Recognizing Trees at a Distance with Discriminative Deep Feature Learning

Zhen Zuo[1]
[1]School of Electrical and Electronic Engineering
Nanyang Technological University
Singapore
Email: zzuo1@e.ntu.edu.sg

Gang Wang[1,2]
[2]Advanced Digital Sciences Center
University of Illinois
Singapore
Email: wanggang@ntu.edu.sg

***Abstract*—We investigate discriminative features that are able to improve classification accuracy on visually similar classes. To this end, we build a deep feature learning network, which learns features with discriminative constraint in each single layer module, and learns multiple levels of features for hierarchical image representation. Specifically, the network encodes the discriminative information by automatically selecting the informative features, and forcing them to be closer to the features extracted from the same class than the features from different classes.**

We also collect a new fine-grained dataset containing 51 common tree species in Singapore. All the images are taken at a distance with large intra class variance, which makes the tree species hard to be distinguished. Our experimental results show that we are able to achieve 78.03% in accuracy on this challenging dataset, which is 8.48% higher than general hand-designed feature.

I. Introduction

Image classification has rapidly moving onto datasets with more categories and more realistic images in the recent decade. However, most datasets focused on a wide variety of basic level classes, such as Caltech-101 [1]. In this paper, we investigate a fine-grained tree species recognition problem, which is much more challenging than the traditional image classification. As shown in Figure 1, because of the large intra class variance and relatively small inter class variance, some species are even difficult for humans to recognize.

What distinguishes one tree species from another can be subtle clues hidden in local image areas. To capture such clues, we start from the very beginning of the image classification system: learning discriminative feature descriptors.

Many famous off-the-shelf features (such as SIFT [2] and HOG [3]) have achieved great success in traditional image recognition. However, they also have their limitations: 1) they are not adaptive to data; 2) they can hardly capture the discriminative characteristics of different classes. In contrast, deep feature learning techniques such as hierarchical spatial-temporal feature [4], sparse auto-encoder [5], convolutional deep belief network [6], and sum-product network [7], have shown outstanding performance on many challenging visual recognition tasks. Nonetheless, existing deep learning methods mainly focused on general feature representation. Not much attention has been paid on discriminative information, which is very crucial in fine-grained image classification.

To utilize discriminative information, we aim to learn features that are more similar to the features extracted from the same class than those from other classes. However, for fine-grained image recognition, local features from the same object category are highly diverse, while features from different categories can be very similar. Thus, strictly forcing features from the same class to be similar is improper. Therefore, we adopt Naive Bayes Nearest Neighbour (NBNN) [8] to directly calculate the distance between each feature and different categories. It is very suitable for our feature learning scheme, but it has only be used as a feature-based image classifier.

