

# Human Pose Estimation with Iterative Error Feedback

João Carreira\*

carreira@eecs.berkeley.edu  
UC Berkeley

Katerina Fragkiadaki†

katef@eecs.berkeley.edu  
UC Berkeley

Pulkit Agrawal

pulkitag@eecs.berkeley.edu  
UC Berkeley

Jitendra Malik

malik@eecs.berkeley.edu  
UC Berkeley

## Abstract

*Hierarchical feature extractors such as Convolutional Networks (ConvNets) have achieved impressive performance on a variety of classification tasks using purely feed-forward 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). IEF shows excellent performance on the task of articulated pose estimation in the challenging MPII and LSP benchmarks, matching the state-of-the-art without requiring ground truth scale annotation.*

## 1. Introduction

Feature extractors such as Convolutional Networks (ConvNets) [23] represent images using a multi-layered hierarchy of features and are inspired by the structure and functionality of the visual pathway of the human brain [13, 1]. Feature computation in these models is purely feed-forward, however, unlike in the human visual system where feedback connections abound [11, 21, 22]. Feedback can be used to modulate and specialize feature extraction in early layers in order to model temporal and spatial context (e.g. *priming* [42]), 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 [48, 39, 37, 28], 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 modeling 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, 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” ( $\epsilon_t$ ) that brings  $y_t$  closer to the ground truth output  $y$ . The correction signal  $\epsilon_t$  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

\*Now at Google DeepMind.

†Now at Google.

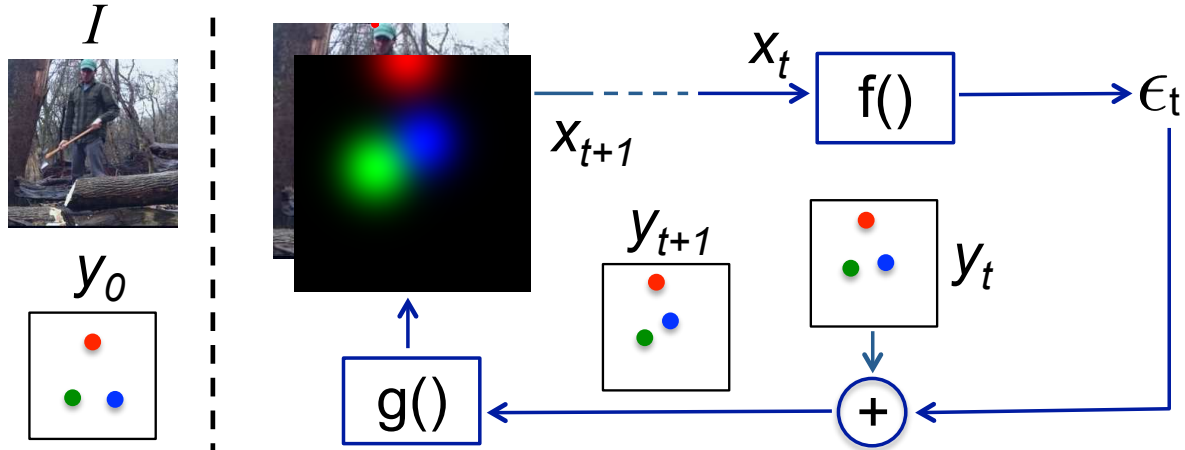


Figure 1: An implementation of Iterative Error Feedback (IEF) for 2D human pose estimation. The left panel shows the input image  $I$  and the initial guess of keypoints  $y_0$ , represented as a set of 2D points. For the sake of illustration we show only 3 out of 17 keypoints, corresponding to the right 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  $\epsilon_t$ . 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.

learn. The working of our model can be mathematically described by the following equations:

$$\epsilon_t = f(x_t) \quad (1)$$

$$y_{t+1} = y_t + \epsilon_t \quad (2)$$

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

where functions  $f$  and  $g$  have additional learned parameters  $\Theta_f$  and  $\Theta_g$ , 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  $y_t, \epsilon_t$ .

