City University of Hong Kong
EE4146 project Group 11

Abstract

In image classification tasks, convolutional neural networks have shown reliable results.This project presented principal component analysis and support vector machine as models for classifying natural scene photos. The performance and structure of different models, especially the modification of layer s components and parameters are the subject of this project. The performance of the model is evaluat ed by comparing the accuracy and error of the results of different models.

Introduction

In recent years, due to the progress of technology, the field of machine learning and scene analysis h as made great progress. The Image Classification problem involves assigning a label to a set of input images that come in a set of categories. In this project, 10500 photographs from six categories (inclu ding street, forest, building, sea, glacier and mountain) are used as the input dataset, with 2200 imag es serving as the testing dataset. When we trained data with different models from where the errors a nd scores were collected. Our analysis of classification performances is based on adjusting different parameters of each model. The objective is to develop a model that is higher in accuracy and lower i n errors.

Objective

- To classific 6 images using machine learning techniques such as Principal components analysis, s upport vector machine, Multilayer Perceptron.
- To Compare different machine learning techniques.

Methodology

A. Dataset

There are 10500 images of 6 categories of data, namely buildings, forest, glacier, mountain, sea and stree t, while there are 2200 images for testing accuracy with the algorithm.

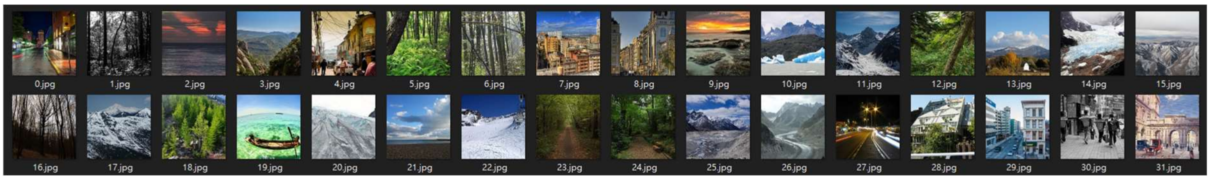

Fig 1. Example of training image

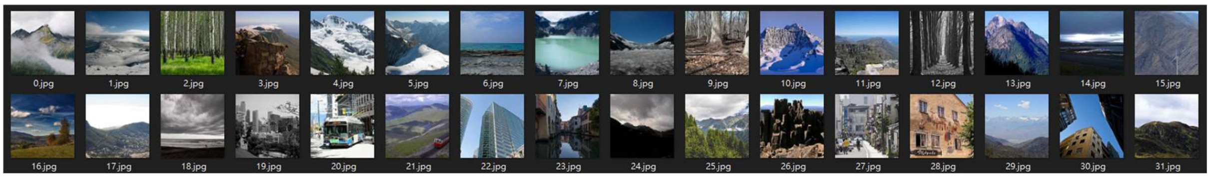

Fig 2. Example of testing image

B. Data Preprocessing

We found that the four classes "building , forest, glacier, sea and street" count of 2000 images but mountains only have 500 images. In order to improve the accuracy, we need to multiply the total number of the mountain with 4. The total number of mountains is equal to 2000 images such that the five categories have the same number.

1	label	Count
2	buildings	2000
3	forest	2000
4	glacier	2000
5	mountain	500
6	sea	2000
7	street	2000