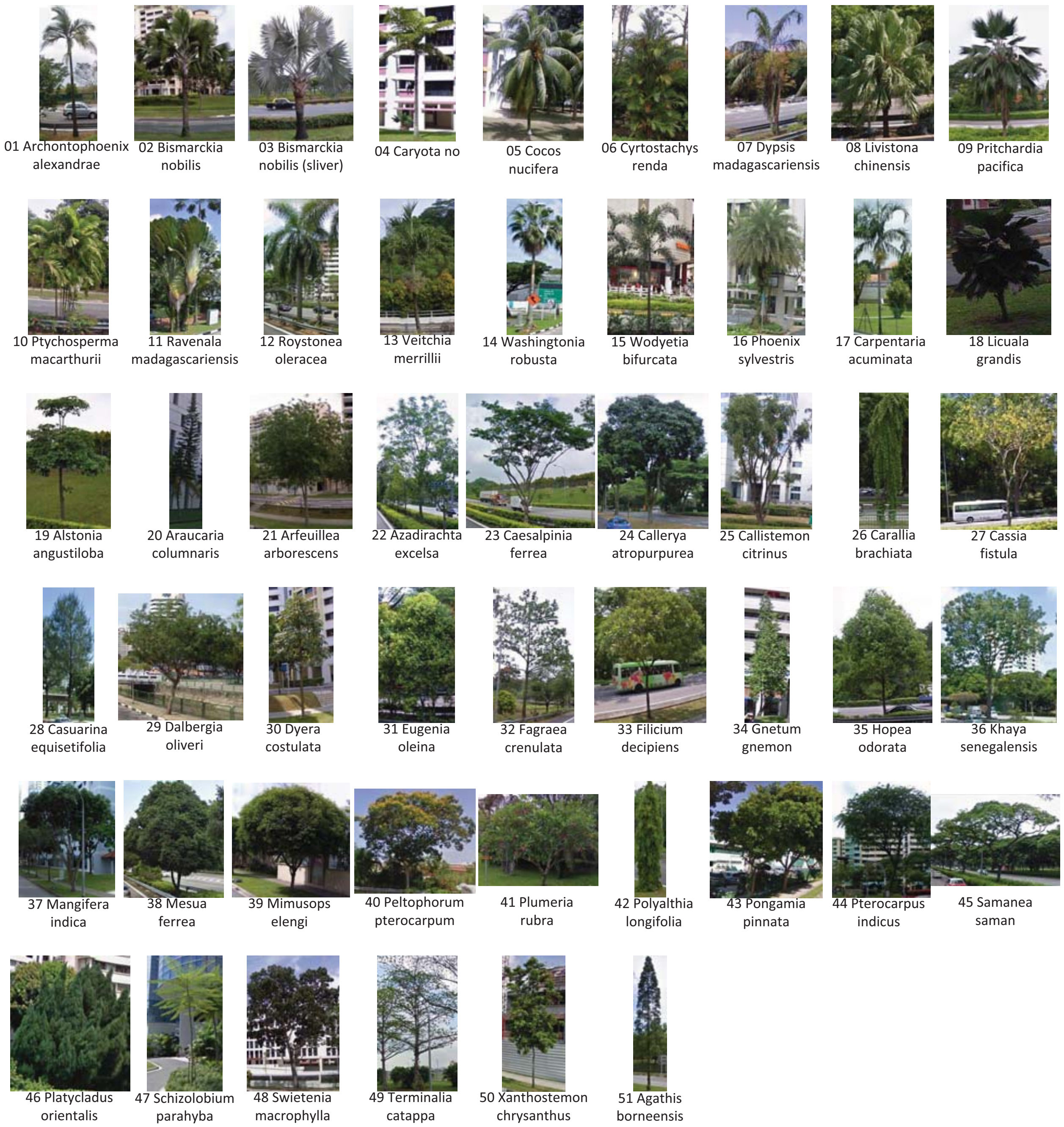
In this paper, we propose a discriminative deep feature learning method, which encodes the discriminative information into deep learning framework.

II. NTU Tree-51 dataset

Tree species recognition is a topic of enormous interest in environmental and ecological sciences. Previously, tree species identification is only conducted by professional botanists. Due to the scarce of experts, it is usually slow, labour intensive, and even prohibitive in cost. Recently, Belhumeur et al. [9] develop a working computer vision system that aids in the identification of tree species. However, it requires a user to photograph an isolated leaf on a blank background, and then the system extracts the leaf shape for recognition. This system is not applicable for large scale survey (e.g., recognizing all the trees in a city), because one must grab a leaf from each tree. And large scale survey of tree species is very useful for detecting problematic invasive alien species at the early stage, biodiversity assessment, urban planning, etc. To overcome the limitation of [9] and to enable large scale survey, in this paper, we propose to recognize trees at a distance. Ideally, we can take pictures of trees on a moving vehicle (similar to the Google Street View cars), and recognize tree species from these pictures, which is fast and cost-efficient.

In this paper, we construct a dataset containing 51 common tree species in Singapore. We get tree images from the Google Street View images, which are captured at a distance on a

This research is supported by the Singapore National Research Foundation under its IDM Futures Funding Initiative and administered by the Interactive & Digital Media Programme Office, Media Development Authority.



