Bird Part Localization Using Exemplar-Based Models with Enforced Pose and Subcategory Consistency

Jiongxin Liu and Peter N. Belhumeur
Columbia University
{liujx09, belhumeur}@cs.columbia.edu

Abstract

In this paper, we propose a novel approach for bird part localization, targeting fine-grained categories with wide variations in appearance due to different poses (including aspect and orientation) and subcategories. As it is challenging to represent such variations across a large set of diverse samples with tractable parametric models, we turn to individual exemplars. Specifically, we extend the exemplar-based models in [4] by enforcing pose and subcategory consistency at the parts. During training, we build pose-specific detectors scoring part poses across subcategories, and subcategory-specific detectors scoring part appearance across poses. At the testing stage, likely exemplars are matched to the image, suggesting part locations whose pose and subcategory consistency are well-supported by the image cues. From these hypotheses, part configuration can be predicted with very high accuracy. Experimental results demonstrate significant performance gains from our method on an extensive dataset: CUB-200-2011 [30], for both localization and classification tasks.

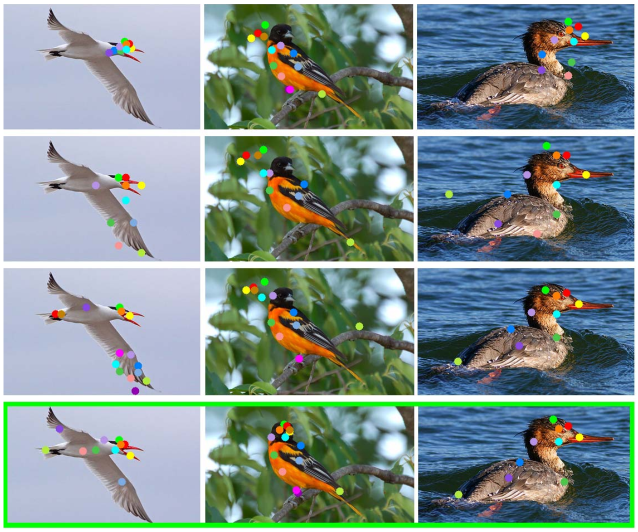


Figure 1. Part localization results from four methods. 1st row: Poselets [6], 2nd row: Mixture of Trees [36], 3rd row: Consensus of Exemplars [4], 4th row: Our method. Please refer to Fig. 3 for the color codes. Our method generates the most accurate part configurations for all the examples.

1. Introduction

Recent history has shown the growing importance of parts in object detection and classification. Especially for fine-grained categories (*e.g.*, birds [29, 17, 34], dogs [24, 20], butterflies [31], etc.), parts capture useful information to differentiate subcategories, and generally the accuracy of part localization has a significant impact on the effectiveness of localized features representing the object. Therefore, localizing the parts automatically and accurately is very important to a working system of fine-grained classification. In this paper, we focus on birds as the test case with the goal of localizing the parts across different bird species.

What makes detecting birds and bird parts difficult are the extreme variations in pose (*e.g.*, walking, perching, flying, swimming, etc.) coupled with possibly greater variations in appearance across species. A well-known detection method which had success on pedestrian [12] could not perform well on birds, as shown in [21]. Even more recent methods fail to detect birds with satisfactory accuracy [18, 6, 3]. In the domain of human pose estimation, there are well-constructed methods [32, 26, 27] that leverage the spatial relations of parts and predict the configuration pretty well, but they are still likely to struggle in the bird domain [8] due to the following limitations: to control the model complexity, they use tree structure and limit the spatial relations to two connected parts, which does not capture higher-order constraints on a group of parts; for the same reason, they use a limited number of pose types to decompose the visual complexity, which is not sufficient for birds with both articulated deformation and wide shape

This work was supported by NSF award 1116631, ONR award N00014-08-1-0638, and Gordon and Betty Moore Foundation grant 2987.

variations; finally, they ignore the subcategory consistency which is an important cue to detect a real bird.

As it is challenging to model the appearance variations of birds accurately with tractable parametric models, we turn to individual exemplars as in [4]. The hope is that when a sufficiently large number of training samples are available, we can always find a configuration similar to the testing sample. In contrast to [4] where exemplars only dictate the layout of parts, we propose to enforce pose and subcategory consistency at the parts. To do this, we design pose-specific detectors to score the pose types for each part, similar to [32]. Our novelty lies in that instead of parameterizing the pose types in an objective function, we associate them with non-parametric exemplars, by which we strictly constrain the co-occurrence of part poses of a bird.

Subcategory consistency means that the appearance at the detected parts should agree on the class membership. To enforce such consistency, we make use of the species labels to build species-specific part detectors. Such detectors are forced to focus on features invariant to poses, such as patterns in the interior of parts. At the testing stage, we predict the part locations by matching likely exemplars to the image so that the estimated parts not only form a globally plausible configuration, but also satisfy the pose and subcategory consistency well. In this way, we achieve significantly more accurate part localization than previous methods, as shown in Fig. 1 and Tab. 2.