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  $\Theta_f$  (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 ( $\epsilon_t$ ) 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 ( $\Theta_f, \Theta_g$ ) of functions  $f$  and  $g$  in our model, are learnt by optimizing equation 4,

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

where,  $\epsilon_t$  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  $\epsilon_t$  (i.e. a termination condition).

We optimize this cost function using stochastic gradient descent (SGD) with every correction step being an independent training example. We grow the training set progres-

---

**Algorithm 1** Learning Iterative Error Feedback with Fixed Path Consolidation

---

```
1: procedure FPC-LEARN
2:   Initialize  $y_0$ 
3:    $E \leftarrow \{\}$ 
4:   for  $t \leftarrow 1$  to  $(T_{steps})$  do
5:     for all training examples  $(I, y)$  do
6:        $\epsilon_t \leftarrow e(y, y_t)$ 
7:     end for
8:      $E \leftarrow E \cup \epsilon_t$ 
9:     for  $j \leftarrow 1$  to  $N$  do
10:      Update  $\Theta_f$  and  $\Theta_g$  with SGD, using loss  $h$ 
      and target corrections  $E$ 
11:    end for
12:  end for
13: end procedure
```

---

sively: 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.

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. The simplest strategy, which we employ, is to predefine  $y_t$  for every iteration using a set of fixed corrections  $e(y, y_t)$  starting from  $y_0$ , obtaining  $(y_0, y_1, \dots, y)$ . We call our overall learning procedure *Fixed Path Consolidation (FPC)* which is formally described by algorithm 1.

The target bounded corrections for every iteration are computed using a function  $e(y, y_t)$ , which 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  $\alpha$  in the direction of  $y$ .

## 2.1. Learning Human Pose Estimation

Human pose was represented by a set of 2D keypoint locations  $y : \{y^k \in \mathbb{R}^2, k \in [1, K]\}$  where  $K$  is the number of keypoints and  $y^k$  denotes the  $k^{\text{th}}$  keypoint. The predicted location of keypoints at the  $t^{\text{th}}$  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 ( $\epsilon_t^k$ ). 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  $t^{\text{th}}$  iteration and  $k^{\text{th}}$  keypoint was calculated as:

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

where  $L$  denotes the maximum displacement for each keypoint location. An interesting property of this function is that it is constant while a keypoint is far from the ground truth and varies only in scale when it is closer than  $L$  to the ground truth. This simplifies the learning problem: given an image and a fixed initial pose, the model just needs to predict a constant direction in which to move keypoints, and to “slow down” motion in this direction when the keypoint becomes close to the ground truth. See fig. 2 for an illustration.

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 set  $L$  to 20 pixels in our experiments. 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.

**ConvNet architecture.** We employed a standard ConvNet architecture pre-trained on Imagenet: the very deep googlenet [36]<sup>1</sup>. 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 Gaussian 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<sup>2</sup> expressed in Cartesian coordinates (the 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  $224 \times 224$ .

## 3. Results

We tested our method on the two most challenging benchmarks for 2D human pose estimation: the MPII Human Pose dataset [2], which features significant scale variation, occlusion, and multiple people interacting, and Leeds Sports Pose dataset (LSP) [19] which features complex poses of people in sports. For each person in every image, the goal is to predict the 2D locations of all its annotated keypoints.

**MPII – Experimental Details.** Human pose is represented

---

<sup>1</sup>The VGG-16 network [34] produced similar results, but required significantly more memory.

<sup>2</sup>Again, we do not bound explicitly the correction at test time, instead the network is taught to predict bounded corrections.