01 Archontophoenix alexandrae
02 Bismarckia nobilis
03 Bismarckia nobilis (sliver)
04 Caryota no
05 Cocos nucifera
06 Cyrtostachys renda
07 Dypsis madagascariensis
08 Livistona chinensis
09 Pritchardia pacifica
10 Ptychosperma macarthurii
11 Ravenala madagascariensis
12 Roystonea oleracea
13 Veitchia merrillii
14 Washingtonia robusta
15 Wodyetia bifurcata
16 Phoenix sylvestris
17 Carpentaria acuminata
18 Licuala grandis
19 Alstonia angustiloba
20 Araucaria columnaris
21 Arfeuillea arborescens
22 Azadirachta excelsa
23 Caesalpinia ferrea
24 Callerya atropurpurea
25 Callistemon citrinus
26 Carallia brachiata
27 Cassia fistula
28 Casuarina equisetifolia
29 Dalbergia oliveri
30 Dyera costulata
31 Eugenia oleina
32 Fagraea crenulata
33 Filicium decipiens
34 Gnetum gnemon
35 Hopea odorata
36 Khaya senegalensis
37 Mangifera indica
38 Mesua ferrea
39 Mimusops elengi
40 Peltophorum pterocarpum
41 Plumeria rubra
42 Polyalthia longifolia
43 Pongamia pinnata
44 Pterocarpus indicus
45 Samanea saman
46 Platycladus orientalis
47 Schizolobium parahyba
48 Swietenia macrophylla
49 Terminalia catappa
50 Xanthostemon chrysanthus
51 Agathis borneensis

Fig. 1. The NTU tree-51 dataset. Each image in an instance of a tree species.

moving vehicle and are aligned well with our purpose. We use the guide of wayside trees in Singapore [10] to help labelling, which not only shows how different trees look like at a distance, but also shows where we can find them in Singapore. At the end, we cropped 2613 street view images to construct our tree dataset. This dataset, named NTU Tree-51, consists of 51 species, each of which contains 30-70 samples. It contains large viewpoint, scale, and illumination variations. Images for the dataset are shown in Figure 1.

III. Approach

We build a discriminative deep feature learning framework for recognizing trees at a distance. To achieve this goal, we embed the discriminative information into the deep Independent Subspace Analysis (ISA) learning network [4], which is a very effective and efficient learning framework. This combination is especially appropriate for fine-grained image classification. The learned features can not only capture the most discriminative class-specific information in each sub-

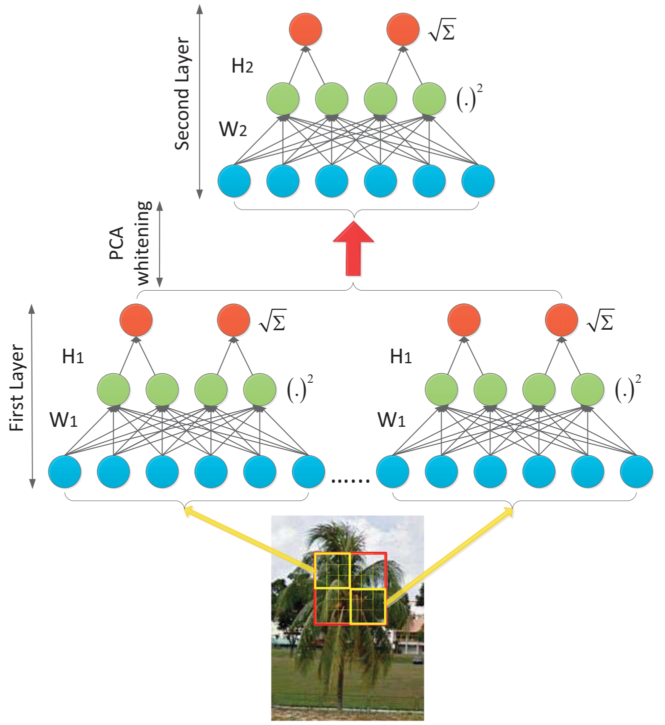


Fig. 2. The deep ISA feature learning framework. In the first layer, features are learned from small image patches (yellow boxes). In the second layer, features are learned from larger image patches (red box), whose inputs can be achieved by concatenating the over-complete first layer features extracted within a larger image area (red box), and then applying PCA for dimension reduction and whitening. In each single ISA learning module, the blue units are input data, the green units are the squared linear transformed data, while the red units are the output learned features after applying energy pooling. (Best viewed in color.)

category class, but also be able to produce multi-levels of image representations.

A. Deep ISA framework

We firstly briefly introduce the basic single layer ISA learning module, and then describe how to stack multi-layer ISA to get hierarchical feature representation.

For each single layer ISA structure, as shown in Figure 2, given an original input data x, it first performs a linear transformation Wx. Then it applies an non-linear energy pooling procedure with matrix H to combine the adjacent elements without overlapping. The output is the single layer feature:

$$q = \sqrt{H(Wx)^2} \tag{1}$$

in which, W need to be learned while H is usually fixed.

