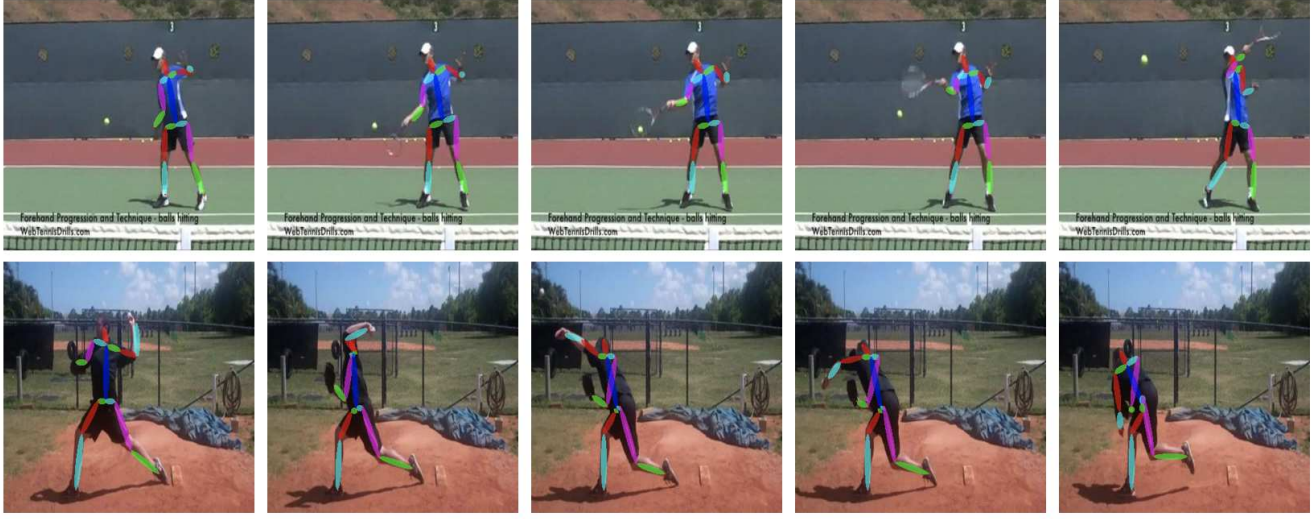
Figure 8. Pose estimation examples from the Penn Action dataset for a sequence of frames.

LSTM significantly outperforms the older methods by achieving a PCKh of 98.9%. In order to obtain results from another method for comparison, we trained CPM for the BBC Pose dataset, obtaining a PCK of 97.6%, which is below the performance of UniPose-LSTM.

Method	PCKh@0.5 for BBC Pose
UniPose-LSTM (ours)	**98.9%**
CPM [52]	97.6%
Charles et al. [11]	74.9%
Buehler et al. [6]	67.5%

Table 5. Pose estimation results and comparisons with other methods for the BBC Pose dataset

Figure 9 shows examples of pose estimation and bounding box detections for subjects in the BBC dataset. Detections are shown for every other frame to illustrate different poses in the sequence. Our network is able to efficiently detect the pose of the signers as well as generate the bounding box containing their signing area.

7. Conclusion

We presented the UniPose framework for pose estimation in single images and videos, respectively. The UniPose pipeline utilizes the WASP module that features a waterfall flow with a cascade of atrous convolutions and multi-scale representations. The large FOV of WASP obtains a better interpretation of the contextual information in the frame, and contributes to more accurate pose estimation.

The results of UniPose and UniPose-LSTM demonstrated state-of-the-art performance on several datasets, i.e.,
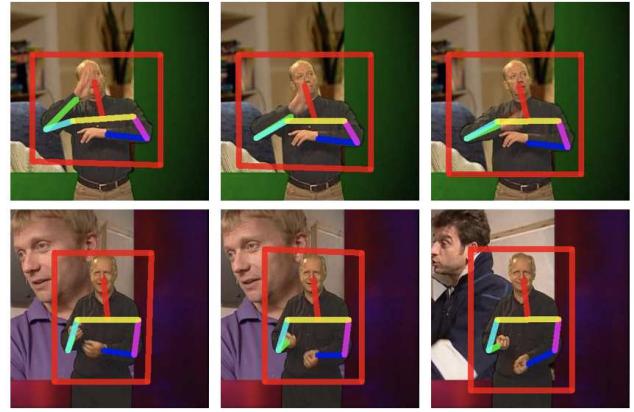

Figure 9. Pose estimation examples from BBC Pose dataset for a sequence of frames.

LSP, MPII, Penn Action and BBC Pose, using various metrics. Our framework shows promise for further use in a broader range of applications, including multiple person pose detection and 3D pose estimation.