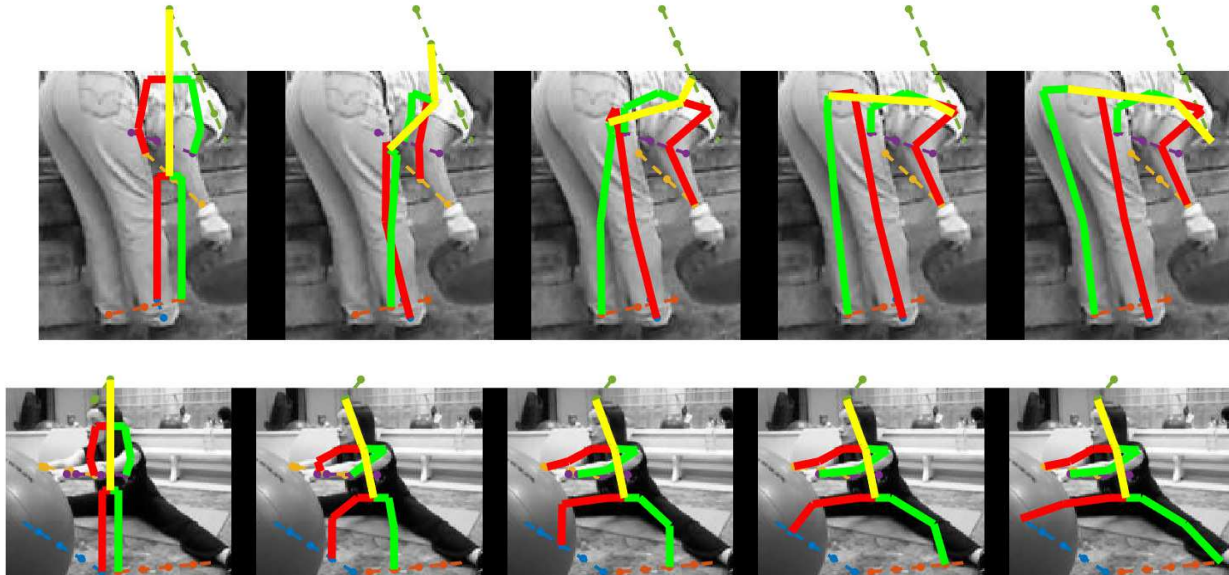


Figure 2: In our human pose estimation running example, the sequence of corrections  $\epsilon_t$  moves keypoints along lines in the image, starting from an initial mean pose  $y_0$  (left), all the way to the ground truth pose  $y$  (right), here shown for two different images. This simplifies prediction at test time, because the desired corrections to each keypoint are constant for each image, up to the last one which is a scaled version. Feedback allows the model to detect when the solution is close and to reduce “keypoint motion”, as in a control system. Linear trajectories are shown for only a subset of the keypoints, to limit clutter.

as a set of 16 keypoints. 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 [37]. 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  $256 \times 256$  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 googlenet models on random crops of  $224 \times 224$  patches, using 6 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 shallower model, the VGG-S ConvNet [3], trained for that task using an  $L_2$  regression loss. We then align our model to the center  $224 \times 224$  patch of the selected window. The MatConvnet library [43] was

employed for these experiments.

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.

**Comparison with State-of-the-Art.** The standard evaluation procedure in the MPII benchmark assumes ground truth scale information is known and images are normalized using this scale information. The current state-of-the-art is the sliding-window approach of Tompson et al [37] and IEF roughly matches this performance, as shown in table 1. In the more realistic setting of unknown scale information, the best previous result so far is from Tompson et al. [37] which was the first work to experiment with this setting and obtained 66.0 PCKh. IEF significantly improves upon this number to 81.3. Note however that the emphasis in Tompson et al’s system was efficiency and they trained and tested their model using original image scales – searching over a multiscale image pyramid or using our automatic rescaling procedure should presumably improve their performance. See the MPII website for more detailed results.

	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	UBody	FBody
Yang & Ramanan [48]	73.2	56.2	41.3	32.1	36.2	33.2	34.5	43.2	44.5
Pischulin et al [29]	74.2	49.0	40.8	34.1	36.5	34.4	35.1	41.3	44.0
Tompson et al. [37]	96.1	91.9	83.9	77.8	80.9	72.3	64.8	84.5	82.0
IEF	95.7	91.6	81.5	72.4	82.7	73.1	66.4	82.0	81.3
Tompson et al. [37]	83.4	77.5	67.5	59.8	64.6	55.6	46.1	68.3	66.0
IEF	95.5	91.6	81.5	72.4	82.7	73.1	66.9	81.9	81.3

Table 1: MPII test set PCKh-0.5 results for Iterative Error Feedback (IEF) and previous approaches, when ground truth scale information at test time is provided (top) and in the more automatic setting when it is not available (bottom). UBody and FBody stand for upper body and full body, respectively.

