Stacked Hourglass Networks for Human Pose Estimation

Alejandro Newell, Kaiyu Yang, and Jia Deng

University of Michigan, Ann Arbor

Abstract. This work introduces a novel Convolutional Network architecture for the task of human pose estimation. Features are processed across all scales and consolidated to best capture the various spatial relationships associated with the body. We show how repeated bottom-up, top-down processing used in conjunction with intermediate supervision is critical to improving the performance of the network. We refer to the architecture as a 'stacked hourglass' network based on the successive steps of pooling and upsampling that are done to produce a final set of estimates. State-of-the-art results are achieved on the FLIC and MPII benchmarks outcompeting all recent methods.

Keywords: Human Pose Estimation

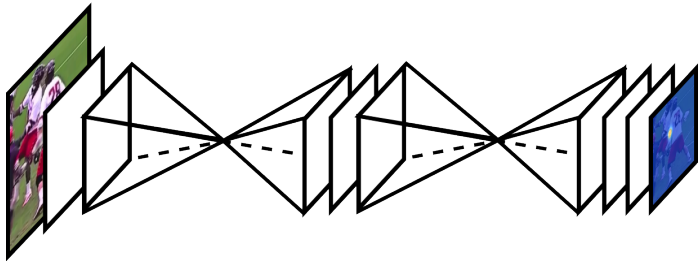


Fig. 1: Our network for pose estimation consists of two stacked hourglass models which allow repeated bottom-up, top-down inference.

1 Introduction

A key step toward understanding people in images and video is accurate pose estimation. Given a single RGB image we determine the precise pixel location of important keypoints of the body. From the top of the head down to the ankles, we do this localization of joints to achieve an understanding of a person's posture and limb articulation. This is useful for higher level tasks like action recognition, and it serves as a fundamental tool in diverse fields from human-computer interaction to animation.

As a well established problem in vision, pose estimation has plagued researchers with a variety of formidable challenges over the years. A good pose

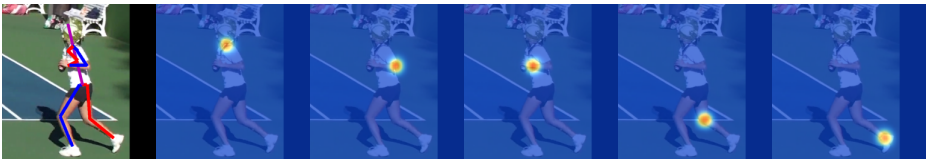


Fig. 2: Example output produced by our network. On the left we see the final pose estimate provided by the max activations across each heatmap. On the right we show sample heatmaps. (from left to right: neck, left elbow, left wrist, right knee, right ankle)

estimation system must handle a huge input space which entails robustness to occlusion and severe deformation, success on rare and novel poses, and invariance to changes in clothing, lighting, and viewing angle. Early work tackles such difficulties using robust image features and sophisticated structured prediction [1,2,3,4,5,6,7,8,9]: the former is used to produce local interpretations, whereas the latter is used to infer a globally consistent pose.

This conventional pipeline, however, has been greatly reshaped by Convolutional Neural Networks (ConvNets) [10,11,12,13,14], a main driver behind an explosive rise in performance across many computer vision tasks. Recent pose estimation systems [15,16,17,18,19,20] universally adopted ConvNets as the main building block, largely replacing hand-crafted features and graphical models; this strategy has yielded drastic improvements on standard benchmarks [1,21,22].

In this work we continue in the same direction and introduce a novel 'stacked hourglass' network design for predicting human pose. The network captures and consolidates information across all scales of the image. We refer to the design as an 'hourglass' given how we visualize the gradual pooling and subsequent upsampling used to get the final output resolution. Like many convolutional approaches that produce pixel-wise outputs, the hourglass network pools down to a very low resolution, then upsamples and combines features across multiple resolutions [15,23]; on the other hand, the hourglass differs from prior designs primarily in its more symmetric topology.

We expand on a single hourglass by placing two modules together end-to-end allowing for repeated bottom-up, top-down inference across scales. With the additional use of intermediate supervision, this repeated bidirectional inference is critical to the network's performance. The final network architecture achieves a significant improvement on the state-of-the-art for two standard pose estimation benchmarks (FLIC [1] and MPII Human Pose [21]). For example, on FLIC the error rate of the elbow is reduced from 4.7% to 1.8%. For MPII, we see a 2-3% improvement across difficult joints up from the most recently available results. [1]

2 Related Work

With the introduction of 'DeepPose' by Toshev et al. [24], research on human pose estimation started to shift from the classic approaches [2,3,4,1,5,6,7,8,9] to

[1] Code is available at `http://www-personal.umich.edu/~alnewell/pose`

deep networks. Toshev et al. use their network to directly regress the x,y coordinates of joints. The work by Tompson et al. [15] instead generates heatmaps by running an image through multiple resolution banks in parallel to simultaneously capture features at a variety of scales. Our network design largely builds off of their work, relying on and adapting their method for combining features across different resolutions.