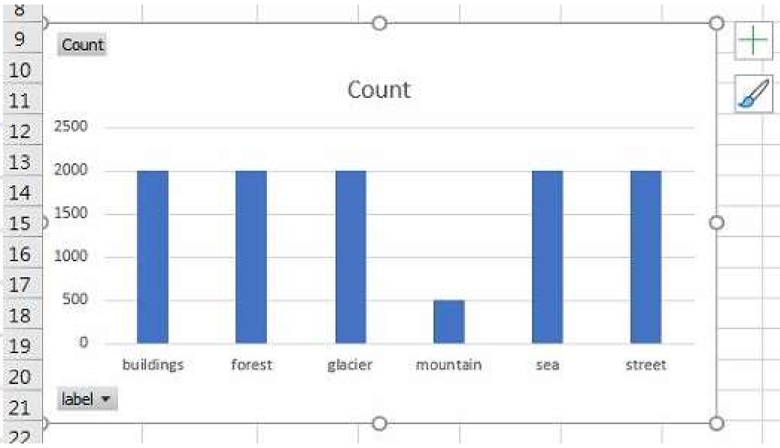

Figure 3. Classes Imbalanced

C. Classification Algorithms and dimension reduction

i.Principal components analysis(PCA)

An investigative data analysis using principal component analysis is used to establish the prediction models. The method is commonly used to reduce dimensionality by assuming only a few principal components of each data point to achieve lower-dimensional data while preserving as much variation as possible.

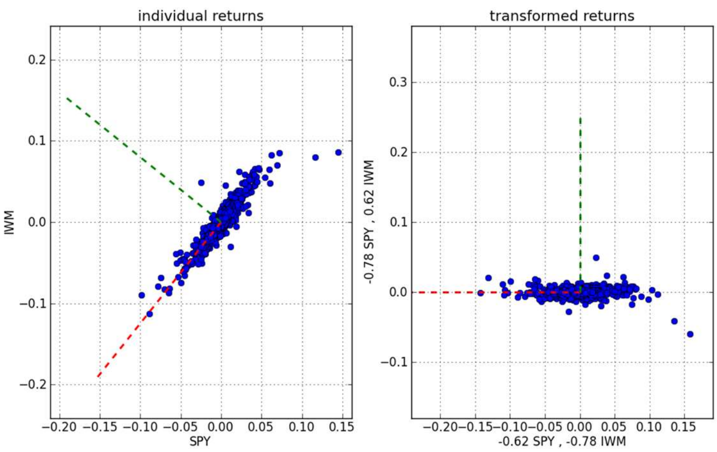

Figure 4. PCA

ii. Support vector machine(SVM)

SVM normally is a two-class classification model, whose basic model is a linear classifier with maximum i nterval defined on the feature space. The learning strategy of SVM is interval maximization. The SVM can only solve linearly differentiable problems.

To solve more complex problems, we can use Kernel SVM. When the training data is linearly inseparable, a nonlinear SVM can be learned by using the kernel trick. The graph below shows that we cannot classify r ed and green in lower dimensional space with linear SVM. After the projection, we can find a non-linear pr ojection to transform the data to a higher dimensional space. In this case, only one linear classifier is neede d to classify the data perfectly in the higher dimensional space.

We choose RBF kernel as our kernel to perform kernel SVM. It is because it has better performance on tes t data. The RBF kernel is a common kernel function used in SVM to map the input data to a high-dimensio nal feature space.

Then we adjusted the gamma and the C because they have a significant impact on the performance of the n onlinear SVM. The C parameter is a regularization parameter to prevent overfitting of misclassified data.

