Deep LAC: Deep Localization, Alignment and Classification for Fine-grained Recognition

Di Lin† Xiaoyong Shen† Cewu Lu‡ Jiaya Jia†

† The Chinese University of Hong Kong ‡ Hong Kong University of Science and Technology

Abstract

We propose a fine-grained recognition system that incorporates part localization, alignment, and classification in one deep neural network. This is a nontrivial process, as the input to the classification module should be functions that enable back-propagation in constructing the solver. Our major contribution is to propose a valve linkage function (VLF) for back-propagation chaining and form our deep localization, alignment and classification (LAC) system. The VLF can adaptively compromise the errors of classification and alignment when training the LAC model. It in turn helps update localization. The performance on fine-grained object data bears out the effectiveness of our LAC system.

1. Introduction

Fine-grained object recognition aims to identify subcategory object classes, which includes finding subtle difference among species of animals, product brands, and even architectural styles. Thanks to recent success of convolutional neural networks (CNN) [13], good performance was achieved on fine-grained tasks [4, 27].

The large flexibility of CNN structures makes fine-grained recognition still have much room to improve. One challenge is that discriminative patterns (e.g., bird head in bird species recognition) appear possibly in different locations, and with rotation and scaling in the collected images. Although research of [17, 10] showed that CNN features are reasonably robust to scale and rotation variation, it is necessary to directly capture these types of change to increase the recognition accuracy [27, 4].

Existing solutions perform localization, alignment, and classification independently and consecutively. This procedure is illustrated in Figure 1 using solid-line arrows where parts are localized, aligned according to templates, and then fed into the classification neural network. Obviously, any error arising during localization could influence alignment and classification.

In this paper, we propose a feedback-control framework

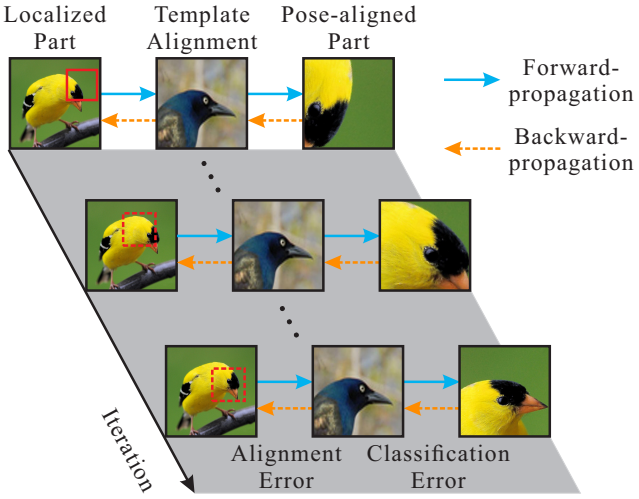


Figure 1: The one-way procedure from localization to template alignment makes each module rely on results from the previous one. Contrarily, back-propagation highlighted by dashed arrow makes it possible to refine localization according to the classification and alignment results. It forms a bi-directional refinement process.

to back-propagate alignment and classification errors to localization, in order to optimally update all states in iterations. This process is highlighted by dashed arrows in Figure 1, which, in our experiments, benefits final classification. This framework is constructed as one deep neural network including all localization, alignment and classification tasks.

The difficulty of forming a neural network for all modules stems from the special requirement of classification sub-network input. As shown in Figure 1, the input to classification is an image after alignment. It cannot achieve the back-propagation chain during the whole network solving due to the fact that the derivation of a constant, which is the aligned region, is zero.

The main focus of this paper is thus to propose a valve linkage function (VLF) in alignment sub-network to opti-