8. Acknowledgements

References

[1] Riza Alp Guler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7297–7306, 2018. 1

[2] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014. 5, 6

[3] Bruno Artacho and Andreas Savakis. Waterfall atrous spatial pooling architecture for efficient semantic segmentation. *Sensors*, 19(24):5361, 2019. 1, 4

[4] Angelos Barmpoutis. Tensor body: Real-time reconstruction of the human body and avatar synthesis from RGB-D. *IEEE transactions on cybernetics*, 43(5):1347–1356, 2013. 1

[5] Vasileios Belagiannis and Andrew Zisserman. Recurrent human pose estimation. In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pages 468–475. IEEE, 2017. 1, 6

[6] Patrick Buehler, Mark Everingham, Daniel P Huttenlocher, and Andrew Zisserman. Upper body detection and tracking in extended signing sequences. *International Journal of Computer Vision*, 95(2):180, 2011. 3, 8

[7] Adrian Bulat and Georgios Tzimiropoulos. Human pose estimation via convolutional part heatmap regression. In *European Conference on Computer Vision*, pages 717–732. Springer, 2016. 6

[8] Z. Cao, T. Simon, S. E. Wei, and Y. Sheikh. Realtime multi-person 2D pose estimation using part affinity fields. *IEEE Computer Vision and Pattern Recognition*, 2017. 1, 2

[9] Zhongzheng Cao, Rui Wang, Xiangyang Wang, Zhi Liu, and Xiaoqiang Zhu. Improving human pose estimation with self-attention generative adversarial networks. In *2019 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 567–572. IEEE, 2019. 2, 6, 7

[10] Yao-Jen Chang, Shu-Fang Chen, and Jun-Da Huang. A kinect-based system for physical rehabilitation: A pilot study for young adults with motor disabilities. *Research in developmental disabilities*, 32(6):2566–2570, 2011. 1

[11] James Charles, Tomas Pfister, Mark Everingham, and Andrew Zisserman. Automatic and efficient human pose estimation for sign language videos. *International Journal of Computer Vision*, 110(1):70–90, 2014. 3, 8

[12] J. Charles, T. Pfister, D. Magee, D. Hogg, and A. Zisserman. Personalizing human video pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 5, 6

[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. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–845, 2018. 1, 3, 4

[14] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *CoRR*, abs/1706.05587, 2017. 1, 4

[15] Yu Cheng, Bo Yang, Bo Wang, Wending Yan, and Robby T. Tan. Occlusion-aware networks for 3d human pose estimation in video. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 1

[16] Xiao Chu, Wei Yang, Wanli Ouyang, Cheng Ma, Alan L Yuille, and Xiaogang Wang. Multi-context attention for human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1831–1840, 2017. 2

[17] Shang-Hua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip Torr. Res2net: A new multi-scale backbone architecture. *CoRR*, abs/1904.01169, 2019. 4

[18] Srujana Gattupalli, Amir Ghaderi, and Vassilis Athitsos. Evaluation of deep learning based pose estimation for sign language recognition. In *Proceedings of the 9th ACM International Conference on PErvasive Technologies Related to Assistive Environments*, page 12. ACM, 2016. 3

[19] Ross Girshick, Ilija Radosavovic, Georgia Gkioxari, Piotr Dollár, and Kaiming He. Detectron, 2018. 2, 6

[20] G. Gkioxari, A. Toshev, and N. Jaitly. Chained predictions using convolutional neural networks. In *arXiv preprint arXiv:1605.02346*, 2016. 3

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *CoRR*, abs/1406.4729, 2014. 3

[22] Umar Iqbal, Martin Garbade, and Juergen Gall. Pose for action - action for pose. *IEEE Conference on Automatic Face and Gesture Recognition (FG'17)*, 2017. 7

[23] Arjun Jain, Jonathan Tompson, Yann LeCun, and Christoph Bregler. Modeep: A deep learning framework using motion features for human pose estimation. In *Asian conference on computer vision*, pages 302–315. Springer, 2014. 2

[24] Sam Johnson and Mark Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *Proceedings of the British Machine Vision Conference*, 2010. doi:10.5244/C.24.12. 5, 6

[25] Lipeng Ke, Ming-Ching Chang, Honggang Qi, and Siwei Lyu. Multi-scale structure-aware network for human pose estimation. In *The European Conference on Computer Vision (ECCV)*, September 2018. 7

[26] Wenbo Li, Zhicheng Wang, Binyi Yin, Qixiang Peng, Yuming Du, Tianzi Xiao, Gang Yu, Hongtao Lu, Yichen Wei, and Jian Sun. Rethinking on multi-stage networks for human pose estimation. *arXiv preprint arXiv:1901.00148*, 2019. 2, 7

[27] Mude Lin, Liang Lin, Xiaodan Liang, Keze Wang, and Hui Cheng. Recurrent 3d pose sequence machines. *CoRR*, abs/1707.09695, 2017. 3

[28] Wei Liu, Andrew Rabinovich, and Alexander C. Berg. Parsenet: Looking wider to see better. *CoRR*, abs/1506.04579, 2015. 6

