A Coarse-Fine Network for Keypoint Localization

Shaoli Huang[1] Mingming Gong[2] Dacheng Tao[1]
[1]UBTECH Sydney AI Centre, SIT, FEIT, The University of Sydney
[2]CAI, FEIT, University of Technology Sydney
shaol.huang@sydney.edu.au gongmingnju@gmail.com dacheng.tao@sydney.edu.au

Abstract

We propose a coarse-fine network (CFN) that exploits multi-level supervisions for keypoint localization. Recently, convolutional neural networks (CNNs)-based methods have achieved great success due to the powerful hierarchical features in CNNs. These methods typically use confidence maps generated from ground-truth keypoint locations as supervisory signals. However, while some keypoints can be easily located with high accuracy, many of them are hard to localize due to appearance ambiguity. Thus, using strict supervision often fails to detect keypoints that are difficult to locate accurately. To target this problem, we develop a keypoint localization network composed of several coarse detector branches, each of which is built on top of a feature layer in a CNN, and a fine detector branch built on top of multiple feature layers. We supervise each branch by a specified label map to explicate a certain supervision strictness level. All the branches are unified principally to produce the final accurate keypoint locations. We demonstrate the efficacy, efficiency, and generality of our method on several benchmarks for multiple tasks including bird part localization and human body pose estimation. Especially, our method achieves 72.2% *AP on the 2016 COCO Keypoints Challenge dataset, which is an* 18% *improvement over the winning entry.*

1. Introduction

Predicting a set of semantic keypoints, such as human body joints or bird parts, is an essential component of understanding objects in images. For example, keypoints help align objects and reveal their subtle differences that are useful for handling the problems with small inter-class variations such as fine-grained categorization [51, 48, 17].

Despite dramatic progress over recent years, keypoint prediction remains a significant challenge due to appearance variations, pose changes, and occlusions. For instance, the local appearances of bird parts may differ vastly across species or different poses (*e.g.* perching, flying, and walking). Localizing keypoints on the human body must be invariant to appearance changes caused by factors like clothing and lighting, and robust to large layout changes of parts due to articulations [40]. To tackle these difficulties, early works combined handcrafted part appearance features and with an associated spatial model to capture both local and global information [24, 33, 31, 46].

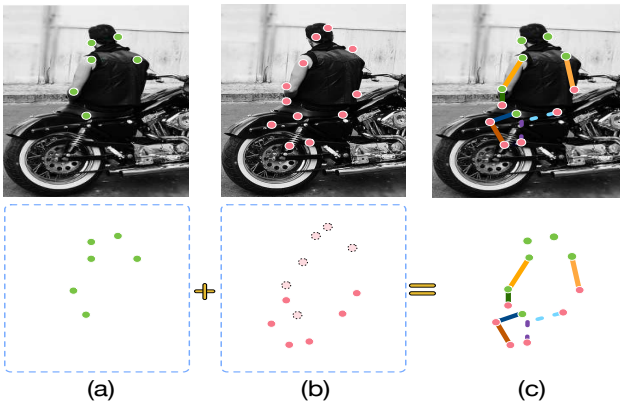

Figure 1: An illustration of the predicted keypoints from our CFN architecture. The left image contains highly accurate keypoints detected by the fine detector with strict supervision, the middle image contains keypoints from coarse detectors with loose supervisions, and the right image shows the final predictions by unifying the fine and coarse detectors.

Recently, convolutional neural networks (CNNs) [38, 36, 16] have significantly reshaped the conventional pipeline by replacing handcrafted features and explicit spatial models with more powerful learned hierarchical representations [40, 35, 43, 27]. The hierarchical representations in CNNs provide us a natural way to implicitly model part appearances and spatial interactions between parts. Thus, considerable effort has been placed into leveraging hierarchical features in CNNs to build a fine keypoint detector which is expected to achieve high localization accuracy [29, 2].

Existing CNN-based keypoint localization methods usually supervise keypoint detectors using confidence maps generated from ground-truth keypoint locations. However, while some keypoints can be easily located with high accuracy, many of them are hard to localize due to appearance ambiguity. For example, the keypoints with distinctive appearances, such as the shoulders and head, can be easily detected with high accuracy, while the keypoints with ambiguous appearance due to body occlusion or low resolution images, have much lower localization accuracies. Thus, the keypoint detector often fails to detect ambiguous keypoints if trained with strict supervision, that is, permitting only a small localization error. Training with looser supervision could help detect the ambiguous or indistinct keypoints, but this comes at a cost to localization accuracy for those keypoints with distinctive appearances.

To address the robustness problem of keypoint localization, we propose a coarse-fine network (CFN) that imposes multi-level supervisions within a deep convolutional neural network (CNN) for keypoint localization. To achieve this, we first propose a fully convolutional Inception network [38] with several branches of varying depths to obtain hierarchical feature representations. Then, we build a coarse part detector on top of each branch of features and a fine part detector which takes features from all the branches as the input.

The constructed detectors have different localization abilities and are complementary to each other. The shallower coarse detectors can produce accurate localizations of keypoints with distinctive appearances; however, they often fail to detect keypoints with ambiguous appearances. The deeper branches can infer the approximate locations of ambiguous keypoints but at the cost of reduced localization accuracy for the unambiguous keypoints. Thus, we supervise these branches of detectors using multi-level label maps with strictness levels that are set according to the localization abilities of these branches. By supervising the part detectors built on hierarchical features with multi-level supervisor signals, our CFN fully explores the diversities of part structures and the diversities of representations in CNNs.