But in contrast to our approach, a critical feature of the method proposed by Tompson et al. [15] is the joint use of a ConvNet and a graphical model. Their graphical model learns typical spatial relationships between joints. Others have recently tackled this in similar ways [20,25,17] with variations on how to approach unary score generation and pairwise comparison of adjacent joints. Chen et al. [25] cluster detections into typical orientations so that when their classifier makes predictions additional information is available indicating the likely location of a neighboring joint. Joint use of a deep ConvNet and graphical model largely defines the pose estimation literature of the past year, whereas we achieve superior performance without the use of a graphical model or any explicit enforcement of the structure of the human body.

In addition to the use of graphical models, there are several examples of iterative or multi-stage training methods. Carreira et al. [19] use what they refer to as Iterative Error Feedback. Each successive run through their network takes as input the image along wih predictions from the previous forward pass and further refines them. Wei et al. [18] build on the work of multi-stage pose machines [26] but now with the use of ConvNets for feature extraction. Given our use of intermediate supervision, our work is similar in spirit to these methods, but our building block (the hourglass model) is different.

Tompson et al. build on their work in [15] with a cascade to refine predictions. This serves to increase efficency and reduce memory usage of their method while improving localization performance in the high precision range [16]. We find that we get sufficient precision in our predictions without a cascade. One consideration is that for many of our failure cases a refinement of position within a local window would not offer much in the way of improvement since often our error cases include either an occluded or misattributed limb. For both situations, any further evaluation at a local scale will not improve the prediction.

There are variations to the pose estimation problem which include use of additional features such as depth or motion cues [27,28]. Also, there is the more challenging task of gracefully dealing with simultaneous annotation of multiple people [29,17]. We focus solely on the task of single person pose estimation from an RGB image.

Our work is closely connected to fully convolutional networks [23] and other designs that process spatial information in multiple scales for dense prediction [15,30,31,32,33,34,35,36,37,38]. Xie et al. [30] give a summary of typical architectures. Our hourglass module before stacking differs from these designs mainly in its more symmetric distribution of model capacity between bottom-up processing (from high resolutions to low resolutions) and top-down processing (from low resolutions to high resolutions). For example, fully convolutional net-

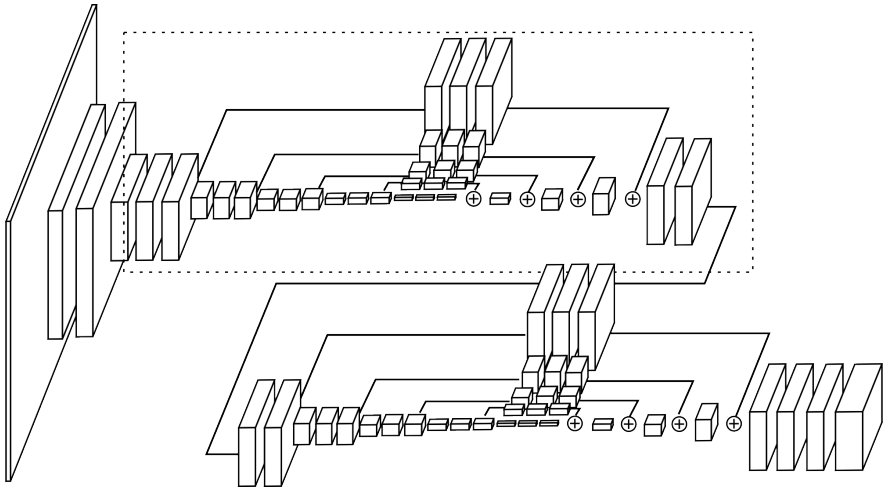


Fig. 3: A detailed illustration of the stacked hourglass design. The dashed lines surround one "hourglass" module. The layers are identical across each module. Each box in the figure corresponds to a residual module as seen in Figure 4. Just before upsampling the number of features in the residual modules are doubled. The layers between hourglass sections as well as the last three layers consists of 1x1 convolutions.

works [23] and holistically-nested architectures [30] are both heavy in bottom-up processing but light in their top-down processing, which consists only of a (weighted) merging of predictions across multiple scales. Another major difference of our work from these prior works is that they perform a single pass of bottom-up, top-down inference, whereas we perform repeated bottom-up, top-down inference by stacking two hourglasses.

Our hourglass module before stacking is also related to the convolution-deconvolution architecture [39,40,41], which deploys a DeconvNet [42] to do pixel-wise prediction. Noh et al. [39] used the conv-deconv architecture to do semantic segmentation, Rematas et al. [41] used it to predict reflectance map of objects. Zhao et al. [40] developed a unified framework for supervised, unsupervised and semi-supervised learning by adding a reconstruction loss. The symmetric topology of these networks is similar, but the nature of the operations is quite different in that we do not use unpooling or deconv layers. Instead, we rely on simple nearest neighbor upsampling and skip connections for top-down processing.

3 Network Architecture

3.1 Hourglass Design

The hourglass design is motivated by the necessity to capture information at every scale. While local evidence is essential for identifying features like faces and

hands, a final pose estimate requires a coherent understanding of the full image. The orientation of the body, the arrangement of limbs, and the relationships of adjacent joints are among the many cues that are best recognized at different scales of the image.