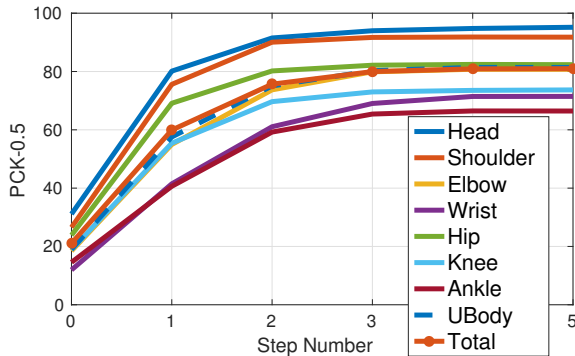


Figure 3: Evolution of PCKh at 0.5 overlap as function of correction step number on the MPII-human-pose validation set, using the finetuned googlenet network. The model aligns more accurately to parts like the head and shoulders, which is natural, because these parts are easier to discriminate from the background and have more consistent appearance than limbs.

**LSP – Experimental Details.** In LSP, differently from MPII, images are usually tight around the person whose pose is being estimated, are resized so people have a fixed size, and have lower resolution. There is also no marking point on the torsos so we initialized the 17th keypoints used in MPII to the center of the image. The same set of keypoints is evaluated as in MPII and we trained a model using the same hyper-parameters on the extended LSP training set. We use the standard LSP evaluation code supplied with the MPII dataset and report person-centric PCP scores in table 2. Our results are competitive with the current state-of-the-art of Chen and Yuille [5].

#### 4. Analyzing IEF

In this section, we perform extensive ablation studies to validate four choices of the IEF model: 1) proceeding iteratively instead of in a single shot, 2) predicting bounded corrections instead of directly predicting the target outputs, 3) curriculum learning of our bounded corrections, and 4) modeling the structure in the full output space (all body

joints in this case) over carrying out independent predictions for each label.

**Iterative v/s Direct Prediction.** For evaluating the importance of progressing towards solutions iteratively we trained models to directly predict corrections to the keypoint locations in a single shot (i.e. direct prediction). Table 3 shows that IEF that additively regresses to keypoint locations achieves PCKh-0.5 of 81.0 as compared to PCKh of 74.8 achieved by directly regressing to the keypoints.

**Iterative Error Feedback v/s Iterative Direct Prediction.** Is iterative prediction of the error important or iterative prediction of the target label directly (as in e.g., [45, 41]) performs comparably? In order to answer this question we trained a model from the pretrained googlenet to iteratively predict the ground truth keypoint locations (as opposed to predicting bounded corrections). For comparing performance, we used the same number of iterations for this baseline model and IEF. Table 3 shows that IEF achieves PCKh-0.5 of 81.0 as compared to PCKh of 73.4 by iterative direct prediction. This can be understood by the fact that the learning problem in IEF is much easier. In IEF, for a given image, the model is trained to predict constant corrections except for the last one which is a scaled version. In iterative direct prediction, because each new pose estimate ends up somewhere around the ground truth, the model must learn to adjust directions and magnitudes in all correction steps.

**Importance of Fixed Path Consolidation (FPC).** The FPC method (see algorithm 1) for training a IEF model makes  $N$  corrections is a curriculum learning strategy where in the  $i^{th}$  ( $i \leq N$ ) training stage the model is optimized for performing only the first  $i$  corrections. Is this curriculum learning strategy necessary or can all the corrections be simultaneously trained? For addressing this question we trained an alternative model that trains for all corrections in all epochs. We trained IEF with and without FPC for the same number of SGD iterations and the performance of both these models is illustrated in figure 4. The figure shows that without FPC, the performance drops by almost 10 PCKh points on the validation set and that there is significant drift when performing several correction steps.

