Human Pose Estimation with Iterative Error Feedback

João Carreira
carreira@eecs.berkeley.edu
UC Berkeley

Pulkit Agrawal
pulkitag@eecs.berkeley.edu
UC Berkeley

Katerina Fragkiadaki
katef@eecs.berkeley.edu
UC Berkeley

Jitendra Malik
malik@eecs.berkeley.edu
UC Berkeley

Abstract

Hierarchical feature extractors such as Convolutional Networks (ConvNets) have achieved strong performance on a variety of classification tasks using purely feedforward processing. Feedforward architectures can learn rich representations of the input space but do not explicitly model dependencies in the output spaces, that are quite structured for tasks such as articulated human pose estimation or object segmentation. Here we propose a framework that expands the expressive power of hierarchical feature extractors to encompass both input and output spaces, by introducing top-down feedback. Instead of directly predicting the outputs in one go, we use a self-correcting model that progressively changes an initial solution by feeding back error predictions, in a process we call Iterative Error Feedback (IEF). We show that IEF improves over the state-of-the-art on the task of articulated human pose estimation on the challenging MPII dataset.

1 Introduction

Feature extractors such as Convolutional Networks (ConvNets) [21] represent images using a multi-layered hierarchy of features and share structural and functional similarities with the visual pathway of the human brain [11, 1]. Feature computation in these models is purely feedforward, however, unlike in the human visual system where feedback connections abound [9, 19, 20]. Feedback can be used to modulate and specialize feature extraction in early layers in order to model temporal and spatial context (e.g. *priming* [34]), to leverage prior knowledge about shape for segmentation and 3D perception, or simply for guiding visual attention to image regions relevant for the task under consideration.

Here we are interested in using feedback to build predictors that can naturally handle complex, structured output spaces. We will use as running example the task of 2D human pose estimation [39, 31, 30], where the goal is to infer the 2D locations of a set of keypoints such as wrists, ankles, etc, from a single RGB image. The space of 2D human poses is highly structured because of body part proportions, left-right symmetries, interpenetration constraints, joint limits (e.g. elbows do not bend back) and physical connectivity (e.g. wrists are rigidly related to elbows), among others. Modeling this structure should make it easier to pinpoint the visible keypoints and make it possible to estimate the occluded ones.

Our main contribution is in providing a generic framework for modelling rich structure in both input and output spaces by learning hierarchical feature extractors over their joint space. We achieve this by incorporating top-down feedback – instead of trying to directly predict the target outputs,

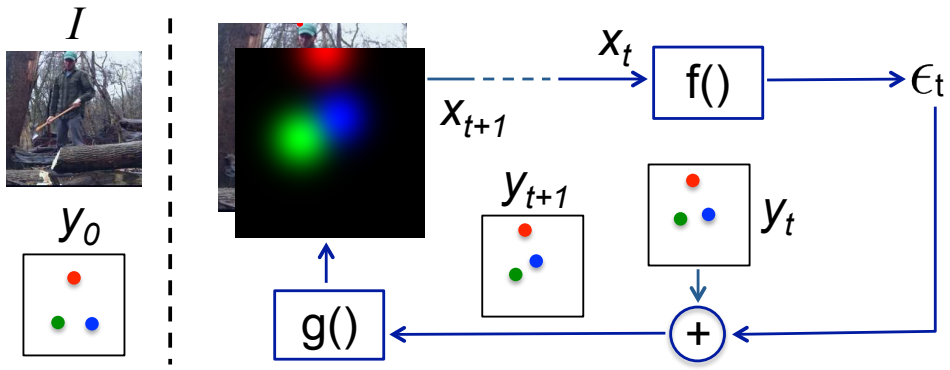

Figure 1: An implementation of Iterative Error Feedback (IEF) for 2D human pose estimation. The left panel shows the input image I and the initial guess of keypoints y_0, represented as a set of 2D points. For the sake of illustration we show only shown 3 out of 17 keypoints, corresponding to the right wrist (green), left wrist (blue) and top of head (red). Consider iteration t: predictor f receives the input x_t – image I stacked with a "rendering" of current keypoint positions y_t – and outputs a correction ϵ_t. This correction is added to y_t, resulting in new keypoint position estimates y_{t+1}. The new keypoints are rendered by function g and stacked with image I, resulting in x_{t+1}, and so on iteratively. Function f was modeled here as a ConvNet. Function g converts each 2D keypoint position into one gaussian heatmap channel. For 3 keypoints there are 3 stacked heatmaps which are visualized as channels of a color image. In contrast to previous works, in our framework multi-layered hierarchical models such as ConvNets can learn rich models over the joint space of body configurations and images.

as in feedforward processing, we predict what is wrong with their current estimate and correct it iteratively. We call our framework Iterative Error Feedback, or IEF.

In IEF, a feedforward model f operates on the augmented input space created by concatenating (denoted by $\oplus$) the RGB image I with a visual representation g of the estimated output y_t to predict a "correction" (ϵ_t) that brings y_t closer to the ground truth output y. The correction signal ϵ_t is applied to the current output y_t to generate y_{t+1} and this is converted into a visual representation by g, that is stacked with the image to produce new inputs $x_{t+1} = I \oplus g(y_t)$ for f, and so on iteratively. This procedure is initialized with a guess of the output (y_0) and is repeated until a predetermined termination criterion is met. The model is trained to produce bounded corrections at each iteration, e.g. $||\epsilon_t||_2 < L$. The motivation for modifying y_t by a bounded amount is that the space of x_t is typically highly non-linear and hence local corrections should be easier to learn. The working of our model can be mathematically described by the following equations:

$$\epsilon_t = f(x_t) \tag{1}$$

$$y_{t+1} = y_t + \epsilon_t \tag{2}$$

$$x_{t+1} = I \oplus g(y_{t+1}), \tag{3}$$

where functions f and g have additional learned parameters Θ_f and Θ_g, respectively. Although we have used the predicted error to additively modify y_t in equation 2, in general y_{t+1} can be a result of an arbitrary non-linear function that operates on y_t, ϵ_t.