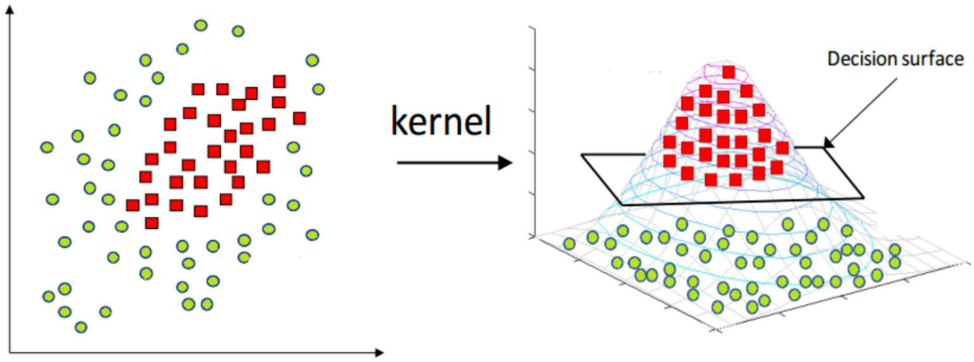

Figure 5. PCA kernel

D. Model Design

First of all, in the first approach, pca (Radial Basis Function Kernel) and svm (Radial Basis Function Kern el) are applied to extract features, and wide resnet 50 is used to extract features. These features provide bett er performance, with a verification rate of 93.3% and a score of 79% on leader board. Then we found that t he training data set is class imbalance. In the original training data set, there are 2000 pictures in each cate gory but only 500 pictures in Mountain. The class imbalance will affect the performance of mode. Therefor e, the pictures in the minority class are adjusted to the same number (500x4 -> 2000) as other classes such that there are a total 12,000 data features in the training data set.

```
#%%

# Inference and save features
label_save = []
feats_save = []
feat_save_path = 'train_feat3/'

if not os.path.exists(feat_save_path):
    os.makedirs(feat_save_path)

k = 10500

for images, labels, image_ids in train_loader:
    B, C, W, H = images.shape
    feat = model(images.cuda()).detach().cpu()
    feats_save.append(feat) # B 2024
    label_save.append(labels)
    for i, imgID in enumerate(image_ids):
        np.save(feat_save_path +'{}_feat_densenet.npy'.format(imgID), feat[i].numpy())
    if labels==3:
        for n in range(3):
            feats_save.append(feat)
            label_save.append(labels)
            for i, imgID in enumerate(image_ids):
                np.save(feat_save_path +'{}_feat_densenet.npy'.format(k), feat[i].numpy())
                k+=1

feats_save = torch.cat(feats_save,dim=0)
label_save = torch.cat(label_save)
```

Figure 6. Code

We tried the first approach with the balanced training set. The result is satisfactory with a score of 87% on the leaderboard.

```
pca = decomposition.KernelPCA(n_components=1024,kernel='rbf')
trainW = pca.fit_transform(trainX)  # fit the training set
valW  = pca.transform(valX)  # use the pca model to transform the test set
svmclf  =svm.SVC(kernel='rbf')
svmclf.fit(trainW, trainY)
predY_svm = svmclf.predict(valW)
acc_svm = metrics.accuracy_score(valY, predY_svm)
print("Kernel PCA svm validation accuracy =", acc_svm)

Kernel PCA svm validation accuracy = 0.9466666666666667
```

```
1 : 0.5876666666666667
2 : 0.804
4 : 0.8496666666666667
8 : 0.891
16 : 0.916
32 : 0.924
64 : 0.9316666666666666
128 : 0.9423333333333334
256 : 0.9446666666666667
512 : 0.9456666666666667
1024 : 0.9466666666666667
```

Figure 7. Code and training result

Moreover, we also investigated densenet201 as follows to compare the result.

```
#%%

# Define pretrained deep feature extractor
model = torchvision.models.wide_resnet50_2(pretrained=True).cuda()
# model = torchvision.models.densenet201(pretrained=True).cuda()
for param in model.parameters():
    param.required_grad = False

# Replacing the final classifier by identity layer (for directly out features)
model.fc = torch.nn.Identity()
model.eval()
```

```
1 : 0.31966666666666665
2 : 0.454
4 : 0.5053333333333333
8 : 0.5576666666666666
16 : 0.595
32 : 0.6346666666666667
64 : 0.636
128 : 0.6306666666666667
256 : 0.642
512 : 0.695
1024 : 0.7033333333333334
```

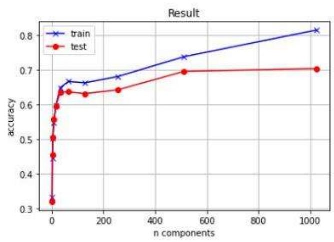

Figure 8. Training result

Since the result is unsatisfactory and worse than using wide resnet50 therefore we retain using wide resnet 50. Moreover, we also tried logistic regression, decision tree, ada boost, KNN, MLP, CV Ridge Classifier and Random forest. However, the majority of the methods did not perform well when compared to SVM. The highest accuracy is MLP except using SVM. Thus, we tried to compare two methods, which are a ML

P model with PCA kernel and a MLP model without PCA kernel. Since MLP could set activation function and solver, it could use grid search CV to find out the most matchable activation function and solver.