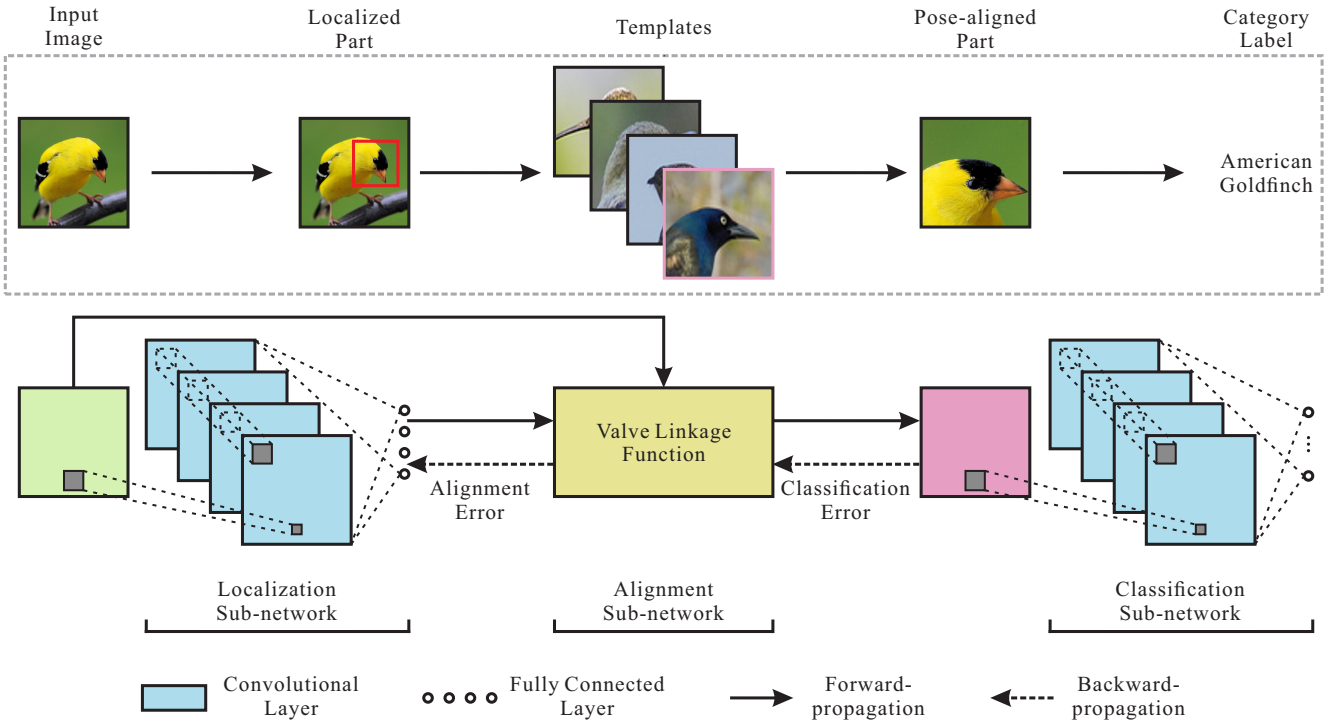


Figure 2: Deep LAC. It consists of localization, alignment and classification sub-networks. With the help of VLF, alignment sub-network outputs pose-aligned part images for classification in the FP stage, while classification and alignment errors can be propagated back to localization in the BP stage.

mally connect the localization and classification modules in our deep LAC framework. The architecture is shown in Figure 2. Because we involve these tasks in one network, forward-propagation (FP) and backward-propagation (BP) solving procedures become available.

In FP, VLF outputs a pose-aligned part image to classification. In BP, it should be a function containing necessary parameters for updating the localization sub-network. Our VLF not only connects all sub-networks, but also functions as information valve to compromise classification and alignment errors. If alignment is good enough in the F-P stage, VLF guarantees corresponding accurate classification. Otherwise, errors propagated from classification finely tune the previous modules. These effects make the whole network reach a stable state. Note this scheme is general, as similar VLF can be proposed for other networks that involve several modules and various parameters.

Other contribution includes the new localization and alignment sub-networks. As shown in Figure 2, localization [20] is with regression of part location. It differs from general object detection by making use of relatively stable relationship between the fine-grained object (e.g., bird) and part region (e.g., bird head), which contrarily cannot be preserved for general objects. On the alignment side, we introduce multi-template selection to effectively handle pose variance of parts.

With our system joint modeling localization, alignment, and classification, decent performance is accomplished in comparison to the solutions where these modules are considered independently. We apply our method to data for automatic classification without part annotation in the testing phase.

2. Related Work

Pioneer work in this area concentrated on constructing discriminative whole-image representation [22, 15, 23]. It was later found suffering from the problem of losing subtle difference between subordinate categories. Localization and alignment can ameliorate this problem by extracting parts from visually similar regions and reducing their variance. Recent work exploits these operations.

Farrell et al. [7] and Yao et al. [26] used templates to get the location of parts. Yang et al. [25] learned templates to localize important parts of fine-grained objects in an unsupervised manner. Gavves et al. [8] aligned images in order to accommodate the possibly large variation of poses. In [5, 1, 24], segmentation and part localization were unified in one framework to alleviate the distracting effect of background. Berg et al. [3] put forward part-based one-vs-one feature (POOF). Zhang et al. [28] enforced DPM to extract

part regions and features as image representation.

Recently, fine-grained recognition was achieved by combining localization, alignment, and deep neural networks. Zhang et al. [28] applied pre-trained convolutional neural network [13] to extract feature on the localized part. In their later work [27], selective search [21] was used for part proposal. Branson et al. [4] studied higher-order geometric warping to align parts. In [27, 4], the fine-tuned CNN model on dataset [22] was used to extract representation on parts. This method accomplished state-of-the-art results on bird identification [22].

These methods did not perform joint refinement of localization, alignment, and classification in one network. Employment of these modules together in our system is found profitable for fine-grained recognition. We give more details below.

3. Our Approach

To recognize fine-grained classes, we learn deep LAC models for distinct and meaningful parts. Features extracted on parts are used in classical classifiers, e.g. SVM. The main network consists of three sub-ones for the aforementioned three tasks – localization sub-network provides part position; alignment sub-network performs template alignment to offset translation, scaling and rotation of localized parts; pose-aligned parts are fed to the classification sub-network.

As aforementioned, the way to connect those three modules within a unified deep neural network is worth studying. In what follows, we first describe localization and classification sub-networks, which are implemented according to the CNN model [13]. Then we detail our alignment sub-network where forward-propagation (FP) and backward-propagation (BP) stages are implemented.

3.1. Localization Sub-network

Our localization sub-network outputs the commonly used coordinates for the top-left and bottom-right bounding-box corners denoted as (x_1, y_1) and (x_2, y_2), given an input natural image for fine-grained recognition. In the training phase, we regress bounding boxes of part regions. Ground truth bounding boxes are generated with part annotation. We unify input image resolution and construct a localization sub-network [13], which consists of 5 convolutional layers and 3 fully connected ones. Our last fully connected layer is a 4-way output for regressing bounding-box corners (x_1, y_1) and (x_2, y_2).