In the running example of human pose estimation, y_t is vector of retinotopic positions of all keypoints that are individually mapped by g into heatmaps (i.e. K heatmaps for K keypoints). The heatmaps are stacked together with the image and passed as input to f (see figure 1 for an overview). The "rendering" function g in this particular case is not learnt – it is instead modelled as a 2D gaussian having a fixed standard deviation and centered on the keypoint location. Intuitively, these heatmaps encode the current belief in keypoint locations in the image plane and thus form a natural representation for learning features over the joint space of body configurations and the RGB image. The dimensionality of inputs to f is $H \times W \times (K+3)$, where H, W represent the height and width of the image and $(K+3)$ correspond to K keypoints and the 3 color channels of the image. We

model f with a ConvNet with parameters Θ_f (i.e. ConvNet weights). As the ConvNet takes $I \oplus g(y_t)$ as inputs, it has the ability to learn features over the joint input-output space.

2 Learning

In order to infer the ground truth output (y), our method iteratively refines the current output (y_t). At each iteration, f predicts a correction (ϵ_t) that locally improves the current output. Note that we train the model to predict bounded corrections, but we do not enforce any such constraints at test time. The parameters (Θ_f, Θ_g) of functions f and g in our model, are learnt by optimizing equation 4,

$$\min_{\Theta_f, \Theta_g} \sum_{t=1}^{T} h(\epsilon_t, e(y, y_t)) \tag{4}$$

where, ϵ_t and $e(y, y_t)$ are predicted and target bounded corrections, respectively. The function h is a measure of distance, such as a quadratic loss. T is the number of correction steps taken by the model. T can either be chosen to be a constant or, more generally, be a function of ϵ_t (i.e. a termination condition).

We optimize this cost function using stochastic gradient descent with every correction step being an independent training example. We grow the training set progressively: we start by learning with the samples corresponding to the first step for N epochs, then add the samples corresponding to the second step and train another N epochs, and so on, such that early steps get optimized longer – they get consolidated. See fig. 2 for an illustration.

As we only assume that the ground truth output (y) is provided at training time, it is unclear what the intermediate targets (y_t) should be. One simple strategy would be to predefine y_t for every iteration using interpolation between y_0 and y, obtaining $(y_0, y_1, ..y)$. We use this strategy to initialize each new step but then let it progressively drift to one that comes more naturally to the model during the learning process. In practice, we update the intermediate targets to what the current model parameters predict starting from y_0, every time we enlarge the training set. This idea relies on not letting the model overfit to the initial trajectory, and that hence there is some (helpful) drift – this can be achieved by early stopping, using a small N. We call this learning procedure *Latent Path Consolidation* and algorithm 1 provides a formal description.

The LPC algorithm generates the target bounded correction for every iteration by computing the function $e(y, y_t)$. e can take different forms for different problems. If for instance the output is 1D, then $e(y, y_t) = max(\text{sign}(y - y_t) \cdot \alpha,\ y - y_t)$ would imply that the target "bounded" error will correct y_t by a maximum amount of α in the direction of y.

2.1 Learning Human Pose Estimation

Human pose was represented by a set of 2D keypoint locations $y : \{y^k \in \Re^2, k \in [1, K]\}$ where K is the number of keypoints and y^k denotes the k^{th} keypoint. The predicted location of keypoints at the t^{th} iteration has been denoted by $y_t : \{y_t^k, k \in [1, K]\}$. The rendering of y_t as heatmaps concatenated with the image was provided as inputs to a ConvNet (see section 1 for details). The ConvNet was trained to predict a sequence of "bounded" corrections for each keypoint (ϵ_t^k) . The corrections were used to iteratively refine the keypoint locations.

Let $u = y^k - y_t^k$ and the corresponding unit vector be $\hat{u} = \frac{u}{||u||_2}$. Then, the target "bounded" correction for the t^{th} iteration and k^{th} keypoint was calculated as:

$$e(y^k, y_t^k) = \min(L, ||u||) \cdot \hat{u} \tag{5}$$

where L denotes the maximum displacement for each keypoint location, which we set set to 20 pixels in our experiments. The target corrections were calculated independently for each keypoint in each example and we used an L_2 regression loss to model h in eq. 4.

We initialized y_0 as the median of ground truth 2D keypoint locations on training images and trained a model for $T = 4$ steps, using $N = 3$ epochs for each new step. We found the fourth step to have little effect on accuracy and used 3 steps in practice at test time.