To avoid degenerating risk of the transformation matrix W, we adopt the reconstruction constraint described in [11], which can be optimized with very fast convergence rates. Then we can learn the transformation matrix W in the following way:

$$\min_{W} \sum_{j=1}^{m} \left(\left\| x_j - W^T W x_j \right\|_2^2 + \gamma q_j \right) \tag{2}$$

Because the learned single layer features are extracted from local image areas, they cannot represent higher level visual information, and they might also bring noise. To get multi-level and robust feature representation, deep feature learning is introduced. In the deep ISA network, they built a two-layer framework, and connect the two layers in a convolution and stacking manner similar to convolutional neural network [12]. As shown in Figure 2, in the first layer, small (16x16) image patches are sent as input. In the second layer, by applying convolution, an over-complete set of first layer features are extracted in larger (32x32) image areas. After applying PCA for dimension reduction, we get the inputs to the second layer. Concatenating the learned features learned by both layers, we get the final multi-layer features.

B. Discriminative term

Traditional general hand-crafted features or feature learning algorithms can hardly capture the most discriminative information in different but highly visually similar image categories. While for the fine-grained image classification problem, different categories may share most of the visual information. Thus, a discriminative information selection procedure is needed.

We build our discriminative term in the following probabilistic way: for a local feature q, we aim to maximizing the posterior of q conditioned on its corresponding class c, while minimizing the posterior of q conditioned on all the other classes $\bar{c}$:

$$\max \frac{P\left(c|q\right)}{P\left(\bar{c}|q\right)} \tag{3}$$

For simplification, we assume the class prior $P\left(c\right)$ is uniform, then the posteriors reduce to likelihood:

$$\frac{P\left(c|q\right)}{P\left(\bar{c}|q\right)} = \frac{P\left(q|c\right)}{P\left(q|\bar{c}\right)} \tag{4}$$

We adopt the NBNN distance measurement as described in [8] to approximate these likelihoods. By applying the Parzen window estimator, $P\left(q|c\right)$ can be succinctly represented as:

$$P\left(q|c\right) = \exp\left(-\|q - NN_c\left(q\right)\|^2\right) \tag{5}$$

in which, $NN_c\left(q\right) = \sqrt{H(Wx^{NN})^2}$ is the nearest neighbour of q from class c in the feature space, and x^{NN} is its corresponding original input data. In this paper, to accelerate the nearest neighbour searching procedure, we employ FLANN [13], which is a library making use of multiple randomized K-D trees to achieve fast nearest neighbour searching approximation.

By bringing Equation 4 and Equation 5 into Equation 3, and taking logarithm, we get our discriminative term:

$$\min \left(\|q - NN_c\left(q\right)\|^2 - \|q - NN_{\bar{c}}\left(q\right)\|^2 \right) \tag{6}$$

in which, $NN_c\left(q\right)$ denotes the nearest neighbour of q from its corresponding class c, and $NN_{\bar{c}}\left(q\right)$ denotes the nearest neighbour of q from classes other than c.

Robust discriminative feature selection. The discriminative term can be unreliable if we randomly pick features extracted from training images, and directly use them in Equation 6. Since in the fine-grained image recognition task, most of the patch information are not distinguishable among several highly similar classes, and the background areas also bring noisy information. To make the discriminative feature learning procedure more robust, we add a threshold to Equation 6. We don't select q for training if $\|q - NN_c(q)\|^2$ is similar to or larger than $\|q - NN_{\bar{c}}(q)\|^2$. Since it means that q is a common feature shared among several classes, or a background feature that is not class-representative, thus, q is not that informative and should be ignored.

C. *Discriminative deep feature learning*

Combining our discriminative information selection constraint with the deep ISA feature learning network, we get our discriminative feature learning framework. In which, the single layer module learning method is shown as following:

$$\min_{W} \sum_{j=1}^{m} \left(\left\| x_j - W^T W x_j \right\|_2^2 + \gamma q_j \right) + \eta \sum_{i=1}^{n} E_s(q_i)$$

$$E_s(q_i) = \begin{cases} \|q_i - NN_c(q_i)\|^2 - \|q_i - NN_{\bar{c}}(q_i)\|^2 & \text{if} E_s(q_i) < \delta \\ 0 & \text{else} \end{cases} \tag{7}$$

where γ and η are regularization parameters, δ is the selection threshold, their values are decided by doing cross validation. As the discriminative term is much more informative, in our experiment, we select smaller number of training patches (small m) for the first part of Equation 7, while employ much more densely extracted training patches (large n) for the second part.