With output $\mathbf{L} = (x_1, y_1, x_2, y_2)$, our localization sub-network is expressed as

$$\mathbf{L} = f_l(\mathbf{W}_l; \mathbf{I}), \tag{1}$$

where $\mathbf{W}_l$ is the weight parameter set and $\mathbf{I}$ is the input image. During training, ground truth locations of parts $\mathbf{L}^{gt}$ are used. The location objective function is given by

$$E_l(\mathbf{W}_l; \mathbf{I}, \mathbf{L}^{gt}) = \frac{1}{2}||f_l(\mathbf{W}_l; \mathbf{I}) - \mathbf{L}^{gt}||^2. \tag{2}$$

We minimize it over $\mathbf{W}_l$. This framework works well on part location regression because the appearance of objects and part regions are generally stable in fine-grained tasks. The location of parts thus can be reasonably predicted. Figure 5 shows the examples of localized bird heads and bodies.

3.2. Classification Sub-network

The classification sub-network is the last module shown in Figure 2. Our classification takes the pose-aligned part image as input, denoted as ϕ^*, and generates the category label. This classification CNN [13] is expressed as

$$y = f_c(\mathbf{W}_c; \phi^*), \tag{3}$$

where $\mathbf{W}_c$ is the weight parameter set in this sub-network. The output is the category label y.

During training, the ground truth label y^{gt} is provided. The predicted category label y in Eq. (3) should be consistent with y^{gt}. We enforce a penalty on y, which is denoted as $E_c(\mathbf{W}_c; \phi^*, y^{gt})$. In classification, we follow the method of [13] to use softmax regression loss in order to penalize the classification error.

Our major contribution in this system is the construction of the alignment sub-network, which is detailed below together with the formulation of ϕ^* in Eq. (3).

3.3. Alignment Sub-network

Alignment sub-network receives part location $\mathbf{L}$ (i.e., bounding box) from the localization module, performs template alignment [18] and feeds a pose-aligned part image to classification, as shown in Figure 2. Our alignment sub-network offsets translation, scaling, and rotation for pose-aligned part region generation, which is important for accurate classification. Apart from pose aligning, this sub-network plays a crucial role on bridging the backward-propagation (BP) stage of the whole LAC model, which helps utilize the classification and alignment results to refine localization.

We propose a new valve linkage function (VLF) as the output of alignment sub-network to accomplish the above goals. In what follows, we present our alignment part and then detail our VLF in line with the FP and BP stages of the LAC model.

3.3.1 Template Alignment

We rectify localized part regions, making their poses close to the templates. To evaluate the similarity of poses, we

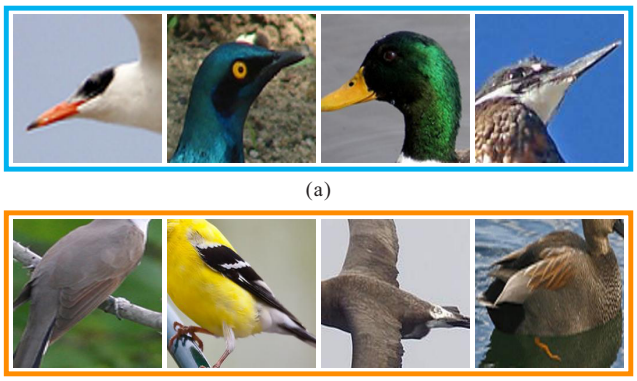


Figure 3: Examples of alignment templates of bird (a) head and (b) body.

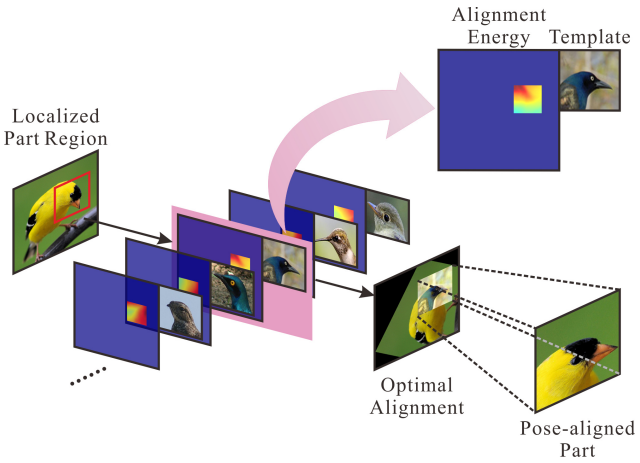


Figure 4: Alignment sub-network selects optimally pose-aligned parts for classification.

define the function between part regions $\mathbf{R}_i$ and $\mathbf{R}_j$ as

$$S[\mathbf{R}_i, \mathbf{R}_j] = \sum_{m=0}^{255}\sum_{n=0}^{255} \mathbf{p}_{ij}(m,n) log(\frac{\mathbf{p}_{ij}(m,n)}{\mathbf{p}_i(m)\mathbf{p}_j(n)}), \quad (4)$$

where $\mathbf{p}_i, \mathbf{p}_j \in \mathbb{R}^c$ form distributions of gray-scale values of uniform-size images $\mathbf{R}_i$ and $\mathbf{R}_j$ respectively. $\mathbf{p}_{ij} \in \mathbb{R}^{256\times256}$ is for the joint distribution. This pose similarity function is based on mutual information [18]. A large value means similar poses between $\mathbf{R}_i$ and $\mathbf{R}_j$.

To resist large pose variation, we generate a template set for alignment. For each pair in N training part images, we calculate the similarity using Eq. (4) and finally form a similarity matrix $S_t \in \mathbb{R}^{N\times N}$. S_t is then processed with spectral clustering [16] to split the N part images into K clusters. From each cluster, we select the part region closest to the cluster center as template to represent this set. To include mirrored poses, we also flip each template. Eventually, we obtain a template set $\mathfrak{T}$.