To identify and utilize these cues the network must have some mechanism to effectively process and consolidate features across scales. Some approaches tackle this with the use of separate pipelines that process the image independently at multiple resolutions and combine features later on in the network [15,18]. Instead, we choose to use a single pipeline that branches off at each resolution. The network reaches its lowest resolution at 4 pixels by 4 pixels allowing small filters to be applied that compare features across the entire image space.

The branching at each layer allows the preservation of spatial accuracy at each scale in addition to the further processing of spatial features that are best captured at a particular resolution. The information available at all of these resolutions is then combined so that the network can produce its predictions. To bring together information across two different resolutions, we use the approach of Tompson et al. [15] and do nearest neighbor upsampling of the lower resolution and an elementwise addition of the features.

After reaching the output resolution of the network, three consecutive rounds of 1x1 convolutions are applied to produce the final network predictions. The final output of the network is a set of heatmaps where the network predicts the probability of each joint's presence at every pixel. The pipeline of the hourglass model is illustrated in Figure 3. (The hourglass is doubled due to the stacked approach to be explained further down) The end result is a pipeline that fully processes and integrates high-order local and global features.

3.2 Layer Implementation

While maintaining the overall hourglass shape, there is still some flexibility in the specific implementation of layers. Different choices can have a dramatic impact on the final performance and training of the network. We explore several options for layer design in our network. Recent work has shown the value of reduction steps with 1x1 convolutions, as well as the benefits of using consecutive smaller filters to capture a larger spatial context. For example, one can replace a 5x5 filter with two separate 3x3 filters. We tested our overall network design, swapping in different layer modules based off of these insights. We experienced a massive increase in network performance after switching from standard convolutional layers with large filters and no reduction steps to newer methods like the residual learning modules presented by He et al. [14] and "Inception"-based designs [12]. After the initial performance improvement with these types of designs, various additional explorations and modifications to the layers did little to further boost performance or training time.

Residual learning modules serve us well. Filters greater than 3x3 are never used, and the bottlenecking restricts the total number of parameters at each layer curtailing total memory usage. The module used in our network is shown in Figure 4. To put this into the context of the full network design, in Figure 3

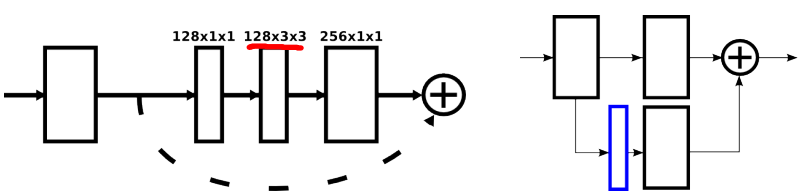


Fig. 4: **Left:** Residual Module [14] that we use throughout our network. **Right:** Illustration of intermediate supervision. The network splits and produces a set of heatmaps (outlined in blue) where a loss can be applied. A 1x1 convolution remaps the heatmaps to match the number of channels of the intermediate features. These are added togethered before continuing forward.

each box except for the first represents a single residual module. The first layer is a standard 7x7 convolution with stride 2, and anywhere else that the resolution drops implies max pooling with a 2x2 window and stride 2. All residual modules in the hourglass output 256 features except for layers right before upsampling where we increase the number of output features to 512.

3.3 Stacked Hourglass with Intermediate Supervision

To allow better processing across scales, we stack two hourglass modules end-to-end. This provides the network with a mechanism for repeated bottom-up, top-down inference allowing for reevaluation of initial estimates and features across the whole image. The key to this approach is the prediction of an initial set of heatmaps upon which we can apply a loss. These first predictions are generated after passing through one hourglass where the network has now had the opportunity to process features at both local and global contexts. Then, the second hourglass module allows these high level features to be processed again across all scales to further capture and understand higher order spatial relationships. This is similar to other pose estimations methods that have demonstrated strong performance with multiple iterative stages and intermediate supervision [19,18].

Consider the limits of applying intermediate supervision with only the use of a single hourglass module. What would be an appropriate place in the pipeline to generate an initial set of predictions? Most higher order features are present only at lower resolutions except at the very end when upsampling occurs. If supervision is provided after the network does upsampling then there is no way for these features to be reevaluated relative to each other on a large scale. If we want the network to best refine predictions they cannot be evaluated exclusively locally, the context and understanding of the full image and relationship to other predictions is absolutely crucial. We could apply supervision earlier in the pipeline before pooling, but at this point the features at a given pixel are the result of a relatively local receptive field thus ignoring critical global cues.

Repeated bottom-up, top-down inference with stacked hourglasses does not have the same concerns. Local and global cues are integrated after the first

hourglass module, and a loss applied here explicitly requires a high-level understanding while only halfway through the pipeline. Then, a subsequent stage of bottom-up, top-down processing allows for a deeper reconsideration of these features across the entire image.

This approach for going back and forth between scales is particularly important because preserving the spatial location of features is essential to do the final localization step. The precise position of a joint is an indispensable cue for other decisions being made by the network. With a structured problem like pose estimation, the output is an interplay of many different features that should come together to form a coherent understanding of the scene. Contradicting evidence and anatomic impossiblity are big giveaways that somewhere along the line a mistake was made, and by going back and forth the network can maintain precise local information while considering and then reconsidering the overall coherence of the features.

