**Stage-1 Experiment Results**

**1 Evaluation dataset and Metric**

* + **Oxford Flower 102**

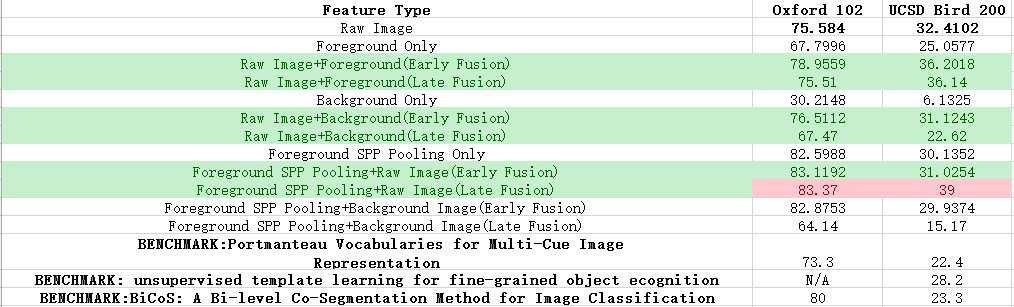
The oxford flower 102 consists of 102 flower categories, each class consists of between 40 and 258 images. Each class have been split into training set, validation set and test set, respectively. Besides, the segmentation results using BiCoS approach are also provided in [1] .

* + **UCSD Bird 200**

The recently published Caltech-UCSD Birds 200 contains 200 bird categories and 6033 images in total. The segmentation result using BiCoS approach, set splits and labels are offered [1].

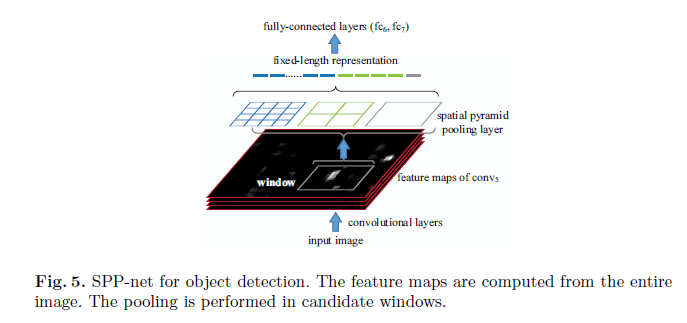
In the following experiments, all the training data, validation data and test data and other irrelevant parameters are as same as that mentioned in [2]

1. **Experiment Result**



(see .xls file for more detail)

* All experiments are conducted under pre-trained [imagenet-vgg-m-1024](http://www.vlfeat.org/matconvnet/models/imagenet-vgg-m-1024.mat) net framework [3].
* For feature types with raw **image/foreground only/background only**, we directly send the image into the CNN, and get the last fully connected layer (1024-d) as output feature, and then train linear SVMs to conduct final classification task.
* For feature types with two features, the two kind of images are send into a same CNN and extract individual features, respectively. Later, we employee early fusion strategy ( concatenate two kinds of features directly followed by training a uniform SVM) or late fusion(train two independent SVM for two kinds of features and average weighted voting at prediction stage) to compare the performance.
* SPP Pooling: Similar with [4], carry on spatial pyramid pooling after the 5th convolutional layer with a recurrent depth at three. Thereby, the feature dimension is (16+4+1)\*512=10752 dim. For the sake of the fact that the later fine-tuning work of two fully connected layers requires large amount of training data to avoid over-fitting issue, we **directly utilize the 10752 dim raw feature as the SPP-Pooling feature.**
* ==========The following part detailed describes the input features========
* **Raw Image input**: resize to fixed size 224\*224, feed to CNN and get 1024-dim feature.
* **Foreground**: using BiCoS segmentation result(binary image, 0 is background, 1 is foreground) to bitwise multiply with the raw image(get foreground), and the background part are filled with mean image.
* **Background**: using inverse-BiCoS segmentation result(binary image, 1 is background, 0 is foreground) to bitwise multiply with the raw image(get background), the foreground part are filled with mean image whereas the background.
* **Foreground SPP:** first get the circumscribed rectangle of the foreground part; send the whole image into CNN, **get SPP feature in the circumscribed rectangle** on the 5th convolution layer. Similar with the following figure:



1. **Experiment Observations**

* Using raw image directly with CNN feature can achieve the benchmark results on two datasets.
* The best performances on both datasets are achieved with **Foreground SPP Pooling+Raw Image(Late Fusion)** strategy(improving the performance on flower and bird dataset by 3.37% and 15.7%, respectively)
* Nevertheless, apart from combinations of **Foreground SPP+Raw Image** , early fusion strategy outperforms the late counterpart in other feature combination types.
* Although the **background Image+Foreground SPP pooling** combination seems more reasonable than others, its actual performance is not as well as we have expected.