Finally, the keypoints produced by each CFN branch are unified to produce the final keypoint locations. As shown in Figure 1, the finally detected keypoints include very accurate ones detected by the fine detector and approximately accurate ones detected by the coarse detectors. The proposed CFN outperforms state-of-the-art approaches by a large margin on bird part localization and human pose estimation datasets. Especially, our method is particularly effective for low resolution persons, while the existing methods perform much worse.

2. Related Works

Bird part detection Bird parts play a remarkable role in fine-grained categorization, especially in bird species recognition where parts have subtle differences. Early works focused on developing handcrafted part appearance features (*e.g.*, HOG [7]) and spatial location models (*e.g.* pictorial models [11]) to capture local and global information, respectively. For example, the deformable part model (DPM) [10] has been extended for bird part localization by incorporating strong supervision or segmentation masks [50, 4]. Liu *et al*. [23, 24] presented a nonparametric model called exemplar to impose geometric constraints on the part configuration. Another line of works utilize unlabeled data and domain adaptation techniques [12, 13, 25] to boost the localization accuracy for bird parts [28, 44].

More recently, convolutional neural networks (CNNs) based methods have been widely used in this task. Inspired by object proposals in object detection, part-based R-CNN [49] extracts CNN features from bottom-up proposals and learns whole-object and part detectors with geometric constraints. Following this strategy, EdgeBox [52] and K-nearest neighbors proposals [48] have been used to improve the quality of part proposals. These methods significantly outperform conventional approaches; however, the proposal generation and feature extraction are computationally expensive. Our approach avoids proposal generation by adopting the fully convolutional architecture which was originally proposed for dense prediction tasks like semantic segmentation [26].

Human pose estimation Classical approaches to articulated pose estimation adopt graphical models to explicitly model the correlations and dependencies of the body part locations [1, 46, 39, 30, 21, 8]. These models can be classified into tree-structured [1, 37, 39, 31], and non-tree-structured [21, 8] models. Attempts have also been made to model complex spatial relationships implicitly based on a sequential prediction framework which learns the inference procedure directly [33, 31].

Again, the advent of deep CNNs have recently contributed to significant improvements in feature representation and have significantly improved human pose estimation [43, 27, 3, 32, 41, 40, 45, 5]. Toshev *et al*. [41] directly regressed x, y joint coordinates with a convolutional network, while more recent work regressed images to confidence maps generated from joint locations [43, 27, 3, 40]. Tompson *et al*. [40] jointly trained a CNN and a graphical model, incorporating long-range spatial relations to remove outliers on the regressed confidence maps. Papandreou *et al*. [29] proposed to use fully convolutional ResNets[16] to predict a confidence map and an offset map simultaneously and aggregated them to obtain accurate predictions. Other works adopted a sequential procedure that refined the predicted confidence maps successively using a series of con-

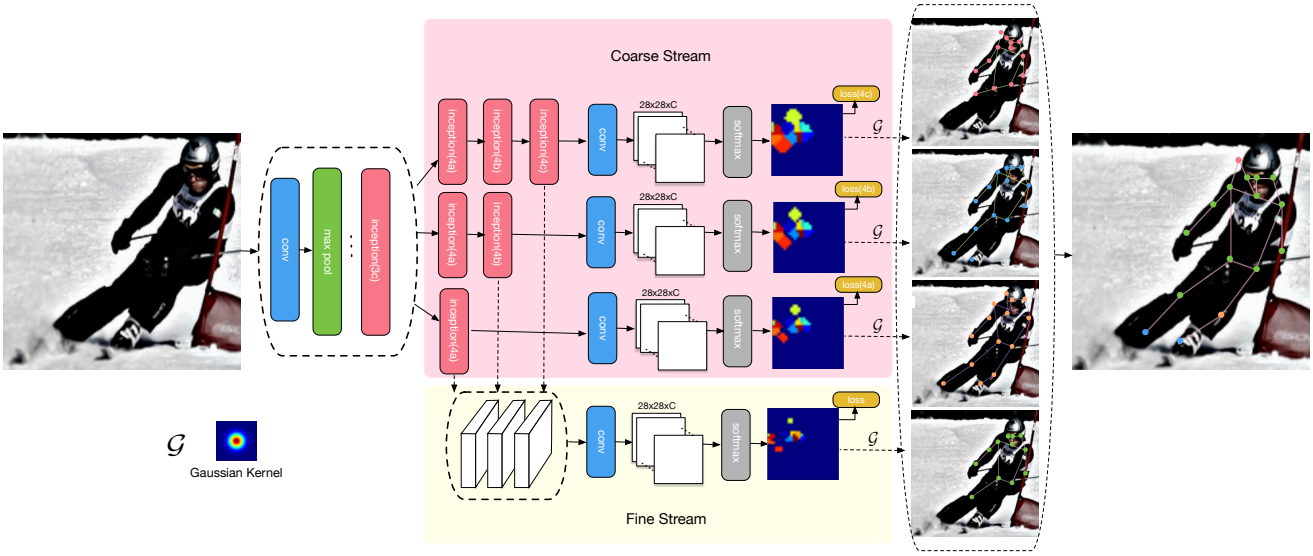

Figure 2: The network architecture of CFN. The coarse stream learns three coarse keypoint detectors using multi-level supervisions and while the fine stream learns a fine detector via strict supervision. Then the coarse predictions and fine predictions are unified for final prediction at the inference stage.

volutional modules [43, 27, 3]. Cao *et al*. [2] proposed a pose estimation framework which adopts both explicit spatial modeling and implicit sequential predictions. In contrast to existing approaches, our approach models the part appearance and spatial relationships using a single network with several branches to capture multi-scale information, which is more efficient because it requires no explicit graphical model-style inference or sequential refinement. Also, we generate label maps used for supervision according to the localization capability of each branch.