Our paper makes the following contributions:

1. We propose the idea of enforcing subcategory consistency for part localization.
2. We show how to impose strong constraints on the parts by using pose and subcategory consistency and associating them with exemplars.
3. We produce state-of-the-art performance on an extensive bird dataset: CUB-200-2011 [30], for both part localization and species classification.

2. Related Work

Parts have a remarkable effect on object detection and classification, as demonstrated by Deformable Part Models (DPMs) [18, 35, 3] and fine-grained classification methods [17, 29, 34, 20, 5]. In DPMs, the parts are defined by local regions with relatively fixed spatial layout. Parts can be labeled as keypoints, and Poselets [7, 6] are proposed for object detection where consistent poselet activations make similar prediction of the keypoints. For fine-grained classification, detecting the object is not enough, as the support region and part locations of the object can vary a lot given the same object bounding box. [2] combines top-down scanning part detectors and bottom-up region hypotheses to generate region-based features for semantic segmentation of objects; while we combine pose-specific and subcategory-specific part detectors to predict part locations.

Accurate part localization requires prior knowledge about the global shape. Statistical shape models like Active Shape Models [23] and Active Appearance Models [11] model the shape with multivariate Gaussian distribution. Though subsequent works improve the model fitting algorithm [22, 25], such models still have difficulty handling a wide range of deformations. Another family of models is tree-based structure, including pictorial structure [19] and its variant [16, 32, 27, 36]. These models encode the spatial relations between parts with tree structure. Mixture of components, either locally [32] or globally [36], is proposed to decompose pose complexity. However, as previously mentioned, such models have the limitations that impair their efficacy in the bird domain.

Non-parametric models can also serve as shape prior. [1, 4] both fit annotated shape models to the image on top of local landmark detection. [1] uses a generic 3D face model, limiting its applicability to relatively rigid shapes; whereas [4] combines the output of local detectors with a set of exemplars, which potentially capture all possible configurations. We also use exemplars, but we impose much stronger constraints on the parts by enforcing pose and subcategory consistency. In addition, we predict part visibilities which are not considered in [4].

Another thread of research is shape regression, which has been successfully applied to facial feature localization [13, 9]. However, these methods are not applicable to our problem because they require rough bounding box of object as input, which is hard to obtain automatically for birds. Moreover, the complex image patterns on birds make it hard to learn features that are correlated to the shape incremental, which limits their efficacy.

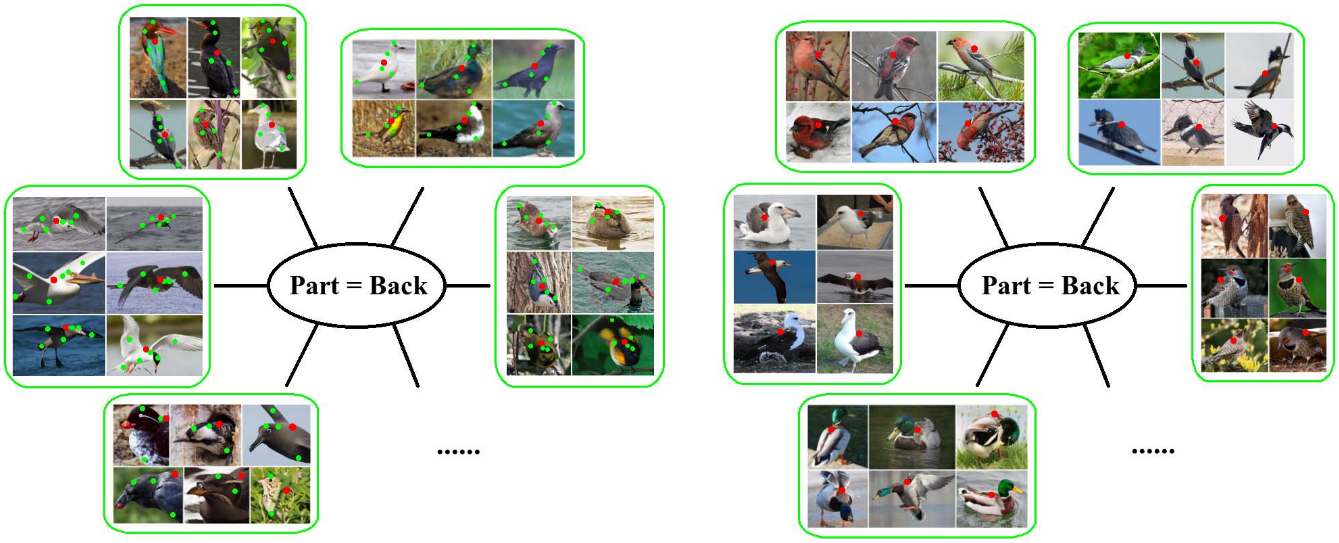
Poselets [6] can also predict the part locations given a cluster of poselet activations. However, by design, Poselets do not target part localization, and the rough prediction of part locations from each poselet activation may deviate greatly from the correct positions. Also, its heuristic way of rescoring and grouping activations usually cannot generate the optimal group of activations, in which case part localization will be hurt much more than object detection.