We reintegrate the initial predictions back into the feature space by mapping them up to a larger number filters with an additional 1x1 convolution. Next, we do an elementwise addition with the intermediate features of the pipeline as visualized in Figure 4. The output here serves directly as the input for the second hourglass module which generates a new, final set of predictions. We apply the loss to both sets of predictions separately using the same ground truth. The details for the loss and ground truth are described below.

3.4 Training Details

We evaluate our network on two benchmark datasets, FLIC [1] and MPII Human Pose [21]. FLIC is composed of 5003 images (3987 training, 1016 testing) taken from films. The images are annotated on the upper body with almost all figures facing the camera straight on. MPII Human Pose consists of around 25k images with annotations for multiple people providing 40k annotated samples (28k training, 11k testing). The test annotations are not provided so in all of our experiments we train on a subset of training images while evaluating on a held-out validation set of around 3000 samples. MPII consists of images taken from a wide range of human activities with a challenging array of widely articulated full-body poses.

Without a graphical model or other postprocessing step the input image to the network must convey all necessary information to determine the target person to generate an annotation for. We accomplish this by centering the person in the input image. This is done with FLIC by centering along the x-axis according to the torsobox annotation, but we do no vertical adjustment or scale normalization. For MPII, it is standard to utilize the scale and center annotations provided with all images. For each sample, we crop around the target person and resize the image to get a 256x256 input for the network. Occasionally people are in such close proximity it remains unclear after centering who the network should provide annotations for; regardless we provide these situations as training samples and will later explore how the network handles the ambiguity. We do data augmentation that includes flipping, rotation (+/- 30 degrees), and scaling. We

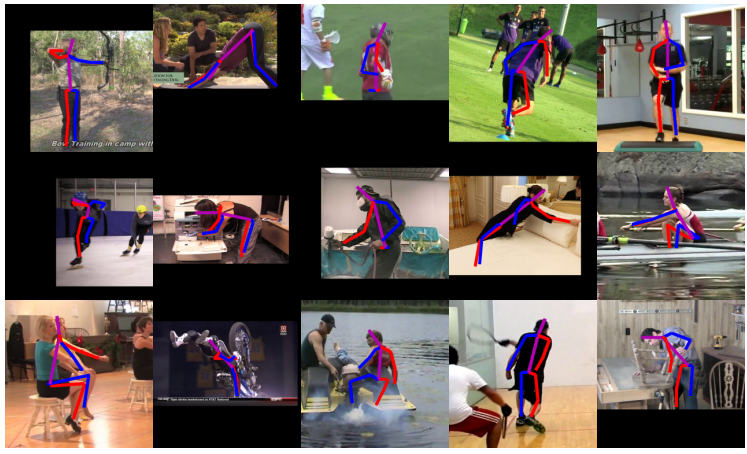


Fig. 5: Example output on MPII's test set. The images are cropped in the same way as they would be when given to the network, which demonstrates how centering and scale normalization is done. The top two rows are randomly sampled; the bottom row provides examples of typical error cases.

avoid translation augmentation of the image since location is the critical cue communicating to the network who should be annotated.

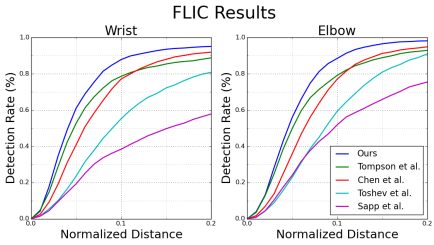
The network is trained using Torch7 [43] and for optimization we use rmsprop [44] with a learning rate of 2.5e-4. Training takes about 5 days on a 12 GB NVIDIA TitanX GPU. We drop the learning rate once by a factor of 5 after validation accuracy plateaus. Batch normalization [13] was also used to improve training. A single forward pass of the network takes 130ms. For generating our final test predictions we run both the original input and a flipped version of the image averaging the heatmaps together (making sure the appropriate left and right versions of the joints correspond to each other). The final prediction of the network is the max activating location on the heatmap.

For supervision we use the same technique as Tompson et al. [15]. A Mean-Squared Error (MSE) loss is applied comparing the predicted heatmap to a ground-truth heatmap with resolution 64x64 consisting of a 2D gaussian (with standard deviation of 1.5 px) centered on the joint location. We do not make use of a cascade or further adjustments to improve the precision of our heatmap prediction. In MPII Human Pose, some joints do not have a corresponding ground truth annotation. In these cases the joint is usually severely occluded, so for supervision we use a ground truth heatmap of all zeros.