Figure 4 shows the pipeline of alignment. Given an input image $\mathbf{I}$, the regressed part bounding box $\mathbf{L}$ generated by localization sub-network and the center $\mathbf{c}^r(\mathbf{L}) = (\frac{x_1+x_2}{2}, \frac{y_1+y_2}{2})$ of the bounding box, we assume the pose-aligned part region is with center location $\mathbf{c}$, rotated with θ degree and is scaled with factor α. To compare it with a template $\mathbf{t}$, we extract the region, denoted as $\phi(\mathbf{c}, \theta, \alpha; \mathbf{I})$. Using the above similarity function, alignment is done by finding $\mathbf{c}$, θ, α and $\mathbf{t}$ that maximize

$$\begin{aligned} &E_a(\mathbf{c}, \theta, \alpha, \mathbf{t}; \mathbf{I}, \mathbf{L}) = \\ &\quad S[\phi(\mathbf{c}, \theta, \alpha; \mathbf{I}), \mathbf{t}] + \lambda \exp(-\frac{1}{2}\|\mathbf{c} - \mathbf{c}^r(\mathbf{L})\|^2), \\ &\mathbf{c} \in [x_1, x_2] \times [y_1, y_2], \theta \in \Theta, \alpha \in \mathfrak{A}, \mathbf{t} \in \mathfrak{T}, \end{aligned} \quad (5)$$

where λ is a constant. Using the second term of Eq. (5), we adjust the aligning center $\mathbf{c}$ according to the regressed center $\mathbf{c}^r(\mathbf{L})$ of parts. This helps resist imperfectly regressed part centers and locate the aligning centers within part regions, making alignment more reliable. Θ, $\mathfrak{A}$, and $\mathfrak{T}$ define the ranges of parameters. A large value from Eq. (5) indicates reliable alignment. Maximizing Eq. (5) is achieved by searching the quantized parametric space.

3.3.2 Valve Linkage Function (VLF)

Our VLF defines the output of the alignment sub-network, which is important to link the sub-networks and make them work as a whole in training and testing. It is expressed as

$$P(\mathbf{L}; \mathbf{I}, \mathbf{L}_f) = \frac{E_a(\mathbf{c}^*, \theta^*, \alpha^*, \mathbf{t}^*; \mathbf{I}, \mathbf{L})}{E_a(\mathbf{c}^*, \theta^*, \alpha^*, \mathbf{t}^*; \mathbf{I}, \mathbf{L}_f)} \phi(\mathbf{c}^*, \theta^*, \alpha^*; \mathbf{I}), \quad (6)$$

where

$$\begin{aligned} &\{\mathbf{c}^*, \theta^*, \alpha^*, \mathbf{t}^*\} = \arg\max_{\mathbf{c},\theta,\alpha,\mathbf{t}} E_a(\mathbf{c}, \theta, \alpha, \mathbf{t}; \mathbf{I}, \mathbf{L}_f), \\ &s.t.\ \ \mathbf{c} \in [x_1, x_2] \times [y_1, y_2], \theta \in \Theta, \alpha \in \mathfrak{A}, \mathbf{t} \in \mathfrak{T}. \end{aligned} \quad (7)$$

Here $\phi(\mathbf{c}^*, \theta^*, \alpha^*; \mathbf{I})$ is the pose-aligned part and $\mathbf{L}_f$ is the output of localization sub-network in the current forward-propagation (FP) stage. The role of valve function in FP and BP is discussed below.

FP stage In the FP stage of the neural network, alignment sub-network receives part location $\mathbf{L}_f$ and aligns it as $P(\mathbf{L}_f; \mathbf{I}, \mathbf{L}_f)$ for further classification. The output is expressed as

$$P(\mathbf{L}_f; \mathbf{I}, \mathbf{L}_f) = \phi(\mathbf{c}^*, \theta^*, \alpha^*; \mathbf{I}), \quad (8)$$

which is exactly the pose-aligned part.

BP stage In the BP stage, the output of alignment sub-network $P(\mathbf{L};\mathbf{I},\mathbf{L}_f)$ becomes a function of $\mathbf{L}$. Therefore, the objective function of LAC model is formulated as

$$J(\mathbf{W}_c,\mathbf{W}_l;\mathbf{I},\mathbf{L}^{gt},y^{gt}) = E_c(\mathbf{W}_c;P(\mathbf{L};\mathbf{I},\mathbf{L}_f),y^{gt}) + E_l(\mathbf{W}_l;\mathbf{I},\mathbf{L}^{gt}), \quad (9)$$

where $\mathbf{W}_c$ and $\mathbf{W}_l$ are the parameters to be determined. E_c and E_l are defined in two other sub-networks. We minimize this objective function to update localization and classification sub-networks during training.

To update the classification sub-network, we compute the gradients of objective function J with respect to $\mathbf{W}_c$. It is the same as those presented in [13].

To update the localization sub-network, gradients with respect to $\mathbf{W}_l$ are computed, written as

$$\nabla_{\mathbf{W}_l} J = \frac{\partial E_l}{\partial \mathbf{W}_l} + \frac{\partial E_c}{\partial \mathbf{W}_l} = \frac{\partial E_l}{\partial \mathbf{W}_l} + \frac{\partial E_c}{\partial P(\mathbf{L};\mathbf{I},\mathbf{L}_f)} \frac{\partial P(\mathbf{L};\mathbf{I},\mathbf{L}_f)}{\partial \mathbf{L}} \frac{\partial \mathbf{L}}{\partial \mathbf{W}_l}, \quad (10)$$

where the former term $\frac{\partial E_l}{\partial \mathbf{W}_l}$ represents the BP stage within localization.