Results and Discussion

A. SVM model with Kernel PCA

In our SVM model, we set the kernel to be "RBF" and use GridSearchCV to estimate the best C and ga mma parameter in the range. GridSearchCV is guaranteed to find the most accurate parameter within a spe cified range of parameters. However, it requires traversing all possible combinations of parameters, which i s very time-consuming in the face of large data sets and multiple parameters. For linear SVM, the best C v alue is 10 and gamma is 1.0

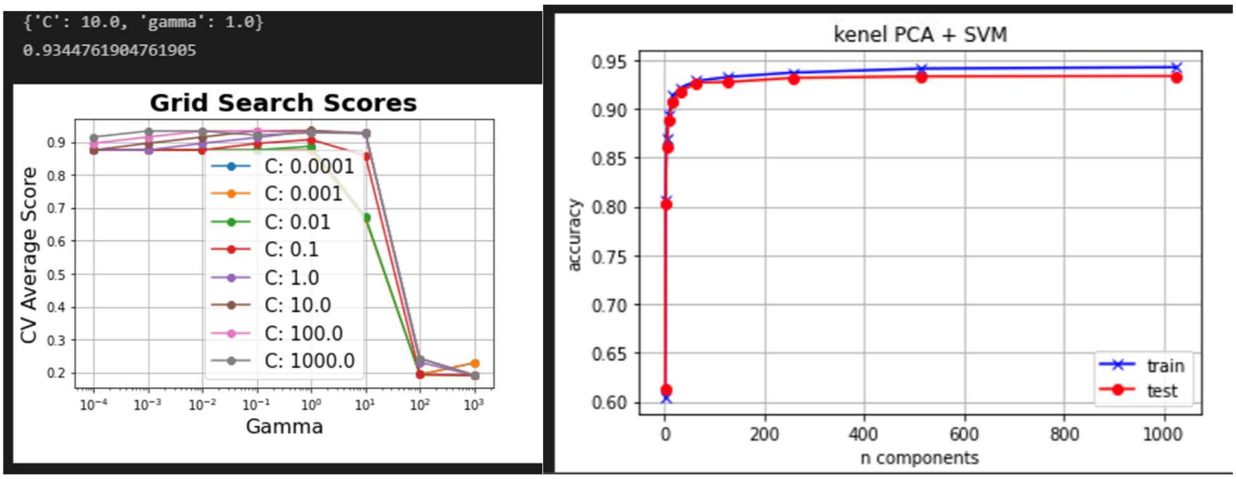

Figure 9. Using different PCA value

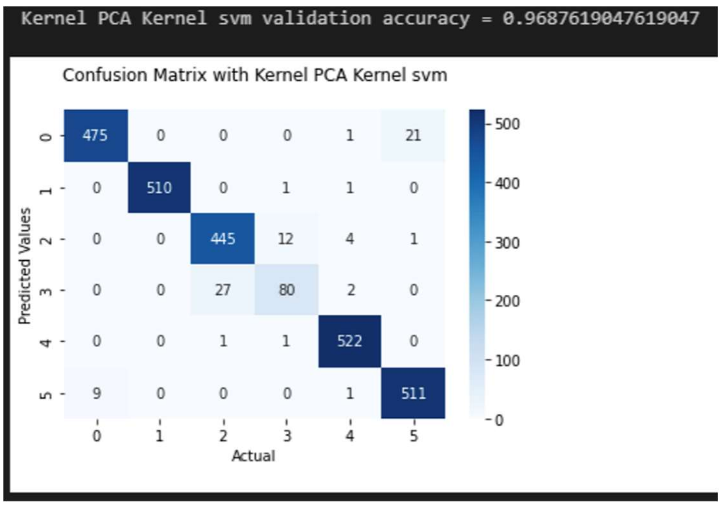

Figure 10. Confusion matrix: Kernel PCA Kernel SVM (n_components=512,kernel='rbf') (C: 10, gamma: 1, kernel=`rbf`)