3. Coarse-Fine Network

In this section, we introduce the CFN architecture and describe the details of each component. As illustrated in Figure 2, the proposed framework consists of shared base convolutional layers and two streams of keypoint detectors. The coarse stream consists of three coarse detector branches, each of which only inputs features within a specific scale range induced by the Inception modules. The main difference in these branches is the number of stacked inception modules, leading to different receptive field sizes. Smaller receptive fields focus more on capturing local appearances, while larger ones are more suitable for modeling the spatial dependencies between parts. Therefore we concatenate feature maps from all the coarse detectors to learn a fine detector that is expected to provide very accurate localizations. Finally, we learn the entire network using multi-level label maps, each of them has a strictness level varying with the localization ability of the corresponding detector.

3.1. Network Architecture

The proposed CFN simultaneously predicts multiple keypoint/part locations from the input image. Our method is inspired by the "recognition using regions" paradigm [14], which has been widely used in general object detection [34]. However, different from object detection, no bounding-boxes are provided for supervision in the keypoint localization task. Instead, we predefine a set of square boxes around the ground-truth keypoint locations as "virtual" bounding-boxes of these keypoints.

Stride, receptive fields, and depth. We build the detector based on Inception-v2 [38], a deep convolutional network architecture that has achieved impressive performance in object recognition. In a convolutional network, the stride and receptive field sizes increase with depth. Thus, deeper layers encode richer contextual information to disambiguate different parts at the cost of reduced localization accuracy. To balance part classification and localization accuracy, we employ the features in the Inception (4a-4c) layers to train the three coarse detectors. The stride of the Inception (4a-4c) layers is 16, and the corresponding receptive field sizes are 107×107, 139×139, and 171×171, respectively. Given an input image of size 224x224, the receptive-field size in deeper layers is too large for a part and may lead to ambiguous detections for closely positioned parts. Thus we increase the input resolution of the network to 448×448 so that the receptive field sizes are appropriate to enclose candidate part regions.

Candidate part regions. After obtaining the final incep-

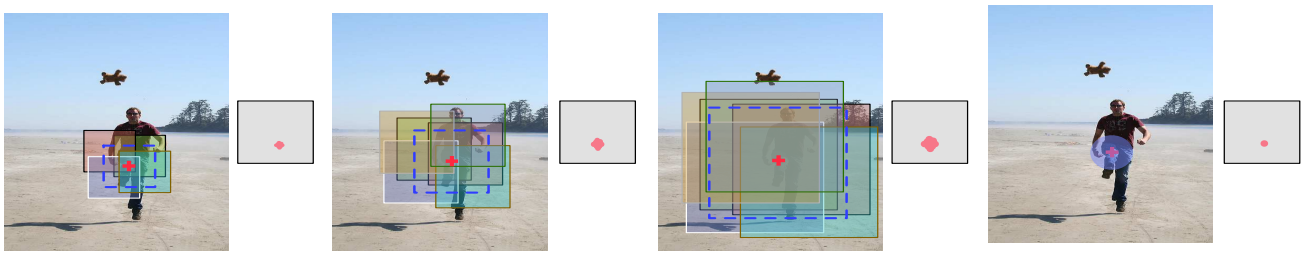

Figure 3: An illustration of how the multi-level supervisions are generated for the coarse and fine detectors. For each coarse detector, we generate a bounding-box with the same size as the receptive fields in that branch as ground-truth, which is shown as a dashed blue box around the keypoint. The supervision map is calculated according to overlap between the generated ground-truth box and the regions induced by receptive fields in the final feature map. For the fine detector, we use a small circle to enforce strict supervisions. (Note that the candidate regions are equally spaced, the unequally spaced candidate regions shown in this figure are for better illustrations.)

tion layers (4a-4c), we can predict the supervisory label map by 1×1 convolutions, which is equivalent to a sliding-window search on the image grid with a large stride size. However, it is unclear how the supervisory label should be generated. One typical way is to place a small circle around the ground-truth keypoint locations, but it is hard to determine the radius of this circle because either a large or small radius will lead to inferior performance. To introduce our multi-level supervision, we brought ideas from object detection that uses overlap between candidate part regions and ground-truth bounding-boxes. In object detection, the candidate object regions are obtained by generating region proposals of various sizes and aspect ratios. However, keypoint localization only aims to infer the central location of parts and thus does not require a bounding-box that bounds the parts tightly. Thus, we define the part regions as squared regions enclosed by the corresponding receptive fields centered at the ground truth locations. For example, the size of the Inception (4a) feature map is 28×28, which means that there are 784 candidate regions of size 107×107, which are uniformly spaced on the input image.

Feature representation. Using regions enclosed by receptive fields as candidate part regions simplifies the feature extraction for part detectors. In the proposed CFN, the cross-channel vector at a spatial position in the feature map is used as a feature for the candidate part region associated with that position. Also, the fine detector relies on multi-scale representations by fusing multiple feature layers each of which is processed by multiple filter sizes through Inception modules. Therefore, the fine detector in our network can model the appearance of the object parts by features from a large number of scales.