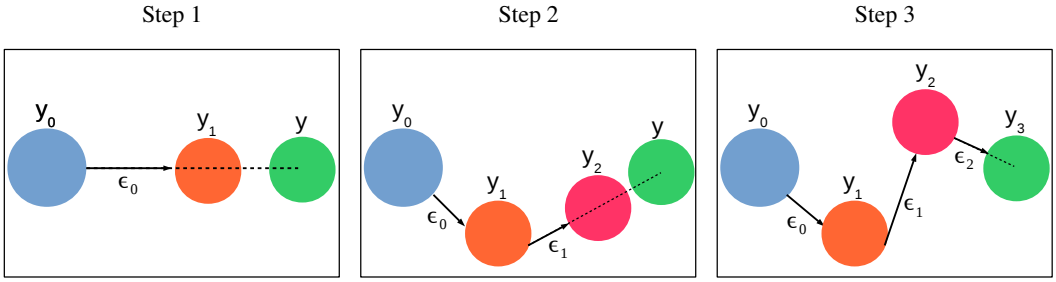

Figure 2: Evolution of the training set in the Latent Path Consolidation learning algorithm, illustrated for a single training image I having ground truth output y and considering paths having three correction steps. We first train the model to predict a correction ϵ_0, which transforms y_0 (the initial guess of y) into y_1 (step 1). Parameters are learned using stochastic gradient descent for this initial correction step considering all training images, for N epochs. Then training examples for the second correction step are added and parameters are again updated with SGD for another N epochs. The process is repeated one last time for the third correction step. Every time the training set is to be augmented with examples for a new step, intermediate corrections ϵ_t are recomputed based on the most recent model parameters, hence the paths are *latent*. The earlier correction steps are trained for more epochs and, hence, change less – we call this *consolidation*. The new corrections, which generate the tip of the path, are bounded and defined on a "straight-line" to the ground truth output y. The intuition for this full procedure is that a more natural manifold of variation of the outputs may naturally emerge, by providing flexible guidance.

Algorithm 1 Learning Iterative Error Feedback with Latent Path Consolidation

procedure LPC-LEARN
  Initialize y_0
  $E \leftarrow \{\}$
  for $t \leftarrow 0$ to $(T_{steps} - 1)$ **do**
    for $s \leftarrow 0$ to t **do**
      if $s < (T_{steps} - 1)$ **then**
        Assemble inputs: $x_s \leftarrow I \oplus g(y_s)$
        Predict corrections using current parameters Θ_f: $\epsilon_s \leftarrow f(x_s)$
        Update latent target outputs: $y_{s+1} \leftarrow y_s + \epsilon_s$
      else
        $\epsilon_s \leftarrow e(y, y_s)$
      end if
      $E \leftarrow E \cup \epsilon_s$
    end for
    for $j \leftarrow 1$ to N **do**
      Update Θ_f and Θ_g using SGD with loss h and target corrections E
    end for
  end for
end procedure

ConvNet architecture. We employed a standard ConvNet architecture pre-trained on Imagenet: the very deep VGG-16 [28]. We modified the filters in the first convolution layer (conv-1) to account for 17 additional channels due to 17 keypoints. In our model, the conv-1 filters operated on 20 channel inputs. The weights of the first three conv-1 channels (i.e. the ones corresponding to the image) were initialized using the weights learnt by pre-training on Imagenet. The weights corresponding to the remaining 17 channels were randomly initialized with white noise of variance 0.1. We discarded the last layer of 1000 units that predicted the Imagenet classes and replaced it with a layer containing 32 units, encoding the continuous $2D$ correction [1] expressed in cartesian coordinates (the

[1] Again, we do not bound explicitly the correction at test time, instead the network is taught to predict bounded corrections.

17th "keypoint" is the location of one point anywhere inside a person, marking her, and which is provided as input both during training and testing, see section 3). We used a fixed ConvNet input size of 224×224.

3 Results

We experimented with human pose estimation on the most challenging 2D dataset in the field: the MPII Human Pose dataset [2], which features significant scale variation, occlusion, and multiple people interacting. The goal is to, for each person in every image, predict the 2D locations of all its annotated keypoints. Most previous work has considered simplifications of the problem, namely by using ground truth person scale information, which constrains size of parts and the distance between them. One exception is the approach of Tompson et al [30], the current leading approach on this problem on the MPII dataset by a large margin, which localizes joints using multiple sliding-windows with loose coupling. Our main experimental goals were to analyze the value of LPC learning and IEF for human pose estimation.

Experimental Details. Human pose is represented as a set of 16 keypoints on MPII. An additional *marking-point* in each person is available both for training and testing, located somewhere inside each person's boundary. We represent this point as an additional channel and stack it with the other 16 keypoint channels and the 3 RGB channels that we feed as input to a ConvNet. We used the same publicly available train/validation splits of [30]. We evaluated the accuracy of our algorithm on the validation set using the standard PCKh metric [2], and also submitted results for evaluation on the test set once, to obtain the final score.

We cropped 9 square boxes centered on the marking-point of each person, sampled uniformly over scale, from $1.4\times$ to $0.3\times$ of the smallest side of the image and resized them to 256×256 pixels. Padding was added as necessary for obtaining these dimensions and the amount of training data was further doubled by also mirroring the images. We used the ground truth height of each person at training time, which is provided on MPII, and select as training examples the 3 boxes for each person having a side closest to $1.2\times$ the person height in pixels. We then trained VGG-16 models on random crops of 224×224 patches, using 3 epochs of consolidation for each of 4 steps. At test time, we predict which one of the 9 boxes is closest to $1.2\times$ the height of the person in pixels, using a VGG-S ConvNet trained for that task using an L_2 regression loss. We then align our model to the center 224×224 patch of the selected window.

We train our models using keypoint positions for both visible and occluded keypoints, which MPII provides in many cases whenever they project on to the image (the exception are people truncated by the image border). We zero out the backpropagated gradients for missing keypoint annotations. Note that often keypoints lie outside the cropped image passed to the ConvNet, but this poses no issues to our formulation – keypoints outside the image can be predicted and are still visible to the ConvNet as tails of rendered gaussians.