4 Results

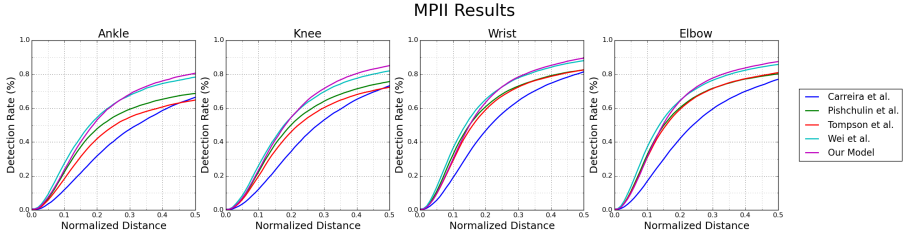
4.1 Evaluation

Evaluation is done using the standard Percentage of Correct Keypoints (PCK) metric which reports the percentage of detections that fall within a normal-



	Elbow	Wrist
Tompson et al.[16]	93.1	89.0
Chen et al.[25]	95.3	92.4
Toshev et al.[24]	92.3	82.0
Sapp et al.[1]	76.5	59.1
Ours	**98.2**	**95.2**

Fig. 6: Pose estimation results on FLIC (PCK@0.2)



	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	Total
Tompson et al., NIPS'14	95.8	90.3	80.5	74.3	77.6	69.7	62.8	79.6
Carreira et al., arXiV'15	95.7	91.7	81.7	72.4	82.8	73.2	66.4	81.3
Tompson et al., CVPR'15	96.1	91.9	83.9	77.8	80.9	72.3	64.8	82.0
Pishchulin et al., arXiV'15	94.1	90.2	83.4	77.3	82.6	75.7	68.6	82.4
Wei et al., arXiV'16	**97.7**	94.5	88.3	83.4	87.9	81.9	78.3	87.9
Our model	97.6	**95.4**	**90.0**	**85.2**	**88.7**	**85.0**	**80.6**	**89.4**

Fig. 7: Pose estimation results on MPII Human Pose (PCKh@0.5)

ized distance of the ground truth. For MPII, PCKh refers to the fact that the normalization is done by a fraction of the head size.

FLIC: Our results on FLIC are very competitive reaching almost perfect performance on the shoulder and elbow (99.4% and 98.2% respectively at a normalized distance threshold of .2), and 95.2% performance on the wrist. For FLIC we remove the second half of the network, running on only one hourglass module and applying no intermediate supervision. It is important to note that these results are observer-centric, which is consistent with how others have evaluated their output on FLIC.

MPII: We achieve state-of-the-art results across the board on the MPII Human Pose dataset. All numbers can be seen in Figure 7. On difficult joints like the wrists, elbows, ankles, and knees we improve upon the most recent state-of-the-art results by a margin of 2-3% reducing the average error rate on these joints from 17% to 14.8%. For the elbow we reach a final accuracy of 90% and for the wrist an accuracy of 85.2%. To see some of the network's predictions on MPII test images please refer to Figure 5.

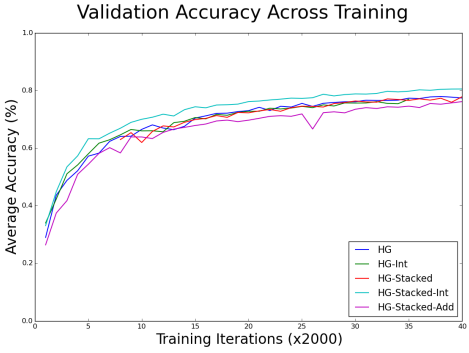


Fig. 8: Comparison of validation accuracy as training progresses. The accuracy is the averaged across the wrists, elbows, knees, and ankles. Our different design choice include: *Int* - Intermediate supervision, *Stacked* - Stacked hourglass design, *Add* - A single loss applied on the addition of the intermediate and final outputs together

4.2 Ablation Experiments

There are two main design choices to explore in this work: the effect of stacking hourglass modules together, and the impact of intermediate supervision. These are not mutually independent as we are limited in how we can apply intermediate supervision depending on the overall architectural design. We will show that applied separately neither of these choices demonstrates a significant impact on performance, but together we see a dramatic improvement in training speed and in the end, final pose estimation performance.

In comparing these different methods we offer the disclaimer that given unlimited time it is possible that all of these approaches may eventually converge to similar levels of performance. For this reason, in this evaluation we will compare the difference in training speed of these methods. The results of the different experiments can be seen in Figure 8 which shows average accuracy on the validation set as training progresses. The accuracy metric considers all joints excluding those associated with the head and torso.

First, to explore the effect of a stacked hourglass design we must demonstrate that the change in performance is a function of the architecture shape and not attributed to an increase in capacity with a larger, deeper network. To make this comparison, we consider two network designs. One is the stacked hourglass as described in this paper, and the other is a single hourglass but we double the number of residual modules at every point in the pipeline. This results in a network with the same number of layers and approximately the same number of parameters as our stacked network architecture. Due to the nature of rearranging the layers, the extended single hourglass has slightly more parameters. In Figure 8 these are referred to as HG-Stacked and HG respectively.

Next we determine how to apply intermediate supervision. For the stacked hourglass design we follow the procedure described earlier in the paper. After the first hourglass we use the features to generate an initial set of heatmaps and apply a separate loss with the same ground truth. Applying this same idea with a single hourglass is nontrivial. Since the higher order features are at lower resolutions, we generate intermediate predictions with lower resolution features and apply supervision with a downsampled version of the ground truth.