Analysis In the second term of Eq. (10), $\frac{\partial E_c}{\partial P(\mathbf{L};\mathbf{I},\mathbf{L}_f)}$ and $\frac{\partial \mathbf{L}}{\partial \mathbf{W}_l}$ pass useful information in the BP stages within classification and localization sub-networks respectively. Without the valve linkage function part $\frac{\partial P(\mathbf{L};\mathbf{I},\mathbf{L}_f)}{\partial \mathbf{L}}$, information propagation from classification to localization would be blocked.

We further show that VLF provides information control from classification to other sub-networks. In the BP stage, $P(\mathbf{L};\mathbf{I},\mathbf{L}_f)$ can be rewritten as

$$P(\mathbf{L};\mathbf{I},\mathbf{L}_f) = \frac{1}{e} E_a(\mathbf{c}^*,\theta^*,\alpha^*,\mathbf{t}^*;\mathbf{I},\mathbf{L})\phi(\mathbf{c}^*,\theta^*,\alpha^*;\mathbf{I}), \quad (11)$$

where $e = E_a(\mathbf{c}^*,\theta^*,\alpha^*,\mathbf{t}^*;\mathbf{I},\mathbf{L}_f)$ is the alignment energy generated in FP stage. With it becoming a constant in backward propagation, $\frac{\partial P(\mathbf{L};\mathbf{I},\mathbf{L}_f)}{\partial \mathbf{L}}$ can be expressed as

$$\frac{\partial P(\mathbf{L};\mathbf{I},\mathbf{L}_f)}{\partial \mathbf{L}} = \frac{1}{e}\phi(\mathbf{c}^*,\theta^*,\alpha^*;\mathbf{I})\frac{\partial E_a}{\partial \mathbf{L}}. \quad (12)$$

And the term $\frac{\partial E_a}{\partial \mathbf{L}}$ is extended to

$$\frac{\partial E_a}{\partial \mathbf{L}} = -\frac{\lambda}{2}\exp(-\frac{1}{2}\|\mathbf{c}-\mathbf{c}^r(\mathbf{L})\|)\frac{\partial \|\mathbf{c}-\mathbf{c}^r(\mathbf{L})\|^2}{\partial \mathbf{L}}, \quad (13)$$

where $\mathbf{c} = (c_x, c_y)$ and

$$\frac{\partial \|\mathbf{c}-\mathbf{c}^r(\mathbf{L})\|^2}{\partial \mathbf{L}} = (\frac{x_1+x_2}{2}-c_x, \frac{y_1+y_2}{2}-c_y, \frac{x_1+x_2}{2}-c_x, \frac{y_1+y_2}{2}-c_y). \quad (14)$$

Here, factor $\frac{1}{e}$ can be deemed as a valve controlling influence from classification. As described in Section 3.3.1, a larger alignment score e corresponds to better alignment in the FP stage. In BP stage, $\frac{1}{e}$ is used to re-weight the BP error $\frac{\partial E_c}{\partial P(\mathbf{L};\mathbf{I},\mathbf{L}_f)}$ from classification. It functions as a compromise between classification and alignment errors.

In this case, a large e means good alignment in the BP stage, for which information from the classification sub-network is automatically reduced given a small $\frac{1}{e}$. In contrast, if e is small, current alignment becomes less reliable. Thus more classification information is automatically introduced by the large $\frac{1}{e}$ to guide $\mathbf{W}_l$ update. Simply put, one can understand $\frac{1}{e}$ as a dynamic learning rate in the BP stage. It is adaptive to matching performance.

With this kind of auto-adjustment mechanism in our VLF connecting classification and alignment, localization can be refined in the BP stage. We verify the powerfulness of this design in experiments.

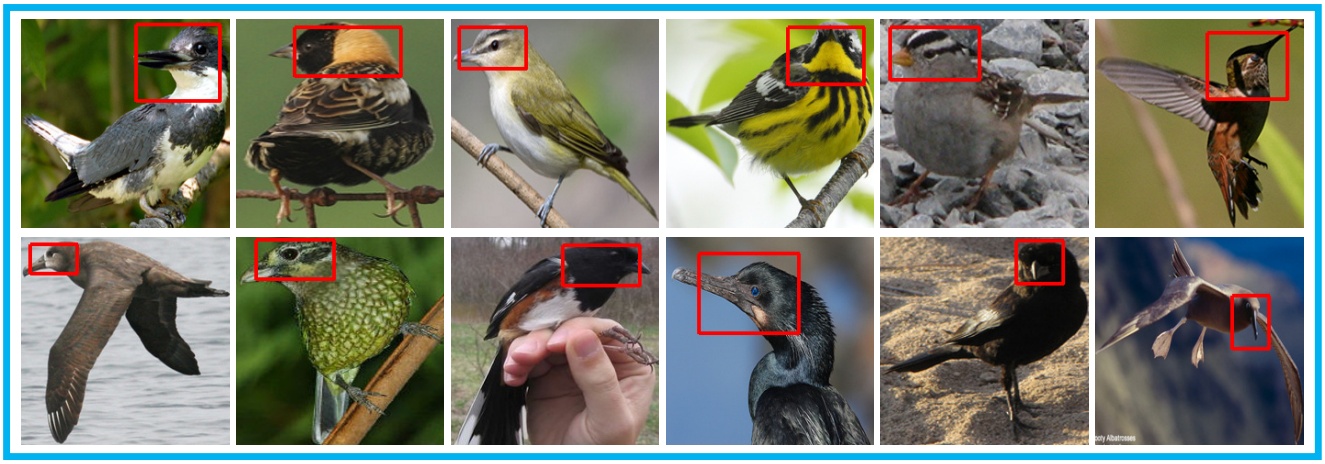
4. Experiments

We evaluate our method on two widely employed datasets: 1) the Caltech-UCSD Bird-200-2011 [22] and 2) Caltech-UCSD Bird-200-2010 [23].