Iterative vs. Single Step Correction. We first compare iterative correction with single step correction. We finetuned the same pretrained model in both cases, limiting the step to 20 pixels for the iterative approach. For the single step model we did not stack additional channels to the image (doing so produced similar results). We used the VGG-16 network on this experiment and trained both models a same number of epochs. The results in Fig. 3 show a large advantage of iterative correction, by over 10 PCKh points, on the validation set. Fig. 4 shows the evolution of PCKh for individual keypoints on ground truth bounding boxes of the validation set.

Comparison with State-of-the-Art. Previous state-of-the-art methods not employing ConvNets [39, 26], have only been evaluated on MPII with images normalized using ground truth scale information. Their results are shown in table 1. We considered the more automatic setting where scale information is not provided at test time but our model still outperformed these methods by roughly 30 PCKh points. The only other method reporting results in this setting is the sliding-window approach of Tompson et al [30], which we also outperform by a margin. The emphasis in Tompson et al's system was however efficiency and they trained and tested their model using original image scales – searching over a multiscale image pyramid could presumably improve performance.

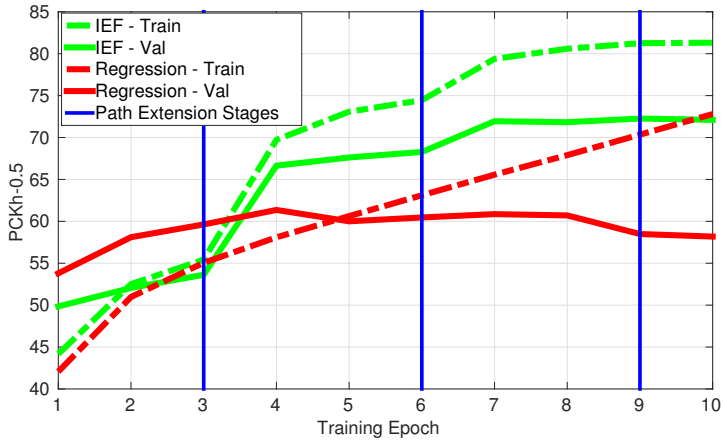

Figure 3: Training and validation PCKh-0.5 scores when training a VGG-16 model for 10 epochs with scale augmentation, using either direct regression (red) or IEF (green). The IEF model parameters are consolidated with stochastic gradient descent during 3 epochs for each step, bounded to a maximum of 20-pixel corrections per keypoint and learned through a total of 3 correction steps in this setup, whose transitions are marked in blue. IEF achieves significantly more accurate results than direct regression, by learning powerful models of the output dependencies.

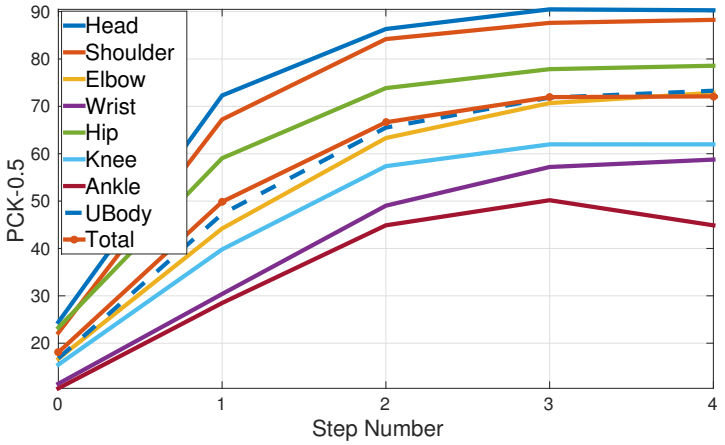

Figure 4: Evolution of PCKh at 0.5 overlap as function of correction step number on the MPII-human-pose validation set, using the finetuned VGG-16 network. The model aligns more accurately to parts like the head and shoulders, which is natural, because these parts are easier to discriminate from the background and have more consistent appearance than limbs.

4 Related Work

There is a rich literature on structured output learning [32, 6] (e.g. see references in [24]) but it is a relatively modern topic in conjunction with feature learning, for computer vision [3, 17, 30, 21].

Here we proposed a feedback-based framework for structured-output learning. Neuroscience models of the human brain suggest that feedforward connections act as information carriers while numerous feedback connections act as modulators or competitive inhibitors to aid feature grouping [12], figure-ground segregation [14] and object recognition [37]. In computer vision, feedback has been primarily used so far for learning selective attention [23]; in [23] attention is implemented by estimating a bounding box in an image for the algorithm to process next, while in [29] attention is formed by selecting some convolutional features over others (it does not have a spatial dimension).