However, in Equation 7, the procedure of searching nearest neighbours for all the training features is time consuming. To accelerate the optimization procedure, we do an approximation with the following three steps: 1) learning an initialized transformation matrix W by using the original ISA feature learning scheme, as shown in Equation 2; 2) calculating the nearest neighbours x^{NN} of all the training patches x in the feature space, which is determined by the W learned in the previous step; 3) fixing the memberships of the nearest neighbours x^{NN} , and updating W by optimizing the complete deep ISA framework with our discriminative term, as shown in Equation 7. Repeating step 2) and 3) for several times can slightly improve the performance, while for efficiency consideration, we merely do these steps for once in our experiments.

IV. Experiments

In this section, we present the experiment results on our NTU tree-51 dataset, which contains 51 common wayside tree species in Singapore, 2613 images in total, and 30-70 images per category. We use 20 images per category for training, and the rest for testing. To make a better comparison with previous algorithms, we only use gray-scale images, and resize all the images to 150x150 unified size.

Algorithm	Accuracy
SPM (SIFT) [17]	69.55%
deep ISA [4]	70.94%
Our method without feature selection	**75.39%**
Our method with feature selection	**78.03%**

TABLE I
Results on NTU tree-51 dataset.

In the first layer, we input 16x16 training patches, and learn 128 dimensional features. In the second layer, we convolve the learned first layer features in 32x32 patch areas with a stride of 2. Then we concatenate the responses and apply PCA to get 300 dimensional input for the second layer, and learn 150 dimensional features.

A. *Preprocessing for discriminative term*

For our discriminative term, the accuracy of the searched nearest neighbours is the key point of good performance. According to previous works based on NBNN [14], [15], we adopt the following three processes to improve the accuracy:

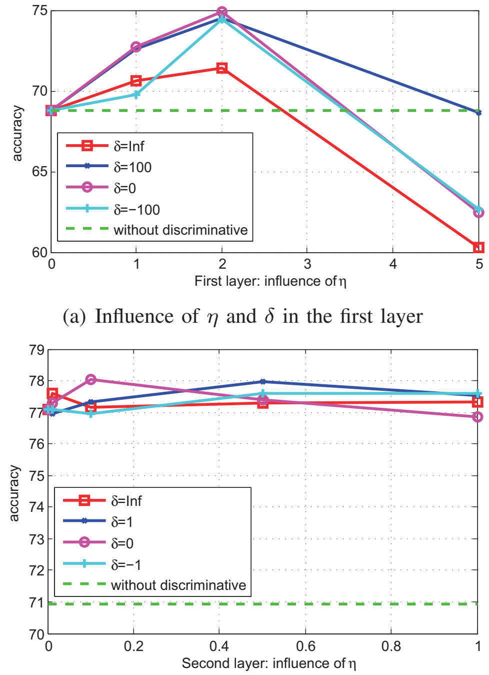
Densely extracting training patches. In our experiment, for each training image, we densely extract 1,500 patches per training image for the first layer, and 300 patches per training image for the second layer to build the class-specific nearest neighbour patch searching pools. However, assigning all the training patches as training patches, and finding their corresponding nearest neighbours is too time consuming. Thus, to reach a compromise between efficiency and accuracy, we apply an asymmetric training model. We merely randomly pick 150 patches per image in the first layer, and 30 patches in the second layer as training query patches, while keep the original size of the nearest neighbour searching pools.

Adding rough spatial information. We introduce spatial information in the similar way as described in [16]. We separate all the training images into 2x2 non-overlapping spatial blocks, and label the extracted training patches with block numbers to indicate their rough spatial location in the original images. When doing nearest neighbour searching for each training patch, we only search among patches with the same block labels, which can not only improve searching accuracy, but also speed up the searching procedure.

Removing possible background patches. We discard the patches extracted from low contrast areas. Since such areas are usually background or uninformative areas, which is not helpful to learn discriminative and robust features.