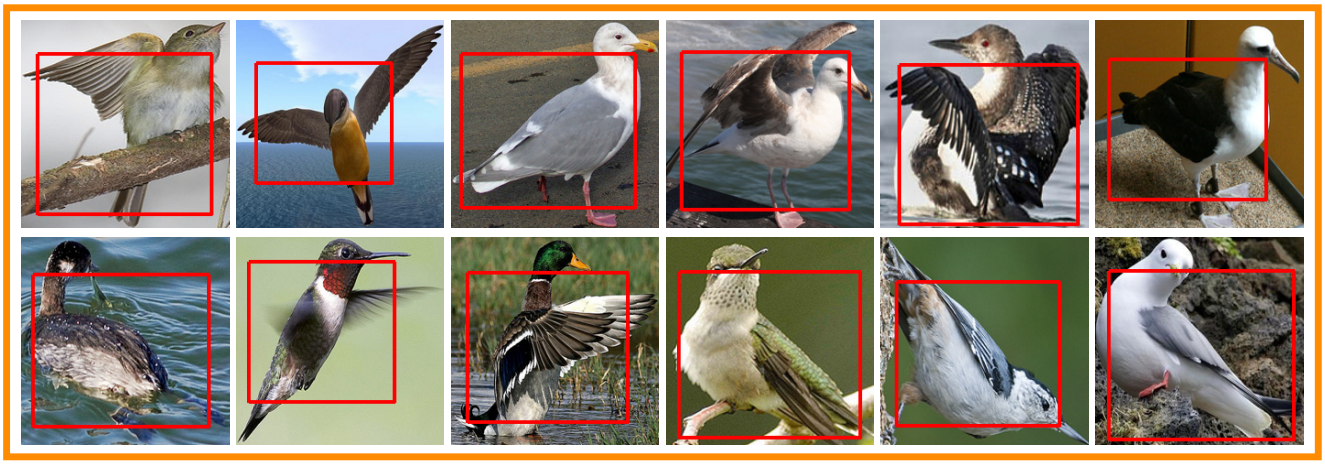
In implementation, we modify the Caffe platform [11] for CNN construction. Bird heads and bodies are considered as semantic parts. We train two deep LACs for them respectively. All CNN models are fine-tuned using the pre-trained ImageNet model. The 6^{th} layer of the CNN classification models (i.e., two part models + one whole image model) is extracted to form a $4096 \times$ 3D feature. Then we follow the popular CNN-SVM scheme [19] to train a SVM classifier on our CNN feature.

The major parametric setting for each part model is as follows. 1) In the localization sub-network, all input images are resized to 227×227. We replace the original 1,000-way fully connected layer with a 4-way layer for regressing part bounding box. The pre-trained ImageNet model is used to initialize our localization sub-network. 2) For alignment, in template selection, all 5,994 part annotations for head or body in the training set of Caltech-UCSD Bird-200-2011 [22] are used. The 5,994 parts are cropped and resized to 227×227. Using spectral clustering, we obtain the 5,994-part split into 30 clusters. From each cluster, we select the part region closest to the cluster center and its mirrored version as two templates. This process forms 60-template $\mathfrak{T}$ eventually.

During template alignment, the rotation degree θ is an integer and its range is $\Theta = [-60, 60]$. Meanwhile, we search the scale α within $\mathfrak{A} = \{2.5, 3, 3.5, 4, 4.5\}$. Another controllable parameter in alignment is λ in Eq. (5). Empirically, we set it to 0.001. Finally the classification sub-network takes input images each with size 227×227.



(a)



(b)

Figure 5: Localization examples of (a) bird head and (b) bird body.

Methods	Head	Body
(Strong DPM) [27, 2]	43.49%	75.15%
(Selective Serach) [27, 21]	68.19%	79.82%
Ours	**74.00%**	**96.00%**

Table 1: Comparison with state-of-the-arts in terms of part localization accuracy on part overlap ≥ 0.5 with ground truth on the CUB-200-2011 dataset.

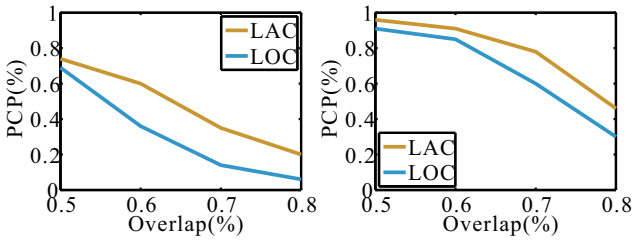
The last fully connected layer is a 200-way one since the two datasets contain 200 categories. Again, the pre-trained ImageNet model is used to initialize weight parameters.

The major computing hardware is a Nvidia TITAN Z graphics card with 5,760 cores and 12GB memory.