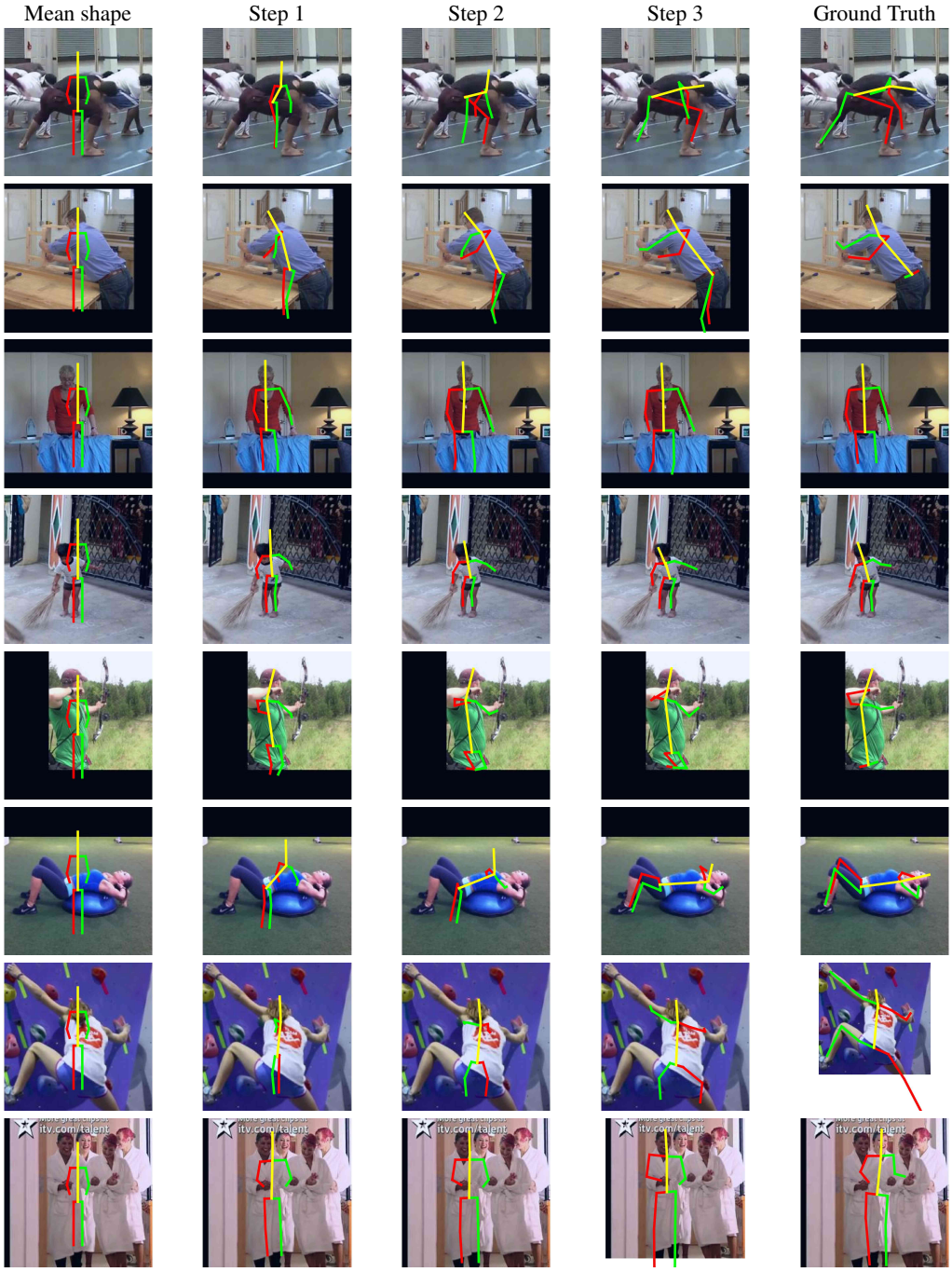

Figure 5: Example poses obtained using the proposed method IEF on the MPII validation set. From left to right we show the sequence of corrections the method makes – on the right is the ground truth pose, including annotated occluded keypoints, which are not evaluated. Note that IEF is robust to left-right ambiguities and is able to rotate the initial pose for a full 180 (first row), can align across occlusions (third row) and can handle scale variation (fourth and fifth rows). The bottom two rows show failure cases. In the first one, the predicted configuration captures the gist of the pose but is misaligned and not scaled properly. The second case shows several people closely interacting and the system fits to the wrong person. The black borders show padding. Best seen in color and with zoom.

	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	UBody	FBody
Yang & Ramanan [39]	73.2	56.2	41.3	32.1	36.2	33.2	34.5	43.2	44.5
Pischulin et al [26]	74.2	49.0	40.8	34.1	36.5	34.4	35.1	41.3	44.0
Tompson et al. [30]	83.4	77.5	67.5	59.8	64.6	55.6	46.1	68.3	66.0
IEF	**92.4**	**89.0**	**73.9**	**60.5**	**79.3**	**62.8**	**49.4**	**74.5**	**73.7**

Table 1: MPII test set PCKh-0.5 results for Iterative Error Feedback (IEF) and previous approaches. Ground truth scale information is not used at test time in both Tompson et al. and in our method. UBody and FBody stand for upper body and full body, respectively. Best PCKh results are highlighted in bold.

Also relevant to this work is the recurrent network proposed by Pinheiro and Collobert [25], which recomputes pixel labellings using increasingly larger spatial context.

Stacked inference methods [27, 35, 36, 33], which employ forms of cascades, are another related family of methods. Differently, some of these methods consider each output in isolation [31], all use different weights or learning models in each stage of inference [30] and they do not optimize for correcting their current estimates but rather attempt to predict the answer from scratch at each stage [22]. In contrast, we predict additive corrections to our solution, which are updated at each iteration, rather than aiming for the answer from scratch at each iteration. Graphical models can also encode dependencies between outputs and are still popular in many applications, including human pose estimation [4].