	Torso	Upper Leg	Lower Leg	Upper Arm	Forearm	Head	Total
Pishchulin et al. [30]	88.9	64.0	58.1	45.5	35.1	85.1	58.0
Tompson et al. [38]	90.3	70.4	61.1	63.0	51.2	83.7	66.6
Fan et al. [9]	95.4	77.7	69.8	62.8	49.1	86.6	70.1
Chen and Yuille [5]	96.0	77.2	72.2	69.7	58.1	85.6	73.6
IEF	95.3	81.8	73.3	66.7	51.0	84.4	72.5

Table 2: Person-centric PCP scores on the LSP dataset test set for IEF and previous approaches.

	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	UBody	FBody
Iterative Error Feedback (IEF)	95.2	91.8	80.8	71.5	82.3	73.7	66.4	81.4	81.0
Direct Prediction	92.9	89.4	74.1	61.7	79.3	64.0	53.3	75.1	74.8
Iterative Direct Prediction	91.9	88.5	73.3	59.9	77.5	61.2	51.8	74.0	73.4

Table 3: PCKh-0.5 results on the MPII validation set for models finetuned from googlenet using Iterative Error Feedback (IEF), direct regression to the keypoint locations (direct prediction), and a model that was trained to iteratively predict human pose by regressing to the ground truth keypoint locations (instead of bounded corrections) in each iteration, starting from the pose in the previous iteration. The results show that our proposed approach results in significantly better performance.

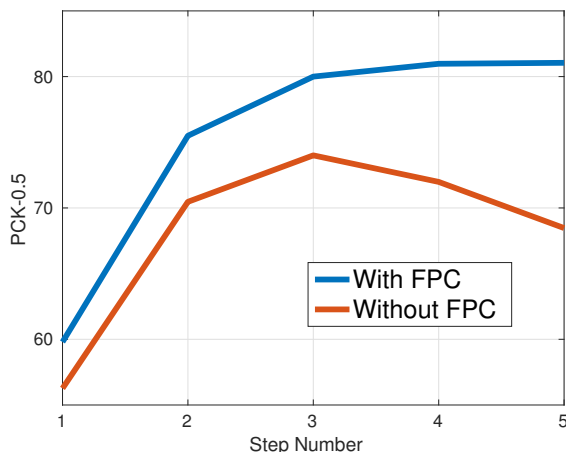


Figure 4: Validation PCKh-0.5 scores for different number of correction steps taken, when finetuning a IEF model from a googlenet base model using stochastic gradient descent with either Fixed Path Consolidation (*With FPC*), or directly over all training examples (*Without FPC*), for the same amount of time. FPC leads to significantly more accurate results, leading to models that can perform more correction steps without drifting. It achieves this by consolidating the learning of earlier steps and progressively increasing the difficulty of the training set by adding additional correction steps.

**Learning Structured Outputs.** One of the major merits of IEF is supposedly that it can jointly learn the structure in input images and target outputs. For human pose estimation, IEF models the space of outputs by augmenting the image with additional input channels having gaussian renderings centered around estimated keypoint locations. If it is the case that IEF learns priors over the appropriate relative lo-

cations of the various keypoints, then depriving the model of keypoints other than the one being predicted should decrease performance.

In order to evaluate this hypothesis we trained three different IEF models and tested how well each predicted the location of the “Left Knee” keypoint. The first model had only one input channel corresponding to the left knee, the second model had two channels corresponding to left knee and the left hip. The third model was trained using all keypoints in the standard IEF way. The performance of these three models is reported in table 4. As a baseline, regression gets 64.6, whereas the IEF model with a single additional input channel for the left knee gets PCKh of 69.2. This shows that feeding back the current estimate of the left knee keypoint allows for more accurate localization by itself. Furthermore, the IEF model over both left knee and left hip gets PCKh of 72.8. This suggests that the relationship between neighboring outputs has much of the information, but modeling all joints together with the image still wins, obtaining a PCKh of 73.8.

## 5. Related Work

There is a rich literature on structured output learning [40, 7] (e.g. see references in [26]) but it is a relatively modern topic in conjunction with feature learning, for computer vision [4, 18, 37, 23].