4.1. Caltech-UCSD Bird-200-2011 Dataset

We first evaluate our method on Caltech-UCSD Bird-200-2011 [22]. This dataset contains 11,788 images of birds, divided into 200 subordinate categories. Each image is labeled with its species and with the bounding box for the whole bird. It also provides annotations of bird parts including back, beak, belly, breast, crown, forehead, left eye, left leg, left wing, nape, right eye, right leg, right wing, tail, and throat.

During the training and testing phases, we make use of the bounding box provided in the dataset to simplify classification, as most previous methods did [14, 3, 5, 8, 28, 27]. Our experiments follow the training/testing split fixed in [22]. We define two kinds of semantic templates, i.e., "head" and "body", as in [27, 4]. Because there is no such annotation, we follow the method of [27] to get corresponding rectangles covering annotated parts distributed within



(a) Bird Head (b) Bird Body

Figure 6: PCPs of (a) bird head and (b) bird body under different overlap rates. "LAC" and "LOC" refer to the LAC model and localization sub-networks respectively.

bird heads and bodies.

The first experiment is to evaluate part localization. Two previous methods [21, 2] localize heads and bodies. With the same experimental setting, we make a comparison in Table 1 based on Percentage of Correctly Localized Parts (PCP) [27], which is computed on the top-ranked part prediction and regards parts with ≥ 0.5 overlap with ground truth as correct.

For the head parts, our result is 74.00% against previous 43.49% [2] and 68.19% [21]. For bird bodies, our accuracy is as high as 96.00% compared to the previous best 79.82%. Figure 5 shows a few examples for part localization, where (a) and (b) visualize the predicted bounding boxes of bird heads and bodies. Our localization sub-network is intriguingly beneficial to bounding box regression.

Localization Module Analysis To further understand the importance of the localization module using our VLF, we move this sub-network out of the joint LAC model and compare it with our overall LAC in terms of PCP with overlaps ≥ 0.5, 0.6, 0.7 and 0.8. The results are plotted in Figure 6. In all configurations, the localization network (LOC) alone performs notably worse than applying the whole LAC model. It is because LOC does not get feedback from alignment and classification while our LAC updates all of them in iterations using our FP and BP processes. Experimental results match our understanding of the network structure.

Sub-network Combination Analysis Our above experimental results manifest that LAC with all three sub-networks is powerful in part localization. We also evaluate its performance in fine-grained classification and experiment with removing one or two components in the following four cases.

First, we remove the localization sub-network by validating the classification accuracy on whole images without localization. The results are listed in the first row of Table 2. Without the localization module, the whole-image classification accuracy is 65.00%.

Methods	Head	Body
Case 1	65.00%	**65.00%**
Case 2	67.83%	43.00%
Case 3	70.00%	48.00%
Case 4	**72.00%**	52.65%

Table 2: Classification accuracy of semantic parts, i.e. head and body, on CUB-200-2011 dataset. We respectively block localization and alignment sub-networks to evaluate performance.

Second, we block the alignment sub-network to interdict FP and BP in the LAC framework. The localization sub-network is used to propose part hypotheses for classification. The remaining localization and classification modules are trained independently in BP stages. The values in the second row of Table 2 indicate that lack of message propagation in alignment is not recommended.

Third, we use VLF in the alignment sub-network to output pose-aligned part for classification in the FP stage. But VLF is disabled in the BP stage to prevent classification and alignment errors from back propagation to localization. In this case, we boost the accuracy to 70.00% on bird heads (the third row of Table 2). The alignment sub-network, along with FP and BP, is thus necessary.

Finally, we use the complete LAC model. It yields the best score 72.00% for head recognition in Table 2, bearing out that our VLF-involved LAC is suitable for fine-grained recognition and actually improves both classification and localization.

We observe about 34% performance gap in the four cases (65.00% vs 43.00%) for classifying the body part. Compared to the high PCP (96.00%) for body localization in Table 1, we conclude that bird bodies are not that distinct for bird species identification. The localization errors are however low. After adding alignment (with VLF), the performance gain is about 22% (52.65% vs 43.00%).

Overall Comparison Our final classification accuracy compared with other state-of-the-arts is presented in Table 3. All results are accomplished under the setting that the bounding box for the entire bird is given in training and testing. In our system, we feed each image into the two trained networks to extract features of head and body.

Table 3 shows using the features of head and body achieve accuracies 72.00% and 52.65%. We concatenate the two feature vectors to form a combined representation. It yields accuracy 78.12%. We finally tune the CNN model based on the whole image using the pre-trained model [11]. The 6^{th} layer of it is extracted for training a SVM classifier, obtaining accuracy 65.00%. After concatenating the

Methods	Accuracy
Lee et al. [14]	41.01%
Berg et al. [3]	56.89%
Goering et al. [9]	57.84%
Chai et al. [5]	59.40%
Gravves et al. [8]	62.70%
Zhang et al. [28]	64.96%
Zhang et al. [27]	76.37%
Ours (head)	72.00%
Ours (body)	52.65%
Ours (head+body)	78.12%
Whole image	65.00%
Ours (head+body) + whole image	**80.26%**

Table 3: Comparison with state-of-the-arts on the CUB-200-2011 dataset.

Methods	Head	Body
No localization	48.00%	**48.00%**
Localization	50.48%	40.71%
Localization & alignment	**54.00%**	45.12%

Table 4: Classification accuracy of semantic parts, i.e. head and body, on CUB-200-2010 dataset. We respectively block localization and alignment sub-network for performance evaluation.

features of head, body and the whole image, our accuracy increases to 80.26%. In comparison, the method of Zhang et al. [27] also considers the same head and body parts and combines the CNN feature of the whole image. We believe our accuracy increase is mainly due to reliable localization and alignment in the VLF-enabled LAC.