3. Pose and Subcategory Detectors

As the building block of our method, we build part detectors that score pose-specific and subcategory-specific features. To do this, we group the samples of each part based on their poses and species memberships.

3.1. Pose Grouping

We obtain the pose grouping by using part annotations, as the keypoint configuration around a part roughly captures its pose, including aspect and orientation. Let X_k denote the k-th exemplar, which contains a binary vector



(a) Pose clusters

(b) Subcategory clusters

Figure 2. Examples of the pose clusters and subcategory clusters for the part "Back", which is marked with a red dot in each image. In (a), the set of visible neighboring parts are marked with green dots, and note how the keypoint configurations are related to the poses.

of part visibilities and locations for visible parts. Taking part i of X_k as an example, we represent its pose with a local shape vector $\triangle_k^i = [\triangle x_k^{i,j_1}, v_k^{j_1}, \dots, \triangle x_k^{i,j_{m_i}}, v_k^{j_{m_i}}]$ where $\{j_1, \dots, j_{m_i}\}$ are the indices of m_i ($m_i = 6$ in our experiment) predefined neighboring parts for all the exemplars. $v_k^j \in \{0, 1\}$ is the visibility flag. if $v_k^j = 0$, *i.e.*, j is not visible, then $\triangle x_k^{i,j} = (0, 0)$; otherwise, $\triangle x_k^{i,j}$ is computed as $x_k^j - x_k^i$ where x_k^i is the location of part i. To deal with birds of different sizes, $\triangle_k^i$ is normalized such that $\sum_j \|\triangle x_k^{i,j}\|^2 = \sum_j v_k^j$. The set of all local shape vectors of part i form a pose space, whose subdivisions define the pose types, and we use k-means to generate N^i types. Fig. 2 (a) shows several examples of pose clusters for the part "Back" where $N^i = 200$.

For each pose cluster of each part, a detector is built using the samples in that cluster as positive samples, where a much larger set of negative samples are randomly drawn from image regions not containing that part of any type. So, by design, the detectors are trained to score the local pose across subcategories.

3.2. Subcategory Grouping

The underlying assumption of subcategory grouping is that samples from the same subcategory have similar appearance at the parts like colors, textures, etc., which holds in our problem because the species labels are already very fine-grained. Given these labels, it is straightforward to obtain the subcategory grouping for each part. The number of clusters is fixed because we have a fixed number (which is 200) of bird species. Fig. 2 (b) shows several examples of subcategory clusters.

Similar to pose detectors, a subcategory detector is built for each cluster of each part. To make the subcategory detectors learn species-specific features in an effective way, we do two things during training: we first normalize the orientation of parts to reduce the noise in the features. The normalization is done by aligning each part sample to a reference part sample using Procrustes analysis with "reflection" enabled based on their local shape vectors defined in Sec. 3.1. Secondly, we run the pose detectors exhaustively on the training images, and collect false activations (which are off the correct part locations) to form the negative training samples. Therefore, the subcategory detectors are able to learn subcategory-specific features across poses.

3.3. Implementation Details

We use linear SVMs implemented in LIBSVM [10] to build pose and subcategory detectors. The features are HOG descriptors extracted using VLFeat toolbox [28]. The scale of part is normalized based on its local shape vector. For each normalized part, HOG descriptors are extracted from a window centered at that part, which contains 5×5 cells with bin size 8. To scan the image over scales, we use a scaling factor of 1.2 to build the image pyramid.

Because pose and subcategory detectors play different roles in our method (see Sec. 4), there are some differences in their features. For pose detectors, we extract two additional HOG descriptors at a coarser scale and a finer scale, which are two levels above and below the normalized scale respectively in the image pyramid. For subcategory detectors, we extract three additional color histograms using 64 color bins, which are obtained through k-means in the RGB color space of training images. The histograms are computed over three regions: an inner circle and two outer rings.

4. Part Localization Approach

We cast the problem of part localization as fitting likely exemplars to an image, with the assumption that we can always find a similar configuration to the testing sample from a sufficiently large training set. Recall that X_k is the k-th exemplar, which contains the locations of visible parts. By using a similarity transformation t, we map X_k to the testing image, obtaining an exemplar-based model $X_{k,t}$. Our goal is to estimate its conditional probability $P(X_{k,t}|I)$, which measures how likely it is that the shape X_k exists in the image at a certain location, scale, and orientation.

In [4] where all the information of the image comes in the form of response maps D, $P(X_{k,t}|I)$ is computed as

$$P(X_{k,t}|I) = P(X_{k,t}|D) = \prod_{i=1}^{n} P(x^i_{k,t}|d^i), \qquad (1)$$

where n denotes the number of parts, $x^i_{k,t}$ is the image location of part i, and d^i is the corresponding response map. However, this formulation cannot be directly applied to our problem: first, it assumes there is a single detector applied at a fixed scale for each part, while we have an ensemble of detectors applied over scales; second, it does not address part visibilities, while there are 736 different combinations of visible parts in the dataset CUB-200-2011 [30].