[29] J. Long, E. Shelhamer, and T. Darrel. Fully convolutional networks for semantic segmentation. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 3, 4

[30] Yue Luo, Jimmy S. J. Ren, Zhouxia Wang, Wenxiu Sun, Jinshan Pan, Jianbo Liu, Jiahao Pang, and Liang Lin. LSTM pose machines. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, abs/1712.06316, 2018. 3, 7

[31] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision*, pages 483–499. Springer, 2016. 1, 2

[32] Aiden Nibali, Zhen He, Stuart Morgan, and Luke Prendergast. 3d human pose estimation with 2d marginal heatmaps. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1477–1485. IEEE, 2019. 1

[33] Guanghan Ning and Heng Huang. Lighttrack: A generic framework for online top-down human pose tracking. *arXiv preprint arXiv:1905.02822*, 2019. 2

[34] George Papandreou, Tyler Zhu, Liang-Chieh Chen, Spyros Gidaris, Jonathan Tompson, and Kevin Murphy. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In *The European Conference on Computer Vision (ECCV)*, September 2018. 1

[35] Dennis Park and Deva Ramanan. N-best maximal decoders for part models. *ICCV*, 2011. 7

[36] Adam Paszke, Abhishek Chaurasia, Sangpil Kim, and Eugenio Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. *CoRR*, abs/1606.02147, 2016. 1

[37] Tomas Pfister, Karen Simonyan, James Charles, and Andrew Zisserman. Deep convolutional neural networks for efficient pose estimation in gesture videos. In *Asian Conference on Computer Vision*, pages 538–552. Springer, 2014. 2

[38] Leonid Pishchulin, Mykhaylo Andriluka, Peter Gehler, and Bernt Schiele. Poselet conditioned pictorial structures. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 588–595, 2013. 2, 6

[39] Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter V Gehler, and Bernt Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4929–4937, 2016. 6

[40] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv*, 2018. 2

[41] Gregory Rogez, Philippe Weinzaepfel, and Cordelia Schmid. LCR-Net: Localization-classification-regression for human pose. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, United States, 2017. 1, 2

[42] J. Shotton, R. Girshick, A. Fitzgibbon, T. Sharp, M. Cook, M. Finocchio, R. Moore, P. Kohli, A. Criminisi, A. Kipman, and A. Blake. Efficient human pose estimation from single depth images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012. 1

[43] Jie Song, Limin Wang, Luc Van Gool, and Otmar Hilliges. Thin-slicing network: A deep structured model for pose estimation in videos. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2, 7

[44] Yale Song, David Demirdjian, and Randall Davis. Continuous body and hand gesture recognition for natural human-computer interaction. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 2(1):5, 2012. 1

[45] Zhihui Su, Ming Ye, Guohui Zhang, Lei Dai, and Jianda Sheng. Cascade feature aggregation for human pose estimation. *arXiv preprint arXiv:1902.07837*, 2019. 2, 7

[46] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2

[47] Wei Tang, Pei Yu, and Ying Wu. Deeply learned compositional models for human pose estimation. In *The European Conference on Computer Vision (ECCV)*, September 2018. 7

[48] Yuandong Tian, C Lawrence Zitnick, and Srinivasa G Narasimhan. Exploring the spatial hierarchy of mixture models for human pose estimation. In *European Conference on Computer Vision*, pages 256–269. Springer, 2012. 2, 6

[49] Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann LeCun, and Christoph Bregler. Efficient object localization using convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 648–656, 2015. 1

[50] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1653–1660, 2014. 1, 2, 3, 6

[51] Luis Unzueta, Jon Goenetxea, Mikel Rodriguez, and Maria Teresa Linaza. dependent 3d human body posing for sports legacy recovery from images and video. In *2014 22nd European Signal Processing Conference (EUSIPCO)*, pages 361–365. IEEE, 2014. 1

[52] Shih-En Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 1, 2, 6, 7, 8

[53] Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, and Cordelia Schmid. Deepflow: Large displacement optical flow with deep matching. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1385–1392, 2013. 2

[54] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 1

[55] Yi Yang and Deva Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2878–2890, 2012. 2, 7

[56] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *ICLR*, 2016. 1

[57] Hong Zhang, Hao Ouyang, Shu Liu, Xiaojuan Qi, Xiaoyong Shen, Ruigang Yang, and Jiaya Jia. Human pose estimation with spatial contextual information. *arXiv preprint arXiv:1901.01760*, 2019. 2, 6, 7

[58] Weiyu Zhang, Menglong Zhu, and Konstantinos G Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. pages 2248–2255, 2013. 5, 7

[59] Xiaowei Zhou, Menglong Zhu, Georgios Pavlakos, Spyridon Leonardos, Konstantinos G. Derpanis, and Kostas Daniilidis. Monocap: Monocular human motion capture using a CNN coupled with a geometric prior. *CoRR*, abs/1701.02354, 2017. 1, 2