As seen in Figure 8, it was not until we applied intermediate supervision with a stacked version of the hourglass network that we saw a noticeable difference in training performance. For the majority of time training, the network has an average accuracy 3% above the rest. It maintains this gap even as validation accuracy begins to plateau for all experiments. In contrast, intermediate supervision with the single hourglass does not result in a noticeable difference in performance from the baseline. The stacked network without intermediate supervision similarly trains at about the same rate as the baseline.

Our last experiment tested an alternative to applying two separate losses. Instead, we experimented with adding the two heatmaps together and applying a single loss. In this way the second half of our stacked model could be seen as serving as a giant residual step to update the heatmap. This setup is a relaxed form of intermediate supervision because the two hourglasses have more freedom to divide up the work between them, whereas applying separate losses specifies the duty of each hourglass and allows no flexibility in the division of labor. We saw that this alternative setup performed worse, and in fact trained significantly slower than our baseline. This indicates that it is important to apply separate losses, the more restrictive form of intermediate supervision.

5 Further Analysis

5.1 Reevaluation of Initial Predictions

Given a stacked hourglass network, we compare the final output to the predictions produced by the first hourglass module. In Table 1 we compare PCKh performance between the two heatmap outputs, and present images in Figure 9 to visualize some interesting examples and to comment on the reevaluation done by the second half of the network.

It is worth noting from the numbers presented in Table 1 that even if the second half of the stacked hourglass architecture is discarded, performance on MPII is still competitive. It is compelling that an architecture with half the parameters, that runs in half the time can still do so well. The second half of the network proves critical in improving performance on harder joints, notably those of the lower body. We see a 4.7% difference in performance on the ankle which is generally one of the worst performing joints in pose estimation systems.

Looking at the images in Figure 9, we notice some adjustments made by the network worth extra attention. For example, with the fireman, it is apparent that the network is thrown off by the limb-like appearance of the firehose when it makes its first estimates. It is good to see the second prediction move the wrist estimate to the appropriate location, but perhaps more compelling is the change in predictions made for the opposite arm. Initially, the predictions for the fireman's right arm (which is hidden behind a firehose) mostly follow the visual cues of the visible left arm, but in the second output, they no longer correlate to any visible evidence whatsoever. Instead, the prediction corresponds better to a realistic projection of where the occluded limb should be.

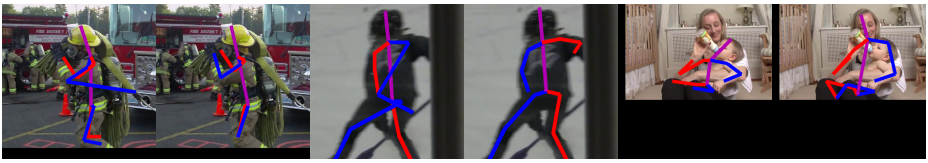


Fig. 9: Example validation images illustrating the change in predictions from the first hourglass (left) to the second (right)

	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle
Initial prediction	96.7	92.9	85.0	79.8	83.7	78.6	74.3
Final prediction	97.7	94.6	88.5	88.3	87.5	83.0	79.0

Table 1: Validation set performance contrasting predictions produced after the first hourglass, and the final output after passing through to the end of the network.

The other images present intriguing cases as well. For the hockey player, the network resolves its confusion over whether joints correspond to the left or right side of the body. And in the last image, we see a revised attempt to distinguish between the joints of the mother and her baby. With all of these cases, the network does the work often reserved for a graphical model, moving the final predictions towards a more coherent state.

5.2 Multiple People

The issue of coherence becomes especially important when there are multiple people in an image. The network has to make its decisions, but we have limited options for communicating who to annotate. For the purposes of this work, we center and scale the target person trusting that the input will be clear enough to parse. Unfortunately, this still leads to ambiguous situations when people are very close together or even overlapping as seen in Figure 10. Since we are training a system to generate pose predictions for a single person, the ideal output in an ambiguous situation would demonstrate a commitment to the joints of just one figure. Even if the predictions are lower quality, this would show a deeper understanding of the task at hand. Estimating a location for the wrist with a disregard for who the wrist may belong to is not the sort of behavior we want from a pose estimation system.

The results in Figure 10 are from an MPII test image. The network must produce predictions for both the boy and girl, and to do so, their respective center and scale annotations are provided. Using those values to crop input images for the network result in the first and third images of the figure. The center annotations for the two dancers are off by just 26 pixels in a 720x1280 image. Qualitatively, the most perceptible difference between the two input images is the change in scale. This difference is sufficient for the network to change its

Fig. 10: The difference made by a slight translation and change of scale of the input image. The network determines who to generate an annotation for based on the central figure. The scaling and shift right of the input image is enough for the network to switch its predictions.

estimate entirely and predict the annotations for the correct figure. Notice, it is not perfect, for the boy's annotations the network proposes the girl's ankles.

A more comprehensive management of annotations for multiple people is out of the scope of this work. Many of the system's failure cases are a result of confusing the joints of multiple people, but it is promising that in many examples with severe overlap of figures the network will appropriately pick out a single figure to annotate.