Multi-level supervisions. To fully explore the diversities of hierarchical representations in CNNs, we simultaneously learn all detectors using multi-level supervisions. Each detector has its own appropriate supervision generated according to receptive field size. Specifically, we generate label maps for a detector by calculating the intersections between the candidate part regions and the "virtual" ground truth part regions. Let $K_c = \{1, \ldots, K\}$ be the set of part classes, and D denote the number of coarse detector branches. Given an output feature map in the d-th branch with size $W \times H$, stride s, offset padding p, and receptive field size r, each location (w, h) in the output feature map corresponds to a receptive field $rf(w, h)$ centered at position $(w^*, h^*) = (w, h) * s - (p - 1) + r/2$ in the input image. For an annotated keypoint location (i, j) with class $k \in K_c$, we define a ground truth region $gt^k(i, j)$ with size $r \times r$ centered at (i, j). To construct a target response map Y^d for the d-th detector branch, we set $Y^d(w, h) = k$ if the candidate region $rf(w, h)$ has an Intersection-over-Union (IoU) higher than 0.5 with the "virtual" ground truth region $gt^k(i, j)$ and set $Y^d(w, h) = 0$ to classify it as the background otherwise. For the fine detector, we generate a strict supervision map by setting $Y^f(w, h) = k$ if $\| (w^*, h^*) - (i_k, j_k) \|_2 \leq \lambda * ref\_lengh$ and set $Y^f(w, h) = 0$ otherwise, where λ is a control threshold of strictness and ref_lengh is the longer side of the object bounding box. The multi-level label maps generated for the detector branches enable detection of keypoints at various localization accuracy levels.

3.2. Learning and Inference

We build diversified part detectors using fully convolutional architectures with different depths and supervisions. For efficient inference, we simultaneously learn all the detection networks with shared base convolutional layers by minimizing a multi-task loss.

Learning. Let $\sigma^d = \varphi(X, W, \Phi^d, \Phi^d_{cls})$ be the last feature maps of size $W \times H \times C$ in the d-th detector branch given input image X, shared weights W, unshared weights Φ^d in the feature layers, and unshared weights Φ^d_{cls} in the classi-

Table 1: Comparison with methods that report per-part PCK(%) and average PCK(%) on CUB200-2011. The abbreviated part names from left to right are: Back, Beak, Belly, Breast, Crown, Forehead, Left Eye, Left Leg, Left Wing, Nape, Right Eye, Right Leg, Right Wing, Tail, and Throat.

α	Methods	Ba	Bk	Be	Br	Cr	Fh	Le	Ll	Lw	Na	Re	Rl	Rw	Ta	Th	Mean
0.1	[51]	85.6	**94.9**	81.9	84.5	**94.8**	**96.0**	**95.7**	64.6	67.8	90.7	93.8	64.9	69.3	74.7	94.5	83.6
	CFN	88.3	94.5	87.3	91.0	93.0	92.7	93.7	**76.9**	**80.5**	**93.2**	**94.0**	**81.2**	**79.2**	**79.7**	**95.1**	**88.0**
0.05	[51]	46.8	62.5	40.7	45.1	59.8	63.7	66.3	33.7	31.7	54.3	63.8	36.2	33.3	39.6	56.9	49.0
	[47]	**66.4**	49.2	56.4	60.4	61.0	60.0	66.9	32.3	35.8	53.1	66.3	35.0	37.1	40.9	65.9	52.4
	CFN	64.1	**87.9**	**57.9**	**65.8**	**80.9**	**83.9**	**90.3**	**58.0**	**50.9**	**79.4**	**89.6**	**62.6**	**51.0**	**57.9**	**84.9**	**70.9**
0.02	[51]	9.4	12.7	8.2	9.8	12.2	13.2	11.3	7.8	6.7	11.5	12.5	7.3	6.2	8.2	11.8	9.9
	[47]	18.6	11.5	13.4	14.8	15.3	14.1	20.2	6.4	8.5	12.3	18.4	7.2	8.5	8.6	17.9	13.0
	CFN	**19.6**	**40.7**	**15.7**	**19.0**	**33.1**	**36.0**	**47.8**	**20.1**	**13.1**	**28.9**	**47.1**	**20.9**	**14.4**	**18.3**	**34.1**	**27.3**

Table 2: Comparison of PCP(%) and over-all PCP(%) on CUB200-2011. The abbreviated part names from left to right are: Back, Beak, Belly, Breast, Crown, Forehead, Eye, Leg, Wing, Nape, Tail, and Throat.

Methods	Ba	Bk	Be	Br	Cr	Fh	Ey	Le	Wi	Na	Ta	Th	Total
[23]	62.1	49.0	69.0	67.0	72.9	58.5	55.7	40.7	71.6	70.8	40.2	70.8	59.7
[24]	64.5	**61.2**	71.7	70.5	76.8	**72.0**	**70.0**	45.0	74.4	79.3	46.2	**80.0**	66.7
[35]	74.9	51.8	**81.8**	77.8	**77.7**	67.5	61.3	52.9	**81.3**	76.1	59.2	78.7	69.1
CFN	**82.2**	57.4	81.3	**80.3**	75.6	63.0	62.5	**70.8**	70.8	**81.1**	**59.7**	73.5	**72.1**

fier layer, respectively. We add one more channels to model the background class and thereby $C = (|K_c|+1)$. We use the multi-level label maps described in Figure 3 as supervisions. Here, we compute the prediction score $Pro^d_{(w,h,k)}$ at the position (w, h, k) in the last feature maps using the softmax function.

Therefore, the loss function on a training image for each branch is defined as bellow:

$$\ell(X, W, \Phi^d, \Phi^d_{cls}, Y^d) = \frac{-1}{W \times H} \sum_{w=0}^{W-1} \sum_{h=0}^{H-1} \sum_{k=0}^{|K_c|} 1\{Y^d_{(w,h)} = k\} log(Pro^d_{(w,h,k)}). \tag{1}$$

The loss function $\ell(X, W, \Phi^f, \Phi^f_{cls}, Y^f)$ for the fine detector is defined similarly as Eqn. 1. Then we use a multi-task loss to train all the coarse detectors and the fine detector jointly:

$$\pounds(\Omega, Y) = \sum_{d=1}^{D} \ell(X, W, \Phi^d, \Phi^d_{cls}, Y^d) + \ell(X, W, \Phi^f, \Phi^f_{cls}, Y^f), \tag{2}$$

where $\Omega = \{W, \{\Phi^d, \Phi^d_{cls}\}^D_{d=1}, \Phi^f_{cls}\}$, $\Phi^f = \{\Phi^d\}^D_{d=1}$, and $Y = \{\{Y^d\}^D_{d=1}, Y^f\}$.