B. MLP model without Kernel PCA

49
```
res = pd.DataFrame(clf.cv_results_)
res.drop(['std_fit_time','mean_score_time','std_score_time','params'], axis=1, inplace=True)
for i in range(5):
    res.drop([f'split{i}_test_score'],axis=1, inplace=True)
res
```

49

	mean_fit_time	param_activation	param_solver	mean_test_score	std_test_score	rank_test_score
0	105.193976	identity	lbfgs	0.925000	0.003865	12
1	903.009869	identity	sgd	0.931000	0.006059	9
2	210.052248	identity	adam	0.927333	0.006540	11
3	84.929987	logistic	lbfgs	0.936778	0.005060	6
4	928.710932	logistic	sgd	0.928111	0.004467	10
5	311.154912	logistic	adam	0.940222	0.006010	3
6	81.294620	tanh	lbfgs	0.934111	0.003906	7
7	862.449848	tanh	sgd	0.941222	0.005735	2
8	195.140477	tanh	adam	0.943000	0.003542	1
9	116.093033	relu	lbfgs	0.932444	0.004061	8
10	495.844654	relu	sgd	0.937667	0.004058	5
11	175.611701	relu	adam	0.938889	0.004856	4

Since MLP could set activation functions and solvers, A grid search CV had been used to test different activation functions and solvers. As the result, we found that tanh plus adam perform well in this situation. It had the highest accurate rate and was the most stable. However, it only reached 85% accurate rate in the project. Thus, we found that a SVM model with kernel PCA is the best solution.

Conclusion

The overall image classification is classified by the method, SVM model with PCA kernel. This data analysis used up a total of 10500 images to retrieve the results, which classifies the different types of images for 87.27 percent of accuracy. Most of the classes had 2000 images and only the mountain class had 500 images, which caused a class imbalance problem. To solve the problem, it needed to up-sample the mountain image in order to achieve a class balance.

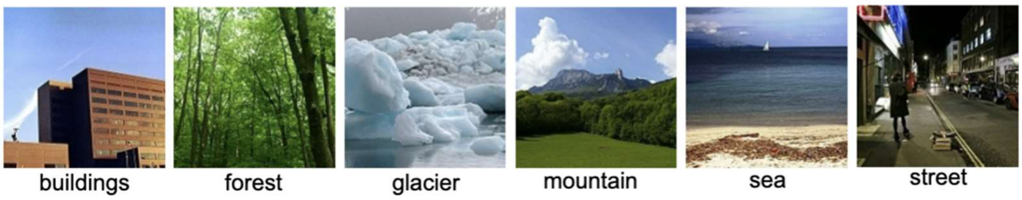

Figure 11. classification images

Based on the results of this project, we note being able to generate higher accuracy models, higher balanced accuracy, and a higher balanced detection rate with balanced data sets. Hence, it is important to have a balanced class for a classification model.

Reference

Wang, W., Zhang, M., Wang, D., & Jiang, Y. (2017, September 6). *Kernel PCA feature extraction and the SVM classification algorithm for multiple-status, through-wall, human being detection - EURASIP journal on Wireless Communications and networking*. SpringerOpen. Retrieved December 8, 2021, from https://jwcn-eurasipjournals.springeropen.com/articles/10.1186/s13638-017-0931-2.

Wikimedia Foundation. (2021, December 3). *Principal component analysis. Wikipedia. Retrieved December 8, 2021, from https://en.wikipedia.org/wiki/Principal_component_analysis.*