Besides addressing the above issues, our major contribution is enforcing pose and subcategory consistency on $X_{k,t}$ to obtain a more accurate estimation of $P(X_{k,t}|I)$.

4.1. Pose Consistency

To evaluate $P(X_{k,t}|I)$ based on pose consistency, we generate a collection of response maps for all the parts × all the pose types, denoted as D_p. The key point is that for each exemplar X_k, we know the visibility of each part; if a part is visible, we also know its pose type. So in evaluating $P(X_{k,t}|D_p)$, we choose the response maps corresponding to the particular pose types of X_k. With these in hand, we compute $P(X_{k,t}|D_p)$ as

$$P(X_{k,t}|D_p) = \left(\prod_{i, v^i_k=1}^{n} P(x^i_{k,t}|d^i_p[c^i_k, s^i_{k,t}]) \right)^{\frac{1}{\sum_i v^i_k}}, \qquad (2)$$

where v^i_k denotes the visibility flag, $d^i_p[c^i_k, s^i_{k,t}]$ denotes the response map for pose type c^i_k at scale $s^i_{k,t}$. $s^i_{k,t}$ can be obtained based on the scaling factor in transformation t and the original size of part i. To obtain probability from Eq. 2, each response map is converted to a probability map using the detector calibration method described in [14]. Because the exemplars usually cannot fit the configuration of a testing sample perfectly, the probability maps are smoothed before evaluating $P(X_{k,t}|D_p)$. For efficiency, we use a max filter implemented by [15]. The filter radius is estimated by measuring the deviation between two corresponding parts from different exemplars after global alignment.

Because of the way $P(X_{k,t}|D_p)$ is computed, it is not plagued by false detections in other irrelevant response maps. Also, because of the reduced visual complexity in each pose cluster, each response map can give fairly accurate estimation of the part locations. For these reasons, the estimation of $P(X_{k,t}|D_p)$ is more reliable than $P(X_{k,t}|D)$ in [4]. From Eq. 2, we can see that given the response maps, the cost of subsequent computations (*i.e.*, evaluating a fixed number of $X_{k,t}$'s) is independent of the number of pose types, as opposed to [32, 36]. Therefore, we can increase the number of pose types a lot.

4.2. Subcategory Consistency

Subcategory Consistency means that the appearance at all the parts should agree with each other on the subcategory membership. Here, we assume that the image cues are contained in D_s, a collection of response maps for all the parts × all the subcategories. Given a subcategory l, we evaluate the likelihood of the image region occupied by $X_{k,t}$ matching a sample from that subcategory as

$$P(X_{k,t}|l, D_s) = \left(\prod_{i, v^i_k=1}^{n} P(x^i_{k,t}|d^i_s[l, s^i_{k,t}, \theta^i_{k,t}]) \right)^{\frac{1}{\sum_i v^i_k}}, \qquad (3)$$

where $d^i_s[l, s^i_{k,t}, \theta^i_{k,t}]$ denotes the response map for part i of subcategory l, at scale $s^i_{k,t}$ and in orientation $\theta^i_{k,t}$. $\theta^i_{k,t}$ can be computed based on the rotation angle in transformation t and the original orientation of part i. We use the same method as pose detector calibration to convert the response maps to probability maps. After computing $P(X_{k,t}|l, D_s)$ for all possible l's, $P(X_{k,t}|I)$ based on subcategory consistency is defined as

$$P(X_{k,t}|D_s) = \max_l P(X_{k,t}|l, D_s). \qquad (4)$$

4.3. Generating Hypotheses

After evaluating $X_{k,t}$'s pose and subcategory consistency, we evaluate $P(X_{k,t}|I)$ as

$$P(X_{k,t}|I) = P(X_{k,t}|D_p)^{\alpha} P(X_{k,t}|D_s)^{(1-\alpha)}, \qquad (5)$$

where parameter $\alpha \in [0, 1]$ controls the weights of $P(X_{k,t}|D_p)$ and $P(X_{k,t}|D_s)$, and α is determined through cross-validation, which is 0.8 in our experiment.

Because applying subcategory detectors in a sliding-window paradigm is very expensive (they need to search over scales and orientations) and not necessary (they are built on top of activations from pose detectors), we only generate the response maps for pose detectors, and construct a random transformation t for X_k as follows:

1. Randomly choose two parts and a scaling factor.
2. Select the two response maps for the chosen parts at the corresponding scales.
3. Randomly choose a local maxima from each map.
4. Compute similarity transformation t that maps the two parts of X_k to the two local maximas.
5. If the scaling factor or rotation angle in t is beyond a predefined range, t is declared as invalid.

By repeating the above procedure multiple times for each exemplar, we generate a large set of models $\{X_{k,t}\}$, whose conditional probabilities are computed using Eq. 5 and the top ranked models constitute the set of likely hypotheses.