5.3 Occlusion

Occlusion performance can be difficult to assess as it often falls into two distinct categories. The first consists of cases where a joint is not visible but its position is apparent given the context of the image. MPII generally provides ground truth locations for these joints, and an additional annotation indicates their lack of visibility. The second situation, on the other hand, occurs when we have absolutely no information about where a particular joint might be. For example, images where only the upper half of the person's body is visible. In MPII these joints will not have a ground truth annotation associated with them.

During training, our system makes no use of the additional visibility annotations, but some form of supervision must be offered when no ground truth location is provided for a joint. One option would be to not backpropagate any error on the predictions of these joints. Instead, we supervise with a ground-truth heatmap of all zeros. This teaches the network not to activate at all when it is impossible to determine a joint's position in the image.

The PCK metric used when evaluating pose estimation systems does not offer any information to illustrate how well these situations are recognized by the network. If there is no ground truth annotation provided for a joint it is impossible to assess the quality of the prediction made by the system, so it is not counted towards the final reported PCK value. Because of this, there is no harm in generating predictions for all joints even though the predictions for completely occluded joints will make no sense. For use in a real system, a degree of metaknowledge is essential, and the understanding that no good prediction can be made on a particular joint is very important. By simply looking at the

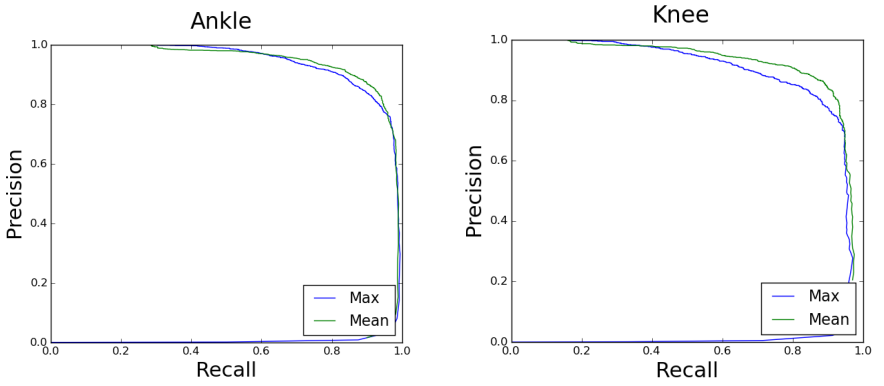


Fig. 11: Precision recall curves showing the accuracy of predicting whether an annotation is present for a joint given either the mean or max activation of its heatmap.

activation of a heatmap, our network gives consistent and accurate predictions of whether or not a ground truth annotation is available for a joint.

We consider the ankle and knee for this analysis since these are occluded most often. Lower limbs are frequently cropped from images, and if we were to always visualize all joint predictions of our network, our example pose figures would look unacceptable given the nonsensical lower body predictions made in these situations. For a simple way to filter out these cases we examine how well one can determine the presence of an annotation for a joint given the corresponding heatmap activation. We consider taking either the maximum value of the heatmap or its mean. The corresponding precision-recall curves can be seen in Figure 11. We find that based solely off of the mean activation of a heatmap it is possible to correctly assess the presence of an annotation for the knee with an accuracy of 96.0% and an annotation for the ankle with an accuracy of 93.8%. This was done on a validation set of 2958 samples of which 16.1% of possible knees and 28.4% of possible ankles do not have a ground truth annotation. This is a promising result demonstrating that the heatmap serves as a useful signal indicating cases of severe occlusion in images.

6 Conclusion

We demonstrate the powerful performance of a stacked hourglass network for producing human pose estimates. The network handles a diverse and challenging set of poses with the second half of the network providing a critical mechanism for reevaluation and assessment of predictions. Intermediate supervision is critical to efficiently training the network, and is best applied in the context of stacked hourglass modules. There are still difficulties that the network does not handle perfectly, though it shows robust performance to people in close proximity and to heavy occlusion.

References

1. Sapp, B., Taskar, B.: Modec: Multimodal decomposable models for human pose estimation. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, IEEE (2013) 3674–3681
2. Felzenszwalb, P., McAllester, D., Ramanan, D.: A discriminatively trained, multiscale, deformable part model. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, IEEE (2008) 1–8
3. Pishchulin, L., Andriluka, M., Gehler, P., Schiele, B.: Strong appearance and expressive spatial models for human pose estimation. In: Computer Vision (ICCV), 2013 IEEE International Conference on, IEEE (2013) 3487–3494
4. Bourdev, L., Malik, J.: Poselets: Body part detectors trained using 3d human pose annotations. In: Computer Vision, 2009 IEEE 12th International Conference on, IEEE (2009) 1365–1372
5. Johnson, S., Everingham, M.: Learning effective human pose estimation from inaccurate annotation. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, IEEE (2011) 1465–1472
6. Ramanan, D.: Learning to parse images of articulated objects. Advances in Neural Information Processing Systems **134** (2006)
7. Yang, Y., Ramanan, D.: Articulated human detection with flexible mixtures of parts. Pattern Analysis and Machine Intelligence, IEEE Transactions on **35**(12) (2013) 2878–2890
8. Ferrari, V., Marin-Jimenez, M., Zisserman, A.: Progressive search space reduction for human pose estimation. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, IEEE (2008) 1–8
9. Ladicky, L., Torr, P.H., Zisserman, A.: Human pose estimation using a joint pixel-wise and part-wise formulation. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, IEEE (2013) 3578–3585
10. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11) (1998) 2278–2324
11. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. (2012) 1097–1105
12. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. arXiv preprint arXiv:1409.4842 (2014)
13. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385 (2015)
15. Tompson, J.J., Jain, A., LeCun, Y., Bregler, C.: Joint training of a convolutional network and a graphical model for human pose estimation. In: Advances in Neural Information Processing Systems. (2014) 1799–1807
16. Tompson, J., Goroshin, R., Jain, A., LeCun, Y., Bregler, C.: Efficient object localization using convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 648–656
17. Pishchulin, L., Insafutdinov, E., Tang, S., Andres, B., Andriluka, M., Gehler, P., Schiele, B.: Deepcut: Joint subset partition and labeling for multi person pose estimation. arXiv preprint arXiv:1511.06645 (2015)