Inference. For each detector in the inference stage, we first obtain the prediction scores for all candidate regions and then compute the prediction map $\mathcal{O}^d$ for each part as follows:

$$\mathcal{O}^d(w, h, k^*) = \begin{cases} 1 & \text{if } \underset{k}{\text{argmax}}\, Pro^d_{(w,h,k)} = k^* \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

As we use loose supervision for each detector, the results $\mathcal{O}^d$ have multiple predicted locations for each part. According to the overlapping receptive field mechanisms in CNNs, the most precise prediction is around the center of the predicted locations. Therefore, we obtain a "blur" prediction by convolving the prediction maps with a 2D Gaussian kernel $\mathcal{G}$ and select the location with the maximum value in the k-th channel as the unique prediction (w^*_k, h^*_k) for the k-th part.

Unified detection. Our system learns four detectors simultaneously and unifies their outputs into the final prediction. The detectors vary in their ability to detect the object parts. The fine detector tends to output accurate and reliable predictions since it receives stacked features from multiple layers. However, we observe that it may miss predictions of some occluded parts, which can be detected by the coarse detectors. To predict a set of parts as precisely and as completely as possible, we combine the outputs from the coarse and fine detectors by using the strategy that the former ones serve as the assistant predictors for the latter one. Let $(w^*_k, h^*_k)^d$ be the k^{th} part prediction with score $Pro^d_{(w^*,h^*,k)}$ from the d-th coarse part de-

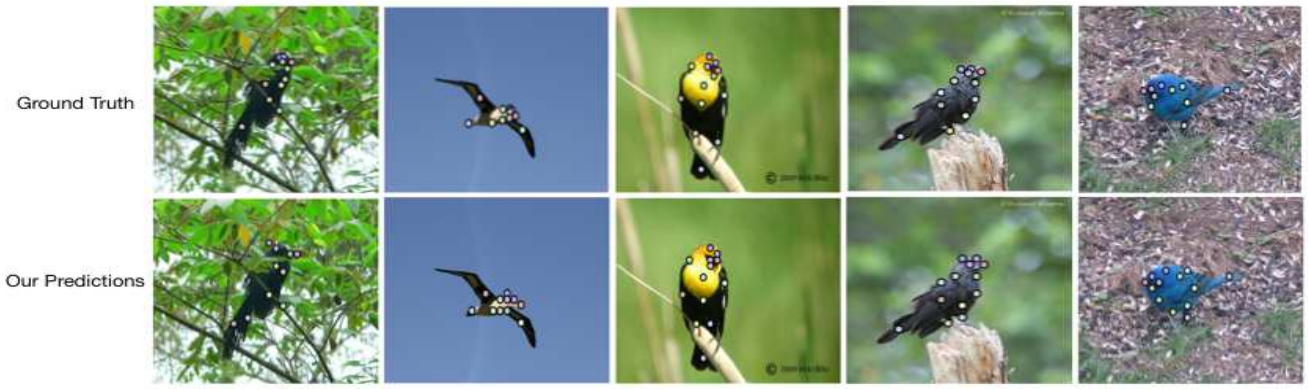

Figure 4: Bird part detection results with occlusion,viewpoint, clustered background, and pose from the test set.

Table 3: Performance comparison between using strict supervision only and using multi-level supervisions.

α	Methods	4a(%)	4b(%)	4c(%)	Fine(%)	Unified(%)
0.1	Str-super	66.1	59.6	79.9	80.8	83.7
	Multi-super	79.2	84.9	82.0	80.8	**88.0**
0.05	Str-super	55.6	49.1	66.6	67.4	69.3
	Multi-super	60.6	59.8	52.4	67.6	**71.0**
0.02	Str-super	22.5	18.8	26.5	26.8	27.3
	Multi-super	20.9	18.3	14.2	26.5	**27.3**

tector, and $(w_k^*, h_k^*)^f$ be the k^{th} part prediction with score $Pro^f_{(w^*,h^*,k)}$ from the fine part detector. Then we obtain the unified detection using the equation bellow:

$$(w_k^{**}, h_k^{**}) = \begin{cases} (w_k^*, h_k^*)^f & \text{if } Pro(w^*, h^*, k)^f \geq \mu \\ (w_k^*, h_k^*)^{d^*} & \text{otherwise,} \end{cases} \tag{4}$$

where $d^* = \underset{d}{\operatorname{argmax}}\, Pro^d_{(w^*,h^*,k)}$, $\mu \in [0, 1]$ is a threshold that controls how much the coarse and fine detectors contribute to the prediction. When $\mu = 0$, only the fine detector is used for detection, and when $\mu = 1$, the final output is determined by the coarse detectors.

4. Experiments

To evaluate the efficacy and generality of our method, we study two different keypoint localization problems including bird part detection and human pose estimation. We compare our CFN with existing methods on three datasets including CUB-200-2011 [42], LSP[19], and MSCOCO-Keypoint [22].