4.2. Caltech-UCSD Bird-200-2010 Dataset

Caltech-UCSD Bird-200-2010 [23] provides 6,033 images from 200 bird categories. It does not offer part annotation and contains less training/testing data, It thus can verify whether our LAC, which is trained on Caltech-UCSD Bird-200-2011, is able to be generalized to this dataset or not.

The classification results are listed in Table 4. The localization and alignment sub-networks are obtained using the training data from Caltech-UCSD Bird-200-2011. The classification sub-network is updated on this dataset after getting the pose-aligned part images.

Our whole-image classification accuracy (in the "No localization" row) is 48.00%. By localization of bird heads, the performance gain is about 5% (50.48% vs 48.00%). The gain is up to 12.5% after further incorporating alignment. The best body recognition accuracy 45.12% is achieved by adding localization and alignment.

Methods	Accuracy
Yao et al. [26]	19.20%
Khan et al. [12]	22.40%
Yang et al. [25]	28.20%
Angelova et al. [1]	30.20%
Deng et al. [6]	32.80%
Goering et al. [9]	35.94%
Farrell et al. [7]	37.12%
Chai et al. [5]	47.30%
Ours (head)	54.00%
Ours (body)	45.12%
Ours (head+body)	58.67%
Whole image	48.00%
Ours (head+body) + whole image	**65.25%**

Table 5: Comparison with the state-of-the-arts on CUB-200-2010 dataset.

In the final experiment, we make comparison with other methods in terms of classification accuracies. The results are tabulated in Table 5. The previous best result is 47.30% [5]. Our bird-head representation obtains 54.00% accuracy. The combined head and body representations yield 58.67% accuracy.

Similar to previous experiments, we also take the whole image into consideration. After combining all three features, our classification performance is boosted to 65.25% – i.e., 32% accuracy increase. We believe better performance can be achieved if the localization and alignment sub-networks are adapted with part annotation, which is however not available for this dataset.

5. Concluding Remarks

We have presented a deep neural network to achieve fine-grained recognition. We share the same observation as previous work that proper localization and alignment of salient object parts are important in this process. Based on it, we contribute a unified LAC system to incorporate localization, alignment and classification as three sub-networks. They are connected with an optimally defined VLF to enable smooth forward- and backward-propagation. Our results show this process improves part finding and matching, which eventually helps classification.

Acknowledgements

This work is supported by a grant from the Research Grants Council of the Hong Kong SAR (project No. 413113). The TITAN Z graphics card used for this project was donated by the NVIDIA Corporation.

References

[1] A. Angelova and S. Zhu. Efficient object detection and segmentation for fine-grained recognition. In *CVPR*, 2013.

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

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

[4] S. Branson, G. Van Horn, S. Belongie, and P. Perona. Bird species categorization using pose normalized deep convolutional nets. *arXiv*, 2014.

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

[6] J. Deng, J. Krause, and L. Fei-Fei. Fine-grained crowdsourcing for fine-grained recognition. In *CVPR*, 2013.

[7] 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. In *ICCV*, 2011.

[8] E. Gavves, B. Fernando, C. Snoek, A. Smeulders, and T. Tuytelaars. Fine-grained categorization by alignments. In *ICCV*, 2013.

[9] C. Goering, E. Rodner, A. Freytag, and J. Denzler. Nonparametric part transfer for fine-grained recognition. In *CVPR*, 2014.

[10] Y. Gong, L. Wang, R. Guo, and S. Lazebnik. Multi-scale orderless pooling of deep convolutional activation features. *ECCV*, 2014.

[11] Y. Jia. Caffe: An open source convolutional architecture for fast feature embedding. `http://caffe.berkeleyvision.org/`, 2013.

[12] F. S. Khan, J. Weijer, A. D. Bagdanov, and M. Vanrell. Portmanteau vocabularies for multi-cue image representation. In *NIPS*, 2011.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[14] Y. J. Lee, A. A. Efros, and M. Hebert. Style-aware mid-level representation for discovering visual connections in space and time. In *ICCV*, 2013.

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

[16] A. Y. Ng, M. I. Jordan, Y. Weiss, et al. On spectral clustering: Analysis and an algorithm. *NIPS*, 2002.

[17] J. Ngiam, Z. Chen, D. Chia, P. W. Koh, Q. V. Le, and A. Y. Ng. Tiled convolutional neural networks. In *NIPS*, 2010.

[18] J. P. Pluim, J. A. Maintz, and M. A. Viergever. Mutual-information-based registration of medical images: a survey. *Medical Imaging*, 2003.

[19] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. *arXiv*, 2014.

[20] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv*, 2013.

[21] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. *IJCV*, 2013.

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

[23] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-ucsd birds 200. Technical report, California Institute of Technology, 2010.

[24] L. Xie, Q. Tian, S. Yan, and B. Zhang. Hierarchical part matching for fine-grained visual categorization. In *ICCV*, 2013.

[25] S. Yang, L. Bo, J. Wang, and L. G. Shapiro. Unsupervised template learning for fine-grained object recognition. In *NIPS*, 2012.

[26] B. Yao, A. Khosla, and L. Fei-Fei. Combining randomization and discrimination for fine-grained image categorization. In *CVPR*, 2011.

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

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