Another line of work aims to inject class-specific spatial priors using coarse-to-fine processing, e.g. features arising from different layers of ConvNets were recently used for instance segmentation and keypoint prediction [13]. For pose inference, combining multiple scales [8, 30] aids in capturing subtle long-range dependencies (e.g. distinguishing the left and right sides of the body which depend on whether a person is facing the camera). The system in our human pose estimation example can be seen as closest to approaches employing "pose-indexed features" [10, 7, 15], but leveraging hierarchical feature learning.

Classic spatial alignment and warping computer vision models, such as snakes, [18] and Active Appearance Models (AAMs) [5] have similar goals as the proposed IEF, but are not learned end-to-end – or learned at all – employ linear shape models and hand designed features and require slower gradient computation which often takes many iterations before convergence. They can get stuck in poor local minimas even for constrained variation (AAMs and small out-of-plane face rotations). IEF, on the other hand, is able to minimize over rich articulated human 3D pose variation, starting from a mean shape. Although extensions that use learning to drive the optimization have been proposed [38], typically these methods still require manually defined energy functions to measure goodness of fit.

5 Conclusions

While standard ConvNets offer hierarchical representations that can capture the patterns of images at multiple levels of abstraction, the outputs are typically modeled as flat image or pixel-level 1-of-K labels, or slightly more complicated hand-designed representations. We aimed in this paper to mitigate this asymmetry by introducing Iterative Error Feedback (IEF), which extends hierarchical representation learning to output spaces, while leveraging at heart the same machinery. IEF works by, in broad terms, moving the emphasis from the problem of *predicting* the state of the external world to one of *correcting* the expectations about it, which is achieved by introducing a simple feedback connection in standard models.

In our pose estimation working example we opted for feeding pose information only into the first layer of the ConvNet for the sake of simplicity. This information may also be helpful for mid-level layers, so as to modulate not only edge detection, but also processes such as junction detection or contour completion which advanced feature extractors may need to compute. We also have only experimented so far feeding back "images" made up of gaussian distributions. There may be more powerful ways to render top-down pose information using parametrized computational blocks (e.g. deconvolution) that can then be learned jointly with the rest of the model parameters using stan-

dard backpropagation. This is desirable in order to attack problems with higher-dimensional output spaces such as 3D human pose estimation [16] or segmentation.

Acknowledgement

This work was supported in part by ONR MURI N00014-14-1-0671 and N00014-10-1-0933. João Carreira was partially supported by the Portuguese Science Foundation, FCT, under grant SFRH/BPD/84194/2012. Pulkit Agrawal was partially supported by a Fulbright Science and Technology Fellowship. We gratefully acknowledge NVIDIA corporation for the donation of Tesla GPUs for this research. We thank Georgia Gkioxari and Carl Doersch for helpful comments.

Appendix 1

Additional positive and negative results are provided, respectively, in figs. 6 and 7.

References

[1] Pulkit Agrawal, Dustin Stansbury, Jitendra Malik, and Jack L Gallant. Pixels to voxels: Modeling visual representation in the human brain. *arXiv preprint arXiv:1407.5104*, 2014.

[2] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 3686–3693. IEEE, 2014.

[3] Liang-Chieh Chen, Alexander G Schwing, Alan L Yuille, and Raquel Urtasun. Learning deep structured models. *arXiv preprint arXiv:1407.2538*, 2014.

[4] Xianjie Chen and Alan L Yuille. Articulated pose estimation by a graphical model with image dependent pairwise relations. In *Advances in Neural Information Processing Systems*, pages 1736–1744, 2014.

[5] Timothy F Cootes, Christopher J Taylor, David H Cooper, and Jim Graham. Active shape models-their training and application. *Computer vision and image understanding*, 61(1):38–59, 1995.

[6] Hal Daumé III, John Langford, and Daniel Marcu. Search-based structured prediction. *Machine learning*, 75(3):297–325, 2009.

[7] P. Dollár, P. Welinder, and P. Perona. Cascaded pose regression. In *CVPR*, 2010.

[8] Xiaochuan Fan, Kang Zheng, Yuewei Lin, and Song Wang. Combining local appearance and holistic view: Dual-source deep neural networks for human pose estimation. June 2015.

[9] Daniel J. Felleman and David C. Van Essen. Distributed Hierarchical Processing in the Primate Cerebral Cortex. *Cerebral Cortex*, 1(1):1–47, January 1991.

[10] Francois Fleuret and Donald Geman. Stationary features and cat detection. Technical Report Idiap-RR-56-2007, 0 2007.

[11] Kunihiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4):193–202, 1980.

[12] Charles D. Gilbert and Mariano Sigman. Brain states: Top-down influences in sensory processing. *Neuron*, 54(5):677 – 696, 2007.

[13] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Hypercolumns for object segmentation and fine-grained localization. *arXiv preprint arXiv:1411.5752*, 2014.

[14] J. M. Hupe, A. C. James, B. R. Payne, S. G. Lomber, P. Girard, and J. Bullier. Cortical feedback improves discrimination between figure and background by V1, V2 and V3 neurons. *Nature*, 394(6695):784–787, August 1998.

[15] Catalin Ionescu, Joao Carreira, and Cristian Sminchisescu. Iterated second-order label sensitive pooling for 3d human pose estimation. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1661–1668. IEEE, 2014.

[16] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(7):1325–1339, 2014.

[17] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep structured output learning for unconstrained text recognition. *ICLR 2015*, 2014.