B. *Results on NTU tree-51*

We test different features on our NTU tree-51 dataset with a two layer spatial pyramid matching scheme. Numerical results of our method compared to other methods are shown in Table I. Our method outperforms the original deep ISA algorithm with 7.09%, and outperforms the SIFT descriptor with 8.48% in accuracy. The comparison results indicate three things: 1) features learned by deep network is able to compete with hand-designed features; 2) our discriminative feature



(a) Influence of η and δ in the first layer

(b) Influence of η and δ in the second layer

Fig. 3. Influence of η and δ in both layers.

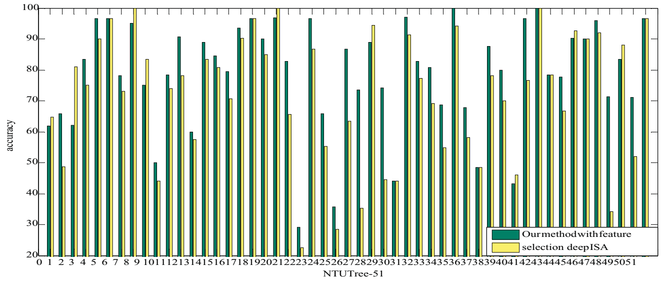


Fig. 4. Per-category classification accuracy of our NTU Tree-51 dataset. The numbers on x axis denote the index numbers of different tree species

learning method can improve fine-grained image classification accuracy; and 3) the feature selection constraint can help to learn more robust features.

C. Discussion

To quantitatively analyze the influences of η and δ in both layers, we vary their values and get the testing results as shown in Figure 3. In both layers, our discriminative feature selection term do help to improve performance with appropriate η and δ. Furthermore, comparing the results of our method with feature selection ($\delta < \mathrm{Inf}$) with our method without feature selection ($\delta = \mathrm{Inf}$), especially in the first layer, we can observe an obvious improvement in robustness and accuracy increasing.

The detailed per-category classification accuracy is shown in Figure 4. Our method outperforms the original deep ISA algorithm in 34 categories and equal performs in 8 categories. Our method get worse than deep ISA in 9 categories, most of these species are highly confusing ones, which make our discriminative feature learning not accurate.

V. Conclusion

We have shown that by combining discriminative information with the deep ISA feature learning framework, we can significantly improve the performance of recognizing similar tree species at a distance. To extract such discriminative information, we force the training features to be more similar to the nearest neighbour features searched from the same class, while be less similar to the nearest neighbour features searched from the other classes. Inserting this discriminative constraint to deep ISA framework, our discriminative deep feature learning method can not only encode class-specific information, but also represent multi-layer visual information.

References

[1] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," *Computer Vision and Image Understanding*, 2007.

[2] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, 2004.

[3] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005.

[4] Q. V. Le, W. Y. Zou, S. Y. Yeung, and A. Y. Ng, "Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis," in *CVPR*, 2011.

[5] Q. V. Le, M. A. Ranzato, R. Monga, M. Devin, K. Chen, G. S. Corrado, J. Dean, and A. Y. Ng, "Building high-level features using large scale unsupervised learning," in *ICML*, 2012.

[6] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *ICML*, 2009.

[7] R. Gens and P. Domingos, "Discriminative learning of sum-product networks," in *NIPS*, 2012.

[8] O. Boiman, E. Shechtman, and M. Irani, "In defense of nearest-neighbor based image classification," in *CVPR*, 2008.

[9] P. Belhumeur, D. Chen, S. Feiner, D. Jacobs, W. J. Kress, H. Ling, I. Lopez, R. Ramamoorthi, S. Sheorey, S. White, and L. Zhang, "Searching the world's herbaria: A system for visual identification of plant species," in *ECCV*, 2008.

[10] Y. C. Wee, *A Guide to the Wayside Trees of Singapore*. Singapore Science Centre, 1989.

[11] Q. V. Le, A. Karpenko, J. Ngiam, and A. Y. Ng, "Ica with reconstruction cost for efficient overcomplete feature learning," in *NIPS*, 2011.

[12] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, 1995.

[13] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in *VISSAPP*, 2009.

[14] T. Tuytelaars, M. Fritz, K. Saenko, and T. Darrell, "The nbnn kernel," in *ICCV*, 2011.

[15] S. McCann and D. G. Lowe, "Local naive bayes nearest neighbor for image classification," in *CVPR*, 2012.

[16] Z. Wang, Y. Hu, and L. T. Chia, "Image-to-class distance metric learning for image classification," in *ECCV*, 2010.

[17] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *CVPR*, 2006.