4.1. Bird Part Localization

The CUB200-2011 [42] is a widely used dataset for bird part localization. It contains 200 bird categories and $11,788$ images with roughly 30 training images per category. Each image has a bounding box and 15 key-point annotations. Here we adopt both the PCP and PCK criteria and compare our results to the reported performance of the state-of-the-art methods. We present the PCP results for each part as well as the total PCP results in Table 2. Compared to the methods that report PCP results, our method improves the overall PCP over the second best approach by 4.3%. Notably, although previous methods show poor performance of the 'leg' and 'back' part detection, our method achieves up to 33.8% and 9.8% improvements for the two parts over the next best method. We also report per-part PCK and mean PCK results compared with other methods with $\alpha \in \{0.1, 0.05, 0.02\}$ in Table 1. Here, a smaller α means more strict error tolerance in the PCK metric. Our method outperforms existing techniques at various α setting. This nicely demonstrates our approach produces more accurate predictions with a higher recall for keypoint localization. Also, the most striking result is that our approach obtains a 35% improvement over the second best method using the strict PCK metric. Figure 4 shows some qualitative results on the CUB200-2011 testing set.

In order to further understand the performance gains provided by our network structure, we also provide intermediate results of using strict supervision and multi-level supervisions. As shown in Table 3, using multi-level supervisions to learn the convolutional network achieves better performance than using the strict supervision alone. This is mainly because imposing appropriate supervision can significantly improve the accuracy of the coarse detectors, thereby enhance the performance of the unified detection. Moreover, the performance gain gradually diminishes as α decreases, because coarse detectors fail to predict very accurate locations and contribute less to the final predictions.

Table 4: Comparison of PCK (%) score at the level of 0.2 on the LSP dataset.

Methods	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	PCK
Carreira et al. [3]	90.5	81.8	65.8	59.8	81.6	70.6	62.0	73.1
Chen&Yuille [46]	91.8	78.2	71.8	65.5	73.3	70.2	63.4	73.4
Yang et al. [45]	90.6	78.1	73.8	68.8	74.8	69.9	58.9	73.6
Yu et al. [47]	87.2	88.2	82.4	76.3	91.4	85.8	78.7	84.3
Insafutdinov et al.[32]*	97.4	92.7	87.5	84.4	91.5	89.9	87.2	90.1
Wei et al. [43]*	97.8	92.5	87.0	83.9	91.5	90.8	89.9	90.5
Chu et al. [6]*	**98.1**	**93.7**	**89.3**	**86.9**	**93.4**	**94.0**	**92.5**	**92.6**
CFN	94.3	87.9	81.8	77.5	83.2	85.8	81.0	84.5

* Methods trained by adding MPII training set to the LSP training and LSP extended training set.

Table 5: Results on COCO keypoint on test-dev and test-standard split

Method	AP	AP OKS=0.50	AP OKS=.75	AP medium	AP large
Test-Dev					
CMU-Pose [2]	61.8	84.9	67.5	57.1	68.2
G-RMI (COCO-only) [29]	64.9	85.5	71.3	62.3	70.0
G-RMI (COCO-int) [29]	68.5	87.1	**75.5**	65.8	**73.3**
Mask R-CNN (Keypoint-only) [15]	62.7	87.0	68.4	57.4	71.1
Mask R-CNN (Keypoint & mask) [15]	63.1	**87.3**	68.7	57.8	71.4
RMPE [9]	61.0	82.9	68.8	57.9	66.5
CFN	**72.6**	86.1	69.7	**78.3**	64.1
Test-Std					
CMU-Pose [2]	61.1	84.4	66.7	55.8	68.4
G-RMI(COCO-only) [29]	64.3	84.6	70.4	61.4	69.6
G-RMI(COCO-int) [29]	67.3	85.4	**73.5**	64.2	**72.6**
CFN	**72.2**	**85.7**	68.8	**78.6**	63.7

4.2. Human Pose Estimation

Leeds Sports Pose Leeds Sports Pose (LSP) [19] dataset is a well established benchmark for human pose estimation. The original LSP dataset contains 2,000 images with 14 joint annotations. To reduce overfitting, we combine the first 1,000 images of the original LSP and all images from the extended LSP dataset [20] for training.

MSCOCO-Keypoint Challenge. The MSCOCO Keypoint dataset consists of 100k people with over 1 million total annotated keypoints for training and 50k people for validation. The testing set is unreleased and includes "test-challenge, "test-dev", and "test-standard" three subsets, each containing about 20K images. The MSCOCO evaluation defines the object keypoint similarity (OKS) and use AP (averaged across all 10 OKS thresholds) as the main metric to evaluate the keypoint performance.

Implementation. For experiments on LSP dataset, we first estimate the center locations and rough scales according to joint annotations or image sizes in order to resize the images into the same scale. We then crop or pad the scaled images into 448×448 according to the center positions. We also adopt the same augmentation scheme in [43] during network training. To address the problem of multi-person pose estimation in MSCOCO-Keypoint dataset, we adopt the Faster R-CNN framework [34] with a pre-trained model[1] on the MSCOCO dataset to obtain person bounding boxes. We extend the bounding boxes by 30 pixels along both sides and crop out the person instances. We also resize the long side of each image to 512 pixels while maintaining its aspect ratio. We pad each resized image with zero pixels and form a training example of size 512×512. Then we randomly crop the image into 448×448 as the input of the multi-level supervised nets. We train our model for 300k iterations using SGD with a momentum of 0.9, a batch size of 16, and an initial learning rate of 0.001 with step decay 100k. We initialize network weights with a pre-trained model on ImageNet which is available online [2].

Results on MSCOCO-Keypoint. We evaluate our method on the MSCOCO-Keypoint dataset. Our model uses a single person-detector and the provided training data only.