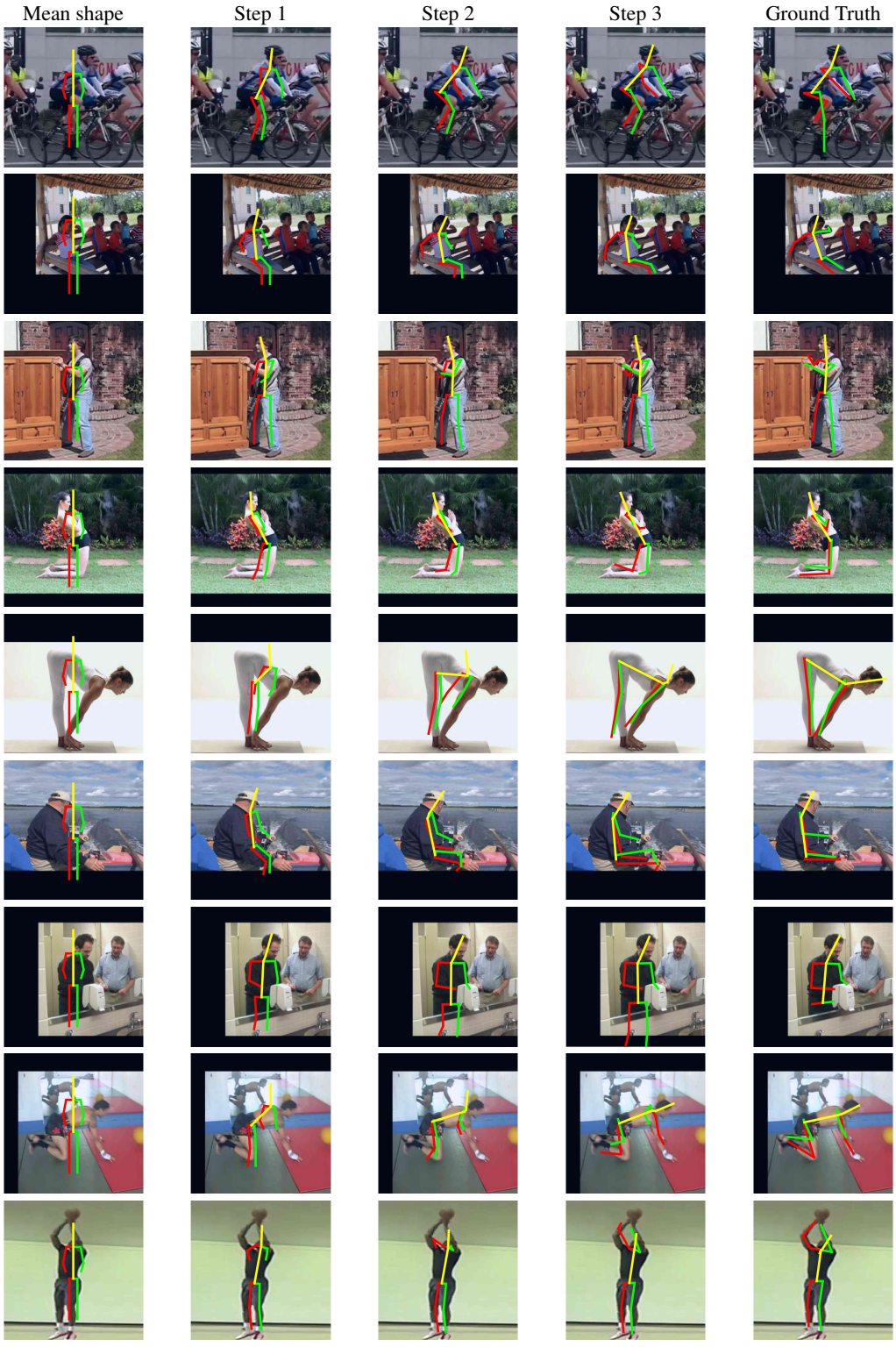

Figure 6: Additional examples of successful cases, with PCKh-0.5 scores above 50.0.