Although subcategories detectors are now applied only to the generated $\{X_{k,t}\}$, it is still very expensive to extract the features due to the large number of models (about $380,000$ in our experiment). Instead, we approximate the procedure by computing $P(X_{k,t}|D_p)$ for all the models first (which is relatively much faster), and keeping the top ranked models (*e.g.*, 400 in our experiment) which will be re-ranked by incorporating $P(X_{k,t}|D_s)$. We observe that the performance is not hurt by this approximation as $P(X_{k,t}|D_p)$ already gives a fairly accurate estimation of the correctness of models matching the testing image.

As the models usually cannot match the testing sample perfectly, we also need to address the issue here when evaluating $P(X_{k,t}|D_s)$. Because we extract the features at the part locations dictated by the models, the errors in the part locations lead to underestimation of $P(X_{k,t}|D_s)$, in which case incorrect models may rank higher than correct ones. Therefore, we adopt a group-based re-ranking strategy. Given the ranked list of 400 models based on $P(X_{k,t}|D_p)$, we group their part configurations. More specifically, we successively take the model with the highest $P(X_{k,t}|D_p)$ out of the list, find and take out other models in the list with configurations close enough to it based on sum of the squared distances (SSD) between corresponding parts, thus forming a group. After that, the term $P(x^i_{k,t}|d^i_s[l, s^i_{k,t}, \theta^i_{k,t}])$ in Eq. 3 will be replaced by the highest value in the group $X_{k,t}$ belongs to. We can do so because the subsequent consensus operation in Sec. 4.4 does not expect the top ranked models (hypotheses) to suggest exactly the correct part locations.

4.4. Predicting Part Configuration

Given a set of $M = 40$ hypotheses with exemplar indices $\{k_m\}_{m=1,\dots,M}$, we first predict the visibility flag v^i for each part i through voting:

$$v^i = \left\{ \begin{array}{lcl} 1 & : & \sum_m v^i_{k_m} > \tau M \\ 0 & : & \text{Otherwise} \end{array} \right. , \qquad (6)$$

where threshold τ is determined through cross-validation, such that the False Invisibility Rate defined in Sec. 5.1 is on par with that of human annotators (about 6%). If a part is predicted as visible, we use the same method as [4] to estimate its location by combining the hypotheses and the probability maps corresponding to the relevant pose types.

As can be seen here, pose detectors mainly play the role in finding the parts while subcategory detectors focus on verifying the hypotheses suggested by pose detectors.

5. Experiments

5.1. Dataset and Evaluation Metrics

We test our method on CUB-200-2011 [30] dataset, which contains 11,788 uncropped images of 200 bird species (about 60 images per species). We use the train/test split provided in the dataset for all the experiments. There are roughly 30 images per species to train, and we do left-right flipping to increase the size of training data. A total of 15 parts were annotated by pixel location and visibility flag in each image through Mechanical Turk.

To gain a thorough view of the performance, we use four metrics to evaluate the localization performance: Percentage of Correctly estimated Parts (PCP), Average Error (AE), False Visibility Rate (FVR) and False Invisibility Rate (FIR). "Correct estimation" means the detected part is within 1.5 standard deviation of a MTurk user's click if visible or both estimated part and ground truth are invisible. "Average error" is computed by averaging the distance between predicted part locations and ground truth (if both are visible), normalized on a per-part basis by the standard deviation and bounded at 5. "False Visibility Rate" is the percentage of parts that are incorrectly estimated as visible; "False Invisibility Rate" is the percentage of parts that are incorrectly estimated as invisible. Note that AE best describes the accuracy of predicted part locations.

5.2. How The Number of Pose Types Matters

To examine the effect of the number of pose types, we only consider pose consistency here (*i.e.*, $\alpha = 1$ in Eq. 5). As shown in Tab. 1, we change the number of types for each part from 1 to $2,000$. Due to the huge visual complexity, we use RBF-SVM to build the detectors when the number of types is 1, which is the case of [4]. From the comparisons, we can see that with roughly fixed FIR (due to the way parameter τ is chosen in Sec. 4.4), the performance measures of PCP, AE and FVR are consistently improved as the number of types increases up to 500. To explain this, on the one hand, the larger the number of pose types, the more the visual complexity can be reduced. On the other hand, finer

Type Num.	PCP	AE	FVR	FIR
1 [4]	**48.70%**	**2.13**	43.90%	**6.72%**
1 (Relaxed)	47.08%	2.30	**39.36%**	7.12%
10 (Pose)	45.79%	2.37	44.21%	4.14%
50 (Pose)	53.07%	2.08	34.02%	4.40%
100 (Pose)	54.66%	2.00	31.21%	4.87%
200 (Pose)	56.88%	1.92	**30.16%**	4.32%
500 (Pose)	**57.03%**	**1.91**	30.21%	4.34%
1000 (Pose)	56.63%	1.94	31.26%	**3.91%**
2000 (Pose)	56.50%	1.97	32.35%	4.08%
10 (App.)	43.30%	2.55	42.10%	**4.48%**
50 (App.)	48.86%	2.29	32.55%	6.43%
100 (App.)	51.05%	2.20	32.01%	5.97%
200 (App.)	**52.10%**	**2.15**	**31.30%**	5.65%
500 (App.)	52.00%	2.17	31.57%	5.71%
1000 (App.)	51.32%	2.21	32.27%	5.46%
2000 (App.)	51.07%	2.26	32.31%	5.58%