[1]https://github.com/rbgirshick/py-faster-rcnn
[2]https://github.com/lim0606/caffe-googlenet-bn

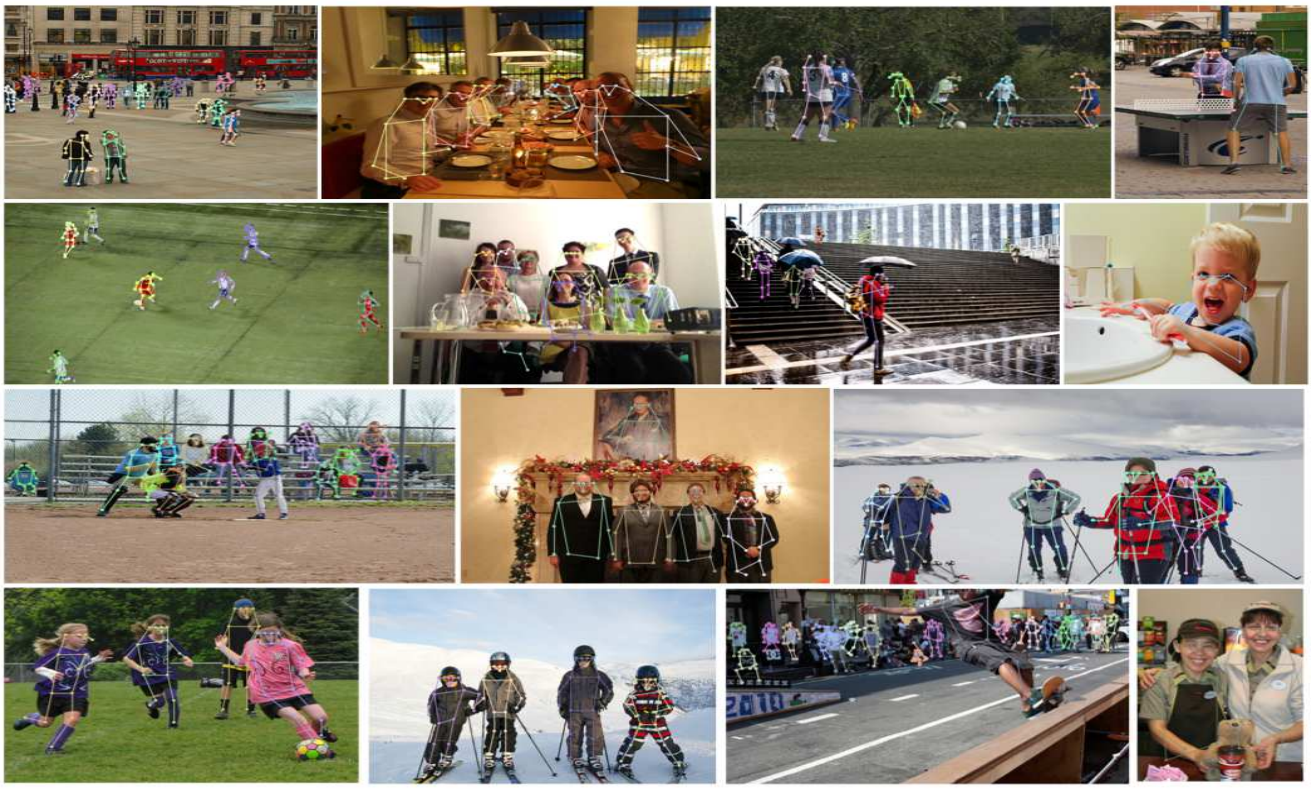

Figure 5: Pose estimation results with occlusion, crowding, deformation, and low resolution from the COCO test set.

Quantitative results evaluated from the online server [3] are given in Table 5. Our method achieves 72.2 AP, which is an 18% improvement over the winning team, and also significantly outperforms the recently proposed methods. Note that the significant overall improvement is mainly attributed to the improvement in the performance on medium-sized persons ($32^2 < area < 96^2$). Table 5 shows that our medium-sized result (78.3 AP^M) is 12.5 higher than the second best method [29] that uses extra data and ensemble person-detector. The superior performance on medium-sized metric (AP^M) demonstrates that the proposed method is particularly effective for the cases where the keypoint appearance is indistinct or ambiguous. Figure 5 shows some qualitative pose estimation results on the MSCOCO testing set. It is also worth noting that our caffe [18] implementation of CFN runs at 48 frames/sec on a TitanX GPU at the inference stage. Our method allows for real-time human pose estimation together with a fast person detector.

Results on LSP. We evaluate our method on the LSP dataset using the Percentage Correct Keypoints (PCK) metric with person-centric (PC) annotations. Our method achieves 84.5% accuracy, which is slightly better than the state-of-the-art method [47] without adding MPII training data. The improvement is not significant possibly because most persons in the LSP dataset are large enough to have distinctive keypoint appearance while our method is more advantageous for medium-sized persons.

[3] https://competitions.codalab.org/competitions/12061

5. Conclusion

In this paper, we have proposed a coase-fine convolutional network for keypoint localization on birds and humans. Our method fully explores hierarchical representations in CNNs by constructing a series of part detectors which are trained using multi-level supervisions. The multi-level supervisions supervise each network branch according to the localization ability of the detectors built from different feature layers in a CNN. The outputs of all the part detectors are principally unified to deliver promising performance for both bird part localization and human pose estimation.

Acknowledgements

The work is partially supported by Australian Research Council Projects FL-170100117, DP-140102164, and LP-150100671.

