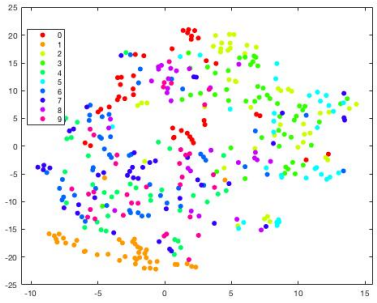
```
%% MATH 6380o. Deep Learning: Towards Deeper Understanding
%% By WU Aoyu

%% This project aims to perform classification task on MNIST dataset via
%% feature extraction by 1. Scatting Network and 2. AlexNet (Transfer
Learning)
%% For scatting network, default parameters are utilized.
%% For AlexNet, the 20th layer 'fc7' is extracted as feature. Meanwhile, the
original image is resized to 227*227 and converted to RGB images (the same
with that of AlexNet).
%% tSNE is used to visualize the extracted features.
%% Classification is performed by SVM.
%% The result suggests that both feature extraction methods produce relative
good results. Using AlexNet yields poorer result, probably because the pre-
trained convolutional neural network is trained on RGB images, while the
dataset in this project is grayscale. Directly converting grayscale images to
RGB might not be a good solution.

clear
%% Environment Set
% ScatNet (http://www.di.ens.fr/data/software/scatnet/license/)
path('scatnet-0.2',path);
addpath_scatnet;

%% Data Load
(http://ufldl.stanford.edu/wiki/index.php/Using_the_MNIST_Dataset)
TrainImages = loadMNISTImages('train-images-idx3-ubyte');
TrainLabels = loadMNISTLabels('train-labels-idx1-ubyte');
NTrain = length(TrainLabels);

TestImages = loadMNISTImages('t10k-images-idx3-ubyte');
TestLabels = loadMNISTLabels('t10k-labels-idx1-ubyte');
NTest = length(TestLabels);

%% Random Sample Data
% Training Data Size: 400
RandTrain = randperm(NTrain,400);
TrainImages = TrainImages(:,:,RandTrain);
TrainLabels = TrainLabels(RandTrain);

% Test Data Size: 100
RandTest = randperm(NTest,100);
TestImages = TestImages(:,:,RandTest);
TestLabels = TestLabels(RandTest);



%% Scattering Net
% Feature Extraction
ScatNetFeature = [];
for i = 1:length(TrainLabels)
    ScatNetFeature(i,:) = ExtractScatNet(getImageByIdx(TrainImages,i));
    if (mod(i,100)==0)
        fprintf('Scattering net feature extraction: %s. \n',i);
    end
end

% Visualize by tSNE
```

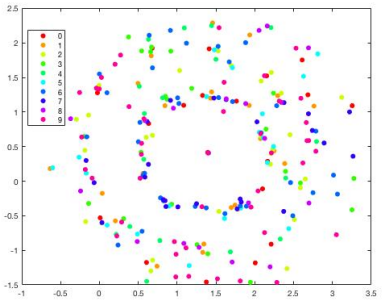
```
ScatNetY = tsne(ScatNetFeature);
figure;
gscatter(ScatNetY(:,1),ScatNetY(:,2),TrainLabels);
```



```
% Train
ScatNetModelSVM = fitcecoc(ScatNetFeature,TrainLabels);

% Test
runScatNetTest(TestImages,TestLabels,ScatNetModelSVM)
```

```
>> runScatNetTest(TestImages,TestLabels,ScatNetModelSVM)

ans =

    0.9000
```

```
%% Transfer Learning: AlexNet
% 227 * 227 * 3 * N
TrainImagesResized = [];
for i = length(TrainLabels)
    resized = imresize(getImageByIdx(TrainImages,i),[227 227]);
    TrainImagesResized(:,:,:,i) = cat(3,resized,resized,resized);
end

for i = length(TestLabels)
    resized = imresize(getImageByIdx(TestImages,i),[227 227]);
    TestImagesResized(:,:,:,i) = cat(3,resized,resized,resized);
end

net = alexnet;
layer = 'fc7';
AlexFeature = activations(net,TrainImagesResized,layer);

% Visualize by tSNE
AlexY = tsne(AlexFeature);
figure;
gscatter(AlexY(:,1),AlexY(:,2),TrainLabels);
```




```
% Train
ScatNetModelSVM = fitcecoc(AlexFeature,TrainLabels);

% Test
runAlexTest(TestImages,TestLabels,net,layer,ScatNetModelSVM)
```

```
>> runAlexTest(TestImages,TestLabels,net,layer,ScatNetModelSVM)

ans =

    0.8400
```