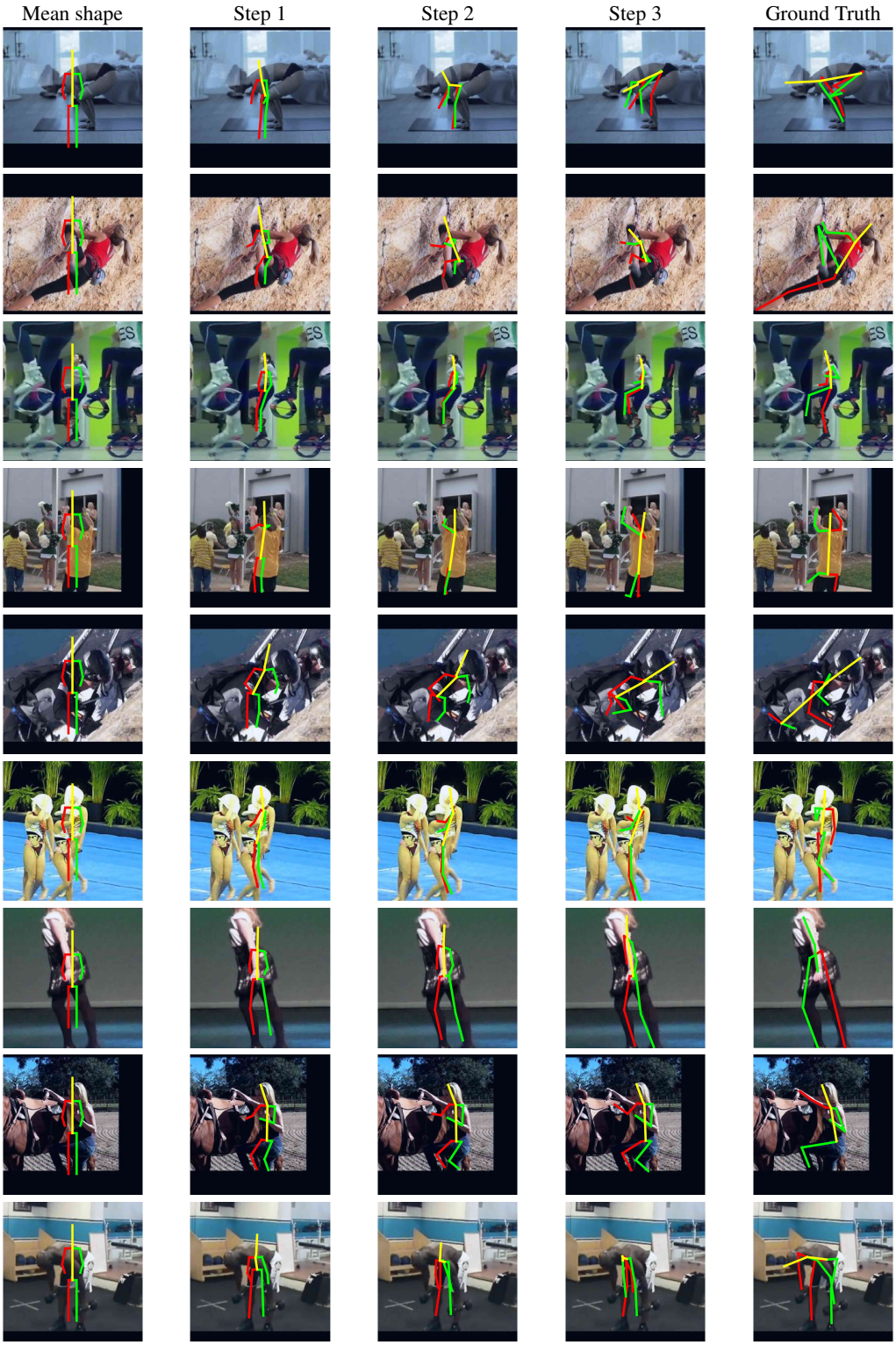

Figure 7: Additional examples of failures, where PCKh-0.5 is lower than 50.0.

[18] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988.

[19] Baker CI Ungerleider LG Mishkin M. Kravitz DJ, Saleem KS. The ventral visual pathway: An expanded neural framework for the processing of object quality. volume 17(1), 2013.

[20] Victor A. F. Lamme and Pieter R. Roelfaema. the distinct modes of vision offered by feedforward and recurrent processing. *Trends in Neurosciences*, 23:571, 2000.

[21] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[22] Quannan Li, Jingdong Wang, Zhuowen Tu, and David P Wipf. Fixed-point model for structured labeling. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 214–221, 2013.

[23] Volodymyr Mnih, Nicolas Heess, Alex Graves, and Koray Kavukcuoglu. Recurrent models of visual attention. In Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2204–2212. Curran Associates, Inc., 2014.

[24] Sebastian Nowozin and Christoph H Lampert. Structured learning and prediction in computer vision. *Foundations and Trends® in Computer Graphics and Vision*, 6(3–4):185–365, 2011.

[25] Pedro Pinheiro and Ronan Collobert. Recurrent convolutional neural networks for scene labeling. In Tony Jebara and Eric P. Xing, editors, *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 82–90. JMLR Workshop and Conference Proceedings, 2014.

[26] Leonid Pishchulin, Mykhaylo Andriluka, Peter Gehler, and Bernt Schiele. Poselet conditioned pictorial structures. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 588–595. IEEE, 2013.

[27] Varun Ramakrishna, Daniel Munoz, Martial Hebert, James Andrew Bagnell, and Yaser Sheikh. Pose machines: Articulated pose estimation via inference machines. In *Computer Vision–ECCV 2014*, pages 33–47. Springer International Publishing, 2014.

[28] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[29] Marijn F Stollenga, Jonathan Masci, Faustino Gomez, and Jürgen Schmidhuber. Deep networks with internal selective attention through feedback connections. In *Advances in Neural Information Processing Systems*, pages 3545–3553, 2014.

[30] Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann LeCun, and Christoph Bregler. Efficient object localization using convolutional networks. June 2015.

[31] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1653–1660. IEEE, 2014.

[32] Ioannis Tsochantaridis, Thomas Hofmann, Thorsten Joachims, and Yasemin Altun. Support vector machine learning for interdependent and structured output spaces. In *Proceedings of the twenty-first international conference on Machine learning*, page 104. ACM, 2004.

[33] Zhuowen Tu. Auto-context and its application to high-level vision tasks. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[34] Endel Tulving and Daniel L Schacter. Priming and human memory systems. *Science*, 247(4940):301–306, 1990.

[35] David Weiss, Benjamin Sapp, and Ben Taskar. Structured prediction cascades. *arXiv preprint arXiv:1208.3279*, 2012.

[36] David H Wolpert. Stacked generalization. *Neural networks*, 5(2):241–259, 1992.

[37] Dean Wyatte, Tim Curran, and Randall C. O'Reilly. The limits of feedforward vision: Recurrent processing promotes robust object recognition when objects are degraded. *J. Cognitive Neuroscience*, pages 2248–2261, 2012.

[38] Xuehan Xiong and Fernando De la Torre. Supervised descent method and its applications to face alignment. In *CVPR*.

[39] Yi Yang and Deva Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, 2011.