Table 1. Part localization results using different numbers of pose types. The best performance is achieved with 500 pose types for each part. Appearance-based clustering [14] can also be used to generate pose types, which is inferior to ours in terms of the performance. Please refer to Sec. 5.1 for the meaning of each metric.

granularity of pose types makes the constraints on pose consistency stronger. As the number of pose types goes beyond $1,000$, the performance becomes slightly worse, possibly due to the fact that there are much fewer positive training samples. Given more than 50 pose types, our method significantly outperforms [4] where a single non-linear detector is used. We choose 200 types in subsequent experiments as it is a good trade-off between accuracy and speed (1.5× faster than 500 types and 2.6× faster than $1,000$ types). We also relax the pose constraint by collapsing the probability maps of all 200 pose types for each part to a single probability map by taking pixel-wise maximum, thus reducing the number of types to 1. But the accuracy drops a lot, demonstrating the effect of enforcing pose consistency.

We also try an alternative method [14] to define the pose types. [14] uses Latent-SVM learning to optimize the ensemble of detectors, leading to appearance-based clustering. As the visual appearance is coupled with pose, [14] actually groups samples similar in pose but with more noise than our pose clustering. Please see Tab. 1 for the comparisons.

5.3. Part Localization

We compare our work with three state-of-the-art techniques: Poselets [6], Mixture of Trees [36] and Consensus of Exemplars [4]. For Poselets-based part localization, we obtain the poselet activations from the authors of [34], and follow [6] to predict the location of each part as the average prediction from its corresponding poselet activations. For Mixture of Trees, we obtain the detected part locations from the authors of [8], which is a special case of [32] and the counterpart of [36] with 12 global components in the bird domain. Note that [8] only detects 13 parts, omitting the two legs. We modified Consensus of Exemplars [4] to deal with part visibilities. Generally, larger FVR comes with larger AE, so [4]'s performance measure of AE benefits from our modification.

Method	PCP	AE	FVR	FIR
Poselets [6]	27.47%	2.89	47.90%	17.15%
Mix. of Trees [36]	40.99%	2.65	32.62%	6.18%
Consensus [4]	48.70%	2.13	43.90%	6.72%
Ours	**59.74%**	**1.80**	**28.48%**	**4.52%**
Human	84.72%	1.00	20.72%	6.03%

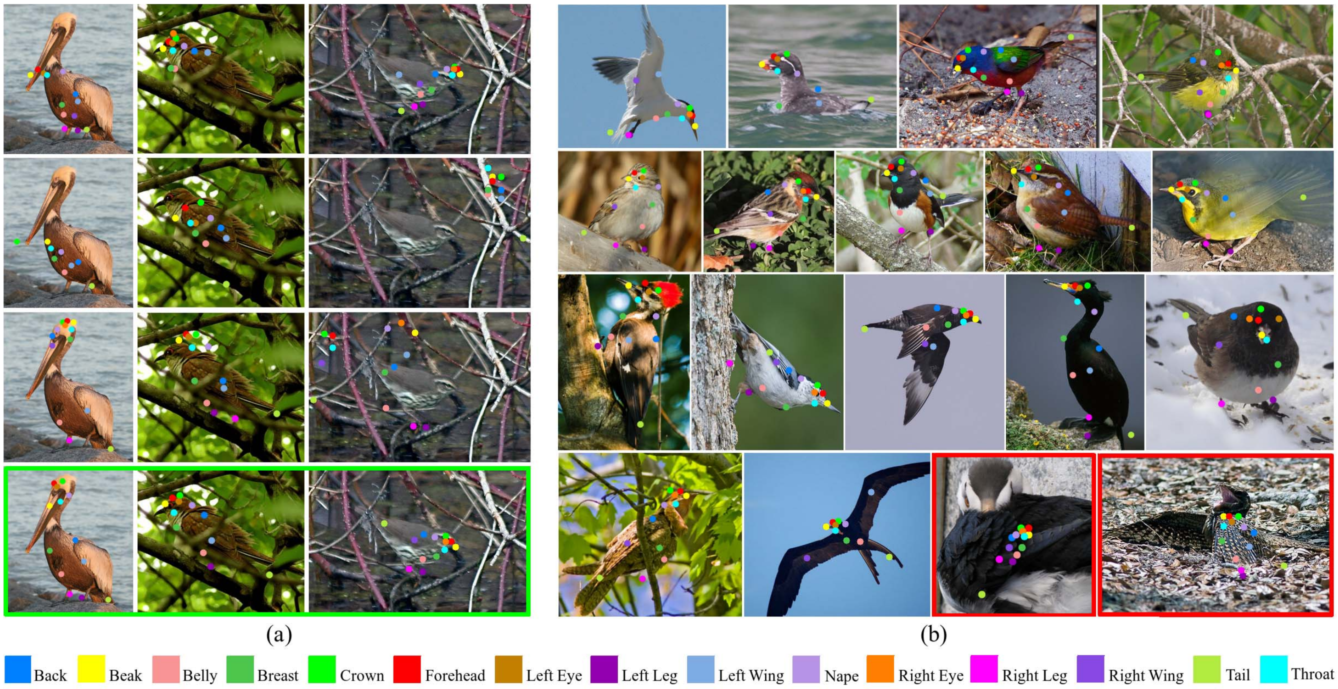
Table 2. Part localization results from different methods. Our method significantly outperforms state-of-the-art techniques on all the four metrics.