18. Wei, S.E., Ramakrishna, V., Kanade, T., Sheikh, Y.: Convolutional pose machines. arXiv preprint arXiv:1602.00134 (2016)
19. Carreira, J., Agrawal, P., Fragkiadaki, K., Malik, J.: Human pose estimation with iterative error feedback. arXiv preprint arXiv:1507.06550 (2015)
20. Fan, X., Zheng, K., Lin, Y., Wang, S.: Combining local appearance and holistic view: Dual-source deep neural networks for human pose estimation. arXiv preprint arXiv:1504.07159 (2015)
21. Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B.: 2d human pose estimation: New benchmark and state of the art analysis. In: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, IEEE (2014) 3686–3693
22. Johnson, S., Everingham, M.: Clustered pose and nonlinear appearance models for human pose estimation. In: BMVC. Volume 2. (2010) 5
23. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 3431–3440
24. Toshev, A., Szegedy, C.: Deeppose: Human pose estimation via deep neural networks. In: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, IEEE (2014) 1653–1660
25. Chen, X., Yuille, A.: Articulated pose estimation by a graphical model with image dependent pairwise relations. In: Advances in Neural Information Processing Systems (NIPS). (2014)
26. Ramakrishna, V., Munoz, D., Hebert, M., Bagnell, J.A., Sheikh, Y.: Pose machines: Articulated pose estimation via inference machines. In: Computer Vision–ECCV 2014. Springer (2014) 33–47
27. Jain, A., Tompson, J., LeCun, Y., Bregler, C.: Modeep: A deep learning framework using motion features for human pose estimation. In: Computer Vision–ACCV 2014. Springer (2014) 302–315
28. Shotton, J., Sharp, T., Kipman, A., Fitzgibbon, A., Finocchio, M., Blake, A., Cook, M., Moore, R.: Real-time human pose recognition in parts from single depth images. Commun. ACM **56**(1) (January 2013) 116–124
29. Chen, X., Yuille, A.: Parsing occluded people by flexible compositions. arXiv preprint arXiv:1412.1526 (2014)
30. Xie, S., Tu, Z.: Holistically-nested edge detection. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 1395–1403
31. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: Advances in neural information processing systems. (2014) 2366–2374
32. Farabet, C., Couprie, C., Najman, L., LeCun, Y.: Learning hierarchical features for scene labeling. Pattern Analysis and Machine Intelligence, IEEE Transactions on **35**(8) (2013) 1915–1929
33. Pinheiro, P., Collobert, R.: Recurrent convolutional neural networks for scene labeling. In: Proceedings of the 31st International Conference on Machine Learning (ICML-14). (2014) 82–90
34. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 2650–2658
35. Mathieu, M., Couprie, C., LeCun, Y.: Deep multi-scale video prediction beyond mean square error. arXiv preprint arXiv:1511.05440 (2015)
36. Couprie, C., Farabet, C., Najman, L., LeCun, Y.: Indoor semantic segmentation using depth information. arXiv preprint arXiv:1301.3572 (2013)

37. Bertasius, G., Shi, J., Torresani, L.: Deepedge: A multi-scale bifurcated deep network for top-down contour detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 4380–4389
38. Hariharan, B., Arbeláez, P., Girshick, R., Malik, J.: Hypercolumns for object segmentation and fine-grained localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 447–456
39. Noh, H., Hong, S., Han, B.: Learning deconvolution network for semantic segmentation. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 1520–1528
40. Zhao, J., Mathieu, M., Goroshin, R., Lecun, Y.: Stacked what-where auto-encoders. arXiv preprint arXiv:1506.02351 (2015)
41. Rematas, K., Ritschel, T., Fritz, M., Gavves, E., Tuytelaars, T.: Deep reflectance maps. arXiv preprint arXiv:1511.04384 (2015)
42. Zeiler, M.D., Krishnan, D., Taylor, G.W., Fergus, R.: Deconvolutional networks. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE (2010) 2528–2535
43. Collobert, R., Kavukcuoglu, K., Farabet, C.: Torch7: A matlab-like environment for machine learning. In: BigLearn, NIPS Workshop. Number EPFL-CONF-192376 (2011)
44. Tieleman, T., Hinton, G.: Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning (2012)