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 [14], figure-ground segregation [16] and object recognition [46]. In computer vision, feedback has been primarily used so far for learning selective attention [25]; in [25] attention is implemented by estimating a bounding box in an image

	Direct Prediction of All Joints	IEF Left Knee	IEF Left Knee + Left Hip	IEF All Joints
Left Knee PCKh-0.5	64.6	69.2	72.8	73.8

Table 4: MPII validation PCKh-0.5 results for left knee localization when using IEF and both training and predicting different subsets of joints. We also show the result obtained using a direct prediction variant similar to plain regression on all joints (having the mean pose Gaussian maps in the input). Modeling global body structure jointly with the image leads to best results by "IEF All Joints". Interestingly, feedback seems to add value by itself and IEF on the left knee, in isolation, significantly outperforms the direct prediction baseline.

for the algorithm to process next, while in [35] attention is formed by selecting some convolutional features over others (it does not have a spatial dimension).

Stacked inference methods [31, 44, 45, 41] are another related family of methods. Differently, some of these methods consider each output in isolation [39], all use different weights or learning models in each stage of inference [37] or they do not optimize for correcting their current estimates but rather attempt to predict the answer from scratch at each stage [24, 41]. In concurrent work, Oberweger et al [27] proposed a feedback loop for hand pose estimation from kinect data that is closely related to our approach. The autocontext work of [41] is also related and iteratively computes label heatmaps by concatenating the image with the heatmaps previously predicted. IEF is inspired by this work and we show how this iterative computation can be carried out effectively with deep Convnet architectures, and with bounded error corrections, rather than aiming for the answer from scratch at each iteration.

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 [15]. For pose inference, combining multiple scales [10, 37] 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" [12, 8, 17], but leveraging hierarchical feature learning. Graphical models can also encode dependencies between outputs and are still popular in many applications, including human pose estimation [5].

Classic spatial alignment and warping computer vision models, such as snakes, [20] and Active Appearance Models (AAMs) [6] 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 [47],

typically these methods still require manually defined energy functions to measure goodness of fit.

## 6. 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 standard back-propagation. This is desirable in order to attack problems with higher-dimensional output spaces such as 3D human pose estimation [32, 33] 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.