References

[1] M. Andriluka, S. Roth, and B. Schiele. Pictorial structures revisited: People detection and articulated pose estimation. In *CVPR*, 2009. 2

[2] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017. 1, 3, 7

[3] J. Carreira, P. Agrawal, K. Fragkiadaki, and J. Malik. Human pose estimation with iterative error feedback. In *CVPR*, 2016. 2, 3, 7

[4] Y. Chai, V. Lempitsky, and A. Zisserman. Symbiotic segmentation and part localization for fine-grained categorization. In *ICCV*, 2013. 2

[5] X. Chu, W. Ouyang, H. Li, and X. Wang. Structured feature learning for pose estimation. In *CVPR*, 2016. 2

[6] X. Chu, W. Yang, W. Ouyang, C. Ma, A. L. Yuille, and X. Wang. Multi-context attention for human pose estimation. *CVPR*, 2017. 7

[7] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 2

[8] M. Dantone, J. Gall, C. Leistner, and L. Van Gool. Human pose estimation using body parts dependent joint regressors. In *CVPR*, 2013. 2

[9] H. Fang, S. Xie, and C. Lu. Rmpe: Regional multi-person pose estimation. *CVPR*, 2017. 7

[10] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *TPAMI*, 32(9):1627–1645, 2010. 2

[11] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 61(1):55–79, 2005. 2

[12] Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning*, pages 1180–1189, 2015. 2

[13] M. Gong, K. Zhang, T. Liu, D. Tao, C. Glymour, and B. Schlkopf. Domain adaptation with conditional transferable components. In *ICML*, pages 2839–2848, 2016. 2

[14] C. Gu, J. J. Lim, P. Arbeláez, and J. Malik. Recognition using regions. In *CVPR*, 2009. 3

[15] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. *arXiv preprint arXiv:1703.06870*, 2017. 7

[16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 2

[17] S. Huang, Z. Xu, D. Tao, and Y. Zhang. Part-stacked cnn for fine-grained visual categorization. In *CVPR*, 2016. 1

[18] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ICM*, 2014. 8

[19] S. Johnson and M. Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *BMVC*, 2010. 6, 7

[20] S. Johnson and M. Everingham. Learning effective human pose estimation from inaccurate annotation. In *CVPR*, pages 1465–1472. IEEE, 2011. 7

[21] L. Karlinsky and S. Ullman. Using linking features in learning non-parametric part models. In *ECCV*, 2012. 2

[22] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 6

[23] J. Liu and P. N. Belhumeur. Bird part localization using exemplar-based models with enforced pose and subcategory consistency. In *ICCV*, 2013. 2, 5

[24] J. Liu, Y. Li, and P. N. Belhumeur. Part-pair representation for part localization. In *ECCV*, 2014. 1, 2, 5

[25] W. Liu and I. Tsang. On the optimality of classifier chain for multi-label classification. In *Advances in Neural Information Processing Systems*, pages 712–720, 2015. 2

[26] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *ECCV*, 2015. 2

[27] A. Newell, K. Yang, and J. Deng. Stacked hourglass networks for human pose estimation. In *CVPR*, 2016. 1, 2, 3

[28] D. Novotny, D. Larlus, and A. Vedaldi. I have seen enough: Transferring parts across categories. 2016. 2

[29] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. Murphy. Towards accurate multi-person pose estimation in the wild. In *CVPR*, 2017. 1, 2, 7, 8

[30] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele. Poselet conditioned pictorial structures. In *CVPR*, 2013. 2

[31] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele. Strong appearance and expressive spatial models for human pose estimation. In *ICCV*, 2013. 1, 2

[32] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. Gehler, and B. Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In *CVPR*, 2016. 2, 7

[33] V. Ramakrishna, D. Munoz, M. Hebert, J. A. Bagnell, and Y. Sheikh. Pose machines: Articulated pose estimation via inference machines. In *ECCV*, 2014. 1, 2

[34] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 3, 7

[35] K. J. Shih, A. Mallya, S. Singh, and D. Hoiem. Part localization using multi-proposal consensus for fine-grained categorization. In *BMVC*, 2015. 1, 5

[36] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 1

[37] M. Sun and S. Savarese. Articulated part-based model for joint object detection and pose estimation. In *ICCV*, 2011. 2

[38] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 1, 2, 3

[39] Y. Tian, C. L. Zitnick, and S. G. Narasimhan. Exploring the spatial hierarchy of mixture models for human pose estimation. In *ECCV*, 2012. 2

[40] 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 *NIPS*, 2014. 1, 2

[41] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *CVPR*, 2014. 2

[42] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 6

[43] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *CVPR*, 2016. 1, 2, 3, 7

[44] Z. Xu, S. Huang, Y. Zhang, and D. Tao. Webly-supervised fine-grained visual categorization via deep domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 2016. 2

[45] W. Yang, W. Ouyang, H. Li, and X. Wang. End-to-end learning of deformable mixture of parts and deep convolutional neural networks for human pose estimation. In *CVPR*, 2016. 2, 7

[46] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, 2011. 1, 2, 7

[47] X. Yu, F. Zhou, and M. Chandraker. Deep deformation network for object landmark localization. *ECCV*, 2016. 5, 7, 8

[48] H. Zhang, T. Xu, M. Elhoseiny, X. Huang, S. Zhang, A. Elgammal, and D. Metaxas. Spda-cnn: Unifying semantic part detection and abstraction for fine-grained recognition. In *CVPR*, 2016. 1, 2

[49] N. Zhang, J. Donahue, R. Girshick, and T. Darrell. Part-based r-cnns for fine-grained category detection. In *ECCV*, 2014. 2

[50] N. Zhang, R. Farrell, F. Iandola, and T. Darrell. Deformable part descriptors for fine-grained recognition and attribute prediction. In *CVPR*, 2013. 2

[51] N. Zhang, E. Shelhamer, Y. Gao, and T. Darrell. Fine-grained pose prediction, normalization, and recognition. *arXiv preprint arXiv:1511.07063*, 2015. 1, 5

[52] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014. 2