As shown in Tab. 2, our part localization outperforms state-of-the-art techniques on all the metrics. The large error rate of Poselets agrees with the fact that by design, they do not target localizing the parts with high accuracy. Compared with the results in Tab. 1, our full model incorporating the subcategory consistency achieves remarkable improvement on AE. In a separate experiment, we set $\alpha = 0$ in Eq. 5, and obtain 58.28% for PCP, 1.86 for AE, 28.88% for FVR, and 5.32% for FIR. It indicates that pose consistency and subcategory consistency are complementary to each other. Without predicting visibilities (*i.e.*, $\tau = 0$ in Eq. 6), our full model obtains 54.36% for PCP, 1.85 for AE, 60.03% for FVR, and 0.28% for FIR, which are not much worse except FVR. Some examples of our bird part localization are shown in Fig. 3. Although birds have very wide variations in appearance and pose, and birds reside in very different environments, our method is still able to detect most of the parts correctly.

5.4. Part-Based Species Classification

To demonstrate how the accuracy of part localization affects the species classification, we feed the estimated part locations to our part-based classification method [20]. We train one vs. all SVMs with RBF kernel for each species, and extract grayscale SIFT and color histograms as features. Specifically, we center 12 SIFT windows at the 15 parts (for symmetrical parts like left/right eyes, we randomly choose one if both are visible), and the features for invisible parts are zeroed out. From the parts on the head and body, we construct two convex hulls respectively, and extract a color histogram from each convex hull with 64 bins obtained using k-means in the RGB color space.

In Fig. 4, we plot the Cumulative Match Characteristic (CMC) curves showing the classification accuracy against ranked guesses. From the CMC curves, we can see that the



(c)

Figure 3. Examples of Bird Part Localization. (a) compares the four methods (From top to bottom: Poselets [6], Mix. of Trees [36], Consensus of Exemplars [4], Our method) on three testing samples. (b) gives more examples of part localization using our method. Red frames denote failure cases. (c) shows the color codes for the 15 parts.

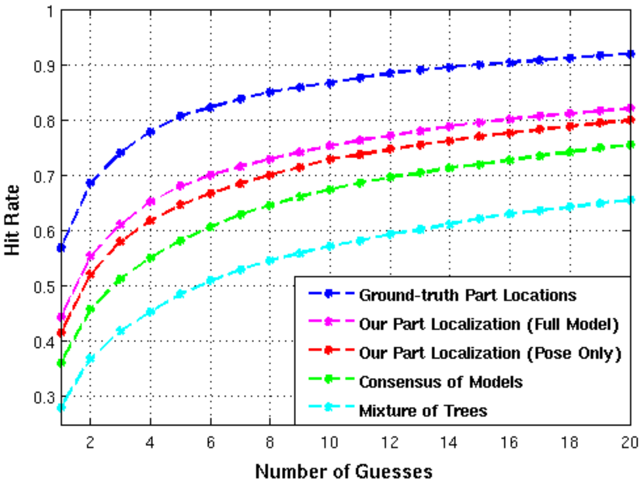


Figure 4. Cumulative Match Characteristic (CMC) curves for bird species classification.

classification performance is consistently improved along with the increased accuracy of part localization. As the classification method is very sensitive to the localization accuracy, we do not include Poselets in the curve comparison (the Rank 1 accuracy for Poselets is below 15%). The upper bound of the classification accuracy can be obtained with the ground-truth part locations. The comparison between our full model and our partial model with only pose consistency shows that adding the subcategory consistency leads to about 3% increase in the Rank 1 accuracy.

We also compare our method with other classification methods on the whole dataset as well as on a subset of 14 species in Tab. 3. Though the other methods may be more sophisticated in extracting the features or designing the classifiers, our method has much better results, which we believe is attributed to the accurate part localization. Moreover, we achieve state-of-the-art performance on the dataset in a fully automatic setting (without using ground-truth bounding boxes or ground-truth part locations from the testing data). Based on the experiment, we do feel accurate part localization goes a long way towards building a working system for fine-grained classification.