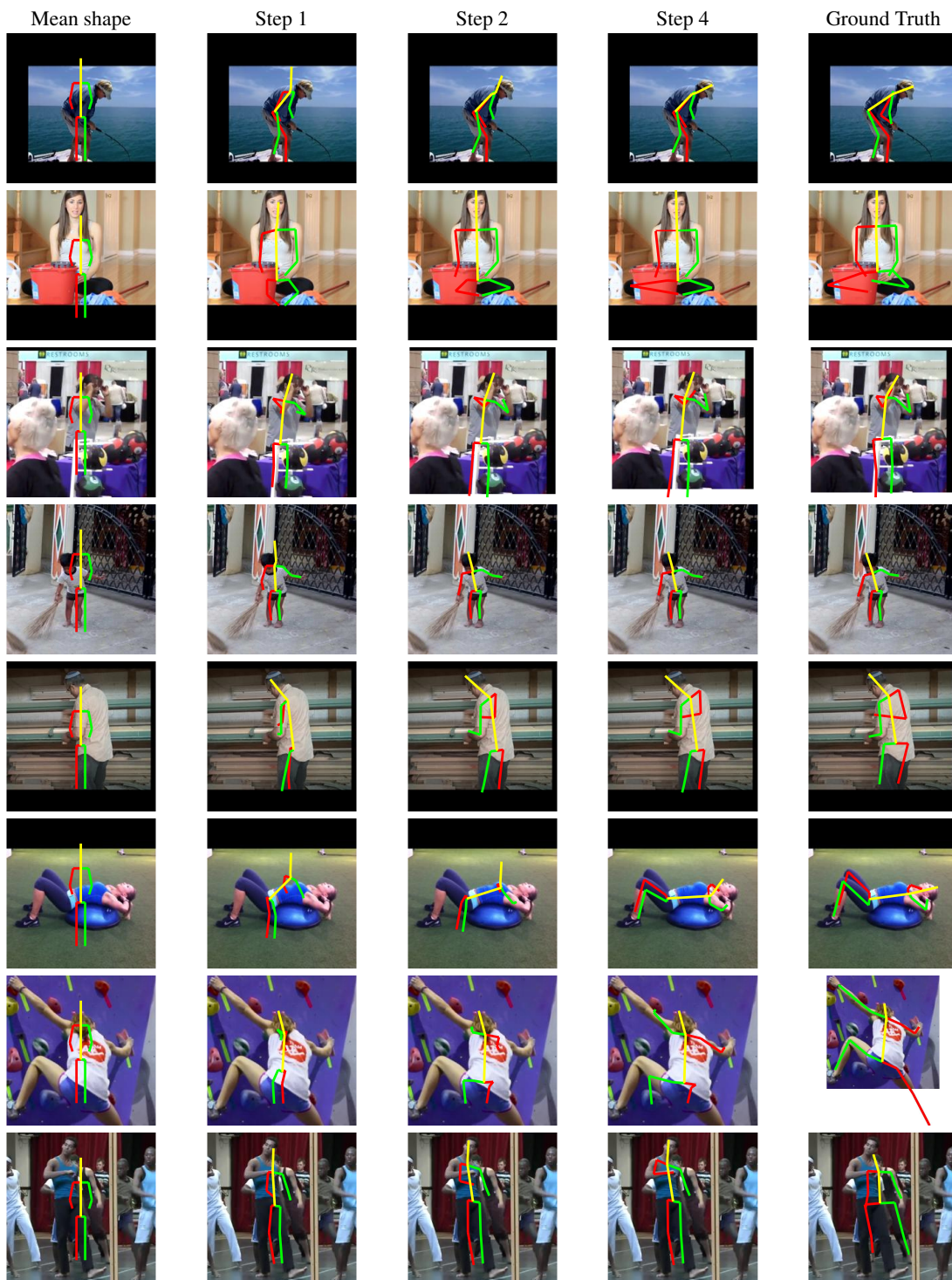


Figure 5: Example poses obtained using the proposed method IEF on the MPII validation set. From left to right we show the sequence of corrections the method makes – on the right is the ground truth pose, including annotated occluded keypoints, which are not evaluated. Note that IEF is robust to left-right ambiguities and is able to rotate the initial pose by up to 180 (first and fifth row), can align across occlusions (second and third rows) and can handle scale variation (second, fourth and fifth rows) and truncation (fifth row). 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 model aligns to the wrong person. The black borders show padding. Best seen in color and with zoom.



