Project: Classification of the remote sensing data set

Data set: the NWPU aerial data set contains approximately 45 categories of 700 images for each category in 256x256 RGB format. The data set can be downloaded from:

<https://umkc.box.com/s/fxvzh5qq2tiob6eklfxfwn89kg3e1io1>

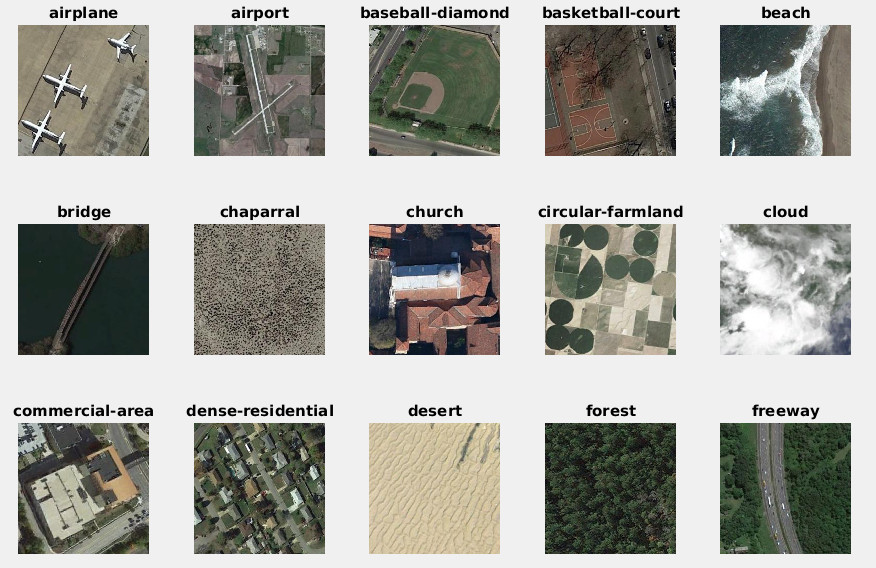


Figure 1. Examples of the images and labels

More details of the dataset can be found at:

<http://www.escience.cn/people/JunweiHan/NWPU-RESISC45.html>

For the project, we only deal with the first 15 categories shown in Fig. 1 here. The data set should be partitioned into training (500 images), validation (100 images), and test (100 images) for each category.



fc1

fc2

fc3

Option A: Use pretrained VGG16 network final conv pooling features before fc1, in this case 512 7x7 feature maps, and use Fisher Vector aggregation to do the classification :

1. [25pts] Compute the PCA and GMM models for the 49-dimension final conv features for kd=[16, 24] and number of components nc=[32, 64, 128] {Hint, refer to HW-2 for solutions} , show your code and results here.
2. [75pts] For each of the 15 categories, determine the optimal subset of dimensions, and components to use in 1-vs-rest classification, e.g, for an kd=16, nc=64 FV, we can choose to turn off certain components in GMM to maximize the classification accuracy, similar to the SCFV case covered in class (Lec 12). The basic idea is to examine the class specific component prior, and turn off ones that are too small. Plot your results.

Option B: Attention model driven classification with triplet loss. The baseline attention model can use that of Biren’s. This option will give you total of x1.5 credits for project.

1. (30pts) Label splitting: Use the best embedding from HW-4, e.g, fc3 with PCA+LDA, and then build a kd-tree {hint: vl\_feat has this function} to partition the training data set (500x15) into a L=7 depth kd-tree, with 128 leaf node. This kd-tree will give each image an appearance label P=[1..128], this in conjunction with the original 15 labels in Y, will give us a P x Y total 128x15 labels.
2. (40pts) Try softmax on an Attention Model VGG backbone to see if performance is good with the splitted lable.
3. (80pts) Develop a triplet loss network with expanded/splitted label, and see if we can out-perform the base line in HW-4.