**4 Further work(valse会议后初步讨论的结果)**

* Parameter examination on the late fusion voting weight, manually find the weight between background information and foreground information.
* 左老师的观点：我们拿到foreground SPP feature（10752维）和raw/background feature（1024维度）以后，做三件事：1. 分别训练1v1线性SVM(和**之前一样，训练完就不变了)**； 2. 对于每个线性的1v1 SVM，根据foreground feature和raw/background image feature，再学习一个回归的模型（比如logistic regression？保证输出范围在0到1之间），对于每个输入图片的SPP feature和raw image feature可以得到一个对应的weight theta(0<theta<1，也就是对前景SPP feature分类结果的confidence index)；3.根据theta和学习到的两个线性model去做vote，得到最终的decision\_value(1v1 SVM中，大于0为正样本，小于0为负样本)

Decision\_value=theta\*decision\_value\_SPP+(1-theta)\*decision\_value\_raw\_image.

（当decision\_value=0.5时，和之前的average voting相同）

**Mathematic form:**

**Minimize the following objective function:**

 %FontSize=10
%TeXFontSize=10
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\min _{w1,b1,w2,b2,\theta}\left\{ \sum _{i}\left[ 1-y_{i}\theta(x_{i}) \left( w_{1}x_{1,i}+b_{1}\right) -\left( 1-\theta(x_{i}) \right) y_{i}\left( w_{2}x_{2,i}+b_{2}\right) \right]_{+} +\lambda _{1}\left\| w_{1}\right\| ^{2}+\lambda _{2}\left\| w_{2}\right\| ^{2}\right\} 
\]
\end{document}

Where:

Model’s Inputs：

* is the foreground feature of the i-th image.
* is the raw image feature of the i-th image.

%FontSize=10.5
%TeXFontSize=10.5
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
x_{i}=(x_{1,i},x_{2,i})
\]
\end{document} is the concatenate version of two features.

* is the label of the i-th image.(1/-1)

%FontSize=10.5
%TeXFontSize=10.5
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\theta(x_{i})
\]
\end{document} is the weight of two kinds of feature. %FontSize=10.5
%TeXFontSize=10.5
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
0<\theta(x_{i})<1
\]
\end{document}

%FontSize=10.5
%TeXFontSize=10.5
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
w_{1},b_{1},w_{2},b_{2}
\]
\end{document} are the ordinary linear SVM parameters for foreground feature and background feature, respectively.

**Modelling of %FontSize=10.5
%TeXFontSize=10.5
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\theta(x_{i})
\]
\end{document}:**

**Remaining Question : How to modelling theta?(E.g. simple logistic model? OR just training theta as a constant,类似孟老师optimal pooling中的theta参数)**

**Algorithm:**

STEP 1: Solve the common linear SVM model for foreground image and raw/background image independently, and acquire %FontSize=10.5
%TeXFontSize=10.5
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
w_{1},b_{1},w_{2},b_{2}
\]
\end{document} (with %FontSize=11
%TeXFontSize=11
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\theta
\]
\end{document} fixed at 0.5 for all input features)

%FontSize=10
%TeXFontSize=10
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\min _{w1,b1}\left\{ \sum _{i}\left[ 1-y_{i}\left( w_{1}x_{1,i}+b_{1}\right)\right]_{+} +\lambda _{1}\left\| w_{1}\right\| ^{2}\right\} 
\]
\end{document}

%FontSize=10
%TeXFontSize=10
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\min _{w2,b2}\left\{ \sum _{i}\left[ 1-y_{i}\left( w_{2}x_{2,i}+b_{2}\right)\right]_{+} +\lambda _{2}\left\| w_{2}\right\| ^{2}\right\} 
\]
\end{document}

STEP 2: minimize over theta:

%FontSize=10
%TeXFontSize=10
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\min _{\theta}\left\{ \sum _{i}\left[ 1-y_{i}\theta(x_{i}) \left( w_{1}x_{1,i}+b_{1}\right) -\left( 1-\theta(x_{i}) \right) y_{i}\left( w_{2}x_{2,i}+b_{2}\right) \right]_{+} \right\} 
\]
\end{document}

**References**

1. BiCoS: Segmentations for Flower Image Datasets and Others <http://www.robots.ox.ac.uk/~vgg/data/bicos/>
2. Yuning, C., et al. (2011). "BiCoS: A Bi-level co-segmentation method for image classification." 2579-2586.
3. Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Return of the Devil in the Details: Delving Deep into Convolutional Nets. Proceedings of the British Machine Vision Conference. BMVA Press, September 2014.
4. He, K., et al. (2015). "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition." IEEE Transactions on Pattern Analysis and Machine Intelligence: 1-1