## References

- [1] P. Agrawal, D. Stansbury, J. Malik, and J. L. Gallant. Pixels to voxels: Modeling visual representation in the human brain. *arXiv preprint arXiv:1407.5104*, 2014.
- [2] M. Andriluka, L. Pishchulin, P. Gehler, and B. 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] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *British Machine Vision Conference*, 2014.
- [4] L.-C. Chen, A. G. Schwing, A. L. Yuille, and R. Urtasun. Learning deep structured models. *arXiv preprint arXiv:1407.2538*, 2014.
- [5] X. Chen and A. 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.
- [6] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models-their training and application. *Computer vision and image understanding*, 61(1):38–59, 1995.
- [7] H. Daumé III, J. Langford, and D. Marcu. Search-based structured prediction. *Machine learning*, 75(3):297–325, 2009.
- [8] P. Dollár, P. Welinder, and P. Perona. Cascaded pose regression. In *CVPR*, 2010.
- [9] X. Fan, K. Zheng, Y. Lin, and S. Wang. Combining local appearance and holistic view: Dual-source deep neural networks for human pose estimation. *arXiv preprint arXiv:1504.07159*, 2015.
- [10] X. Fan, K. Zheng, Y. Lin, and S. Wang. Combining local appearance and holistic view: Dual-source deep neural networks for human pose estimation. June 2015.
- [11] D. J. Felleman and D. C. Van Essen. Distributed Hierarchical Processing in the Primate Cerebral Cortex. *Cerebral Cortex*, 1(1):1–47, Jan. 1991.
- [12] F. Fleuret and D. Geman. Stationary features and cat detection. Technical Report Idiap-RR-56-2007, 0 2007.
- [13] K. 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.
- [14] C. D. Gilbert and M. Sigman. Brain states: Top-down influences in sensory processing. *Neuron*, 54(5):677 – 696, 2007.
- [15] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. *arXiv preprint arXiv:1411.5752*, 2014.
- [16] 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, Aug. 1998.
- [17] C. Ionescu, J. Carreira, and C. 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.
- [18] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Deep structured output learning for unconstrained text recognition. *ICLR 2015*, 2014.
- [19] S. Johnson and M. 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.
- [20] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988.
- [21] B. C. U. L. M. M. Kravitz DJ, Saleem KS. The ventral visual pathway: An expanded neural framework for the processing of object quality. volume 17(1), 2013.
- [22] V. A. F. Lamme and P. R. Roelfaema. the distinct modes of vision offered by feedforward and recurrent processing. *Trends in Neurosciences*, 23:571, 2000.
- [23] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [24] Q. Li, J. Wang, Z. Tu, and D. P. Wipf. Fixed-point model for structured labeling. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 214–221, 2013.
- [25] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu. Recurrent models of visual attention. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2204–2212. Curran Associates, Inc., 2014.
- [26] S. Nowozin and C. H. Lampert. Structured learning and prediction in computer vision. *Foundations and Trends® in Computer Graphics and Vision*, 6(3–4):185–365, 2011.
- [27] M. Oberweger, P. Wohlhart, and V. Lepetit. Training a feedback loop for hand pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3316–3324, 2015.
- [28] T. Pfister, J. Charles, and A. Zisserman. Flowing convnets for human pose estimation in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1913–1921, 2015.
- [29] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele. Poselet conditioned pictorial structures. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 588–595. IEEE, 2013.
- [30] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele. Strong appearance and expressive spatial models for human pose estimation. In *International Conference on Computer Vision (ICCV), 2013 IEEE Conference on*, pages 3487–3494. IEEE, 2013.
- [31] V. Ramakrishna, D. Munoz, M. Hebert, J. A. Bagnell, and Y. Sheikh. Pose machines: Articulated pose estimation via inference machines. In *Computer Vision–ECCV 2014*, pages 33–47. Springer International Publishing, 2014.
- [32] G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter-sensitive hashing. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 750–757. IEEE, 2003.

- [33] L. Sigal, M. Isard, H. Haussecker, and M. J. Black. Loose-limbed people: Estimating 3D human pose and motion using non-parametric belief propagation. *International Journal of Computer Vision*, 98(1):15–48, May 2011.
- [34] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [35] M. F. Stollenga, J. Masci, F. Gomez, and J. Schmidhuber. Deep networks with internal selective attention through feedback connections. In *Advances in Neural Information Processing Systems*, pages 3545–3553, 2014.
- [36] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. June 2015.
- [37] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler. Efficient object localization using convolutional networks. June 2015.
- [38] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 1799–1807. 2014.
- [39] A. Toshev and C. 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.
- [40] I. Tsochantaris, T. Hofmann, T. Joachims, and Y. 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.
- [41] Z. 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.
- [42] E. Tulving and D. L. Schacter. Priming and human memory systems. *Science*, 247(4940):301–306, 1990.
- [43] A. Vedaldi and K. Lenc. Matconvnet-convolutional neural networks for matlab. *arXiv preprint arXiv:1412.4564*, 2014.
- [44] D. Weiss, B. Sapp, and B. Taskar. Structured prediction cascades. *arXiv preprint arXiv:1208.3279*, 2012.
- [45] D. H. Wolpert. Stacked generalization. *Neural networks*, 5(2):241–259, 1992.
- [46] D. Wyatte, T. Curran, and R. C. O’Reilly. The limits of feed-forward vision: Recurrent processing promotes robust object recognition when objects are degraded. *J. Cognitive Neuroscience*, pages 2248–2261, 2012.
- [47] X. Xiong and F. De la Torre. Supervised descent method and its applications to face alignment. In *CVPR*.
- [48] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, 2011.