6. Conclusion

In this paper, we propose a simple and novel approach for bird part localization, as the test case of fine-grained categories. We introduce the idea of enforcing subcategory consistency at the parts, and show how to generate likely hypotheses of part configurations using exemplar-based models with enforced pose and subcategory consistency. The improved hypotheses over [4] enable us to better estimate

Method	200 species	14 species
Birdlets [17]	-	40.25%
Template bagging [33]	-	44.73%
Pose pooling kernel [34]	28.18%	57.44%
Ours	**44.13%**	**62.42%**

Table 3. Mean average precision (mAP) on the full 200 categories as well as 14 categories from [17] for different classification methods. [17] and [33] are not directly comparable to ours as they use an earlier version of the dataset.

the part locations using consensus. Experimental results demonstrate that our method achieves state-of-the-art performance for both part localization and species classification on the challenging dataset CUB-200-2011 [30].

References

[1] B. Amberg and T. Vetters. Optimal landmark detection using shape models and branch and bound. *Proc. ICCV*, 2011.

[2] P. Arbelaez, B. Hariharan, C. Gu, S. Gupta, L. Bourdev, and J. Malik. Semantic segmentation using regions and parts. *Proc. CVPR*, 2012.

[3] H. Azizpour and I. Laptev. Object detection using strongly-supervised deformable part models. *Proc. ECCV*, 2012.

[4] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. *Proc. CVPR*, 2011.

[5] T. Berg and P. N. Belhumeur. Poof: Part-based one-vs-one features for fine-grained categorization, face verification, and attribute estimation. *Proc. CVPR*, 2013.

[6] L. Bourdev, S. Maji, T. Brox, and J. Malik. Detecting people using mutually consistent poselet activations. *Proc. ECCV*, 2010.

[7] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. *Proc. ICCV*, 2009.

[8] S. Branson, P. Perona, and S. Belongie. Strong supervision from weak annotation: Interactive training of deformable part models. *Proc. ICCV*, 2011.

[9] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. *Proc. CVPR*, 2012.

[10] C.-C. Chang and C.-J. Lin. Libsvm : a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2011.

[11] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. *IEEE TPAMI*, 2001.

[12] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *Proc. CVPR*, 1:886–893, 2005.

[13] M. Dantone, J. Gall, G. Fanelli, and L. V. Gool. Real-time facial feature detection using conditional regression forests. *Proc. CVPR*, 2012.

[14] S. Divvala, A. A. Efros, and M. Hebert. How important are deformable parts in the deformable parts model? *Parts and Attributes Workshop, ECCV*, 2012.

[15] P. Dollár. Piotr's Image and Video Matlab Toolbox (PMT). `http://vision.ucsd.edu/~pdollar/toolbox/doc/index.html`.

[16] M. Everingham, J. Sivic, and A. Zisserman. hello! my name is... buffy automatic naming of characters in tv video. *Proc. BMVC*, 2006.

[17] R. Farrell, O. Oza, N. Zhang, V. I. Morariu, T. Darrell, and L. S. Davis. Birdlets: Subordinate categorization using volumetric primitives and pose-normalized appearance. *Proc. ICCV*, 2011.

[18] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE TPAMI*, 2010.

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

[20] J. Liu, A. Kanazawa, D. Jacobs, and P. Belhumeur. Dog breed classification using part localization. *Proc. ECCV*, 2012.

[21] T. Malisiewicz, A. Gupta, and A. A. Efros. Ensemble of exemplar-svms for object detection and beyond. *Proc. ICCV*, 2011.

[22] I. Matthews and S. Baker. Active appearance models revisited. *IJCV*, 2004.

[23] S. Milborrow and F. Nicolls. Locating facial features with an extended active shape model. *Proc. ECCV*, pages 504–513, 2008.

[24] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. V. Jawahar. Cats and dogs. *Proc. CVPR*, 2012.

[25] J. M. Saragih, S. Lucey, and J. F. Cohn. Face alignment through subspace constrained mean-shifts. *Proc. ICCV*, 2009.

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

[27] N. Ukita. Articulated pose estimation with parts connectivity using discriminative local oriented contours. *Proc. CVPR*, 2012.

[28] A. Vedaldi and B. Fulkerson. Vlfeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/`, 2008.

[29] C. Wah, S. Branson, P. Perona, and S. Belongie. Multiclass recognition and part localization with humans in the loop. *Proc. ICCV*, 2011.

[30] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. *Computation & Neural Systems Technical Report*, CNS-TR-2011-001, 2011.

[31] J. Wang, K. Markert, and M. Everingham. Learning models for object recognition from natural language descriptions. *Proc. BMVC*, 2009.

[32] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. *Proc. CVPR*, 2011.

[33] B. Yao, G. Bradski, and L. Fei-Fei. A codebook-free and annotation-free approach for fine-grained image categorization. *Proc. CVPR*, 2012.

[34] N. Zhang, R. Farrell, and T. Darrell. Pose pooling kernels for sub-category recognition. *Proc. CVPR*, 2012.

[35] L. Zhu, Y. Chen, A. Yuille, and W. Freeman. Latent hierarchical structural learning for object detection. *Proc. CVPR*, 2010.

[36] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. *Proc. CVPR*, 2012.