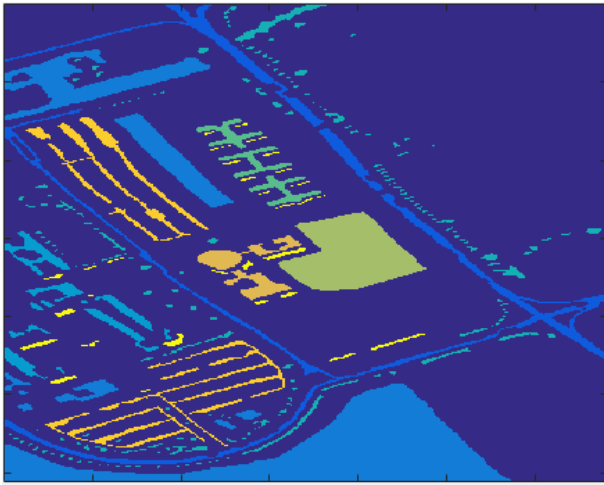


INTRODUCTION TO REMOTE SENSING

Trained KNN Classification on Hyperspectral Image

Emre ALABAY - 150207051

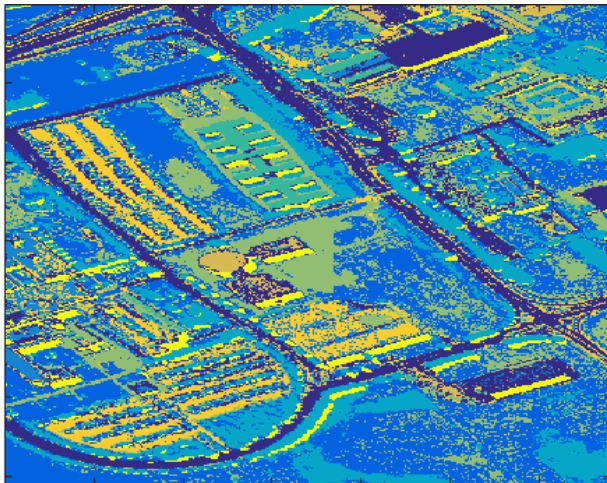
- The Groundtruth Image with the classes that we use to train KNN Classifier



-KNN Classified Hyperspectral Image

(Result for K=1 and Ratio(TrainData/Groundtruth)=0.3)

Success Rate of KNN Classification = %77.6019



	Ratio=0.02	Ratio=0.04	Ratio=0.06	Ratio=0.08	Ratio=0.1
K=1	%69.50	%70.53	%70.47	%72.01	%72.98
K=5	%70.73	%71.47	%70.95	%72.12	%73.04
K=9	%69.82	%71.62	%71.04	%71.96	%72.92
K=13	%69.76	%71.51	%71.10	%71.87	%72.81
K=17	%69.81	%71.32	%71.19	%71.83	%72.63

CODE

```

close all;
clear all;
knn = input('The number of K ?: ');
ratio = input('What is the ratio of TrainData/Groundtruth ?: ');

load('PaviaU.mat');%Hyperspectral
load('PaviaU_gt.mat');%Groundtruth

[spat1,spat2,spec]=size(paviaU);%Finding size of hyperspectral image
[row, col, val] = find(paviaU_gt);%Val=number of GT_points
Num_of_Classes = max(paviaU_gt(:));%Number of Classes

t=0;
for n=1:1:Num_of_Classes % Finding number of each class
    [row, col, val] = find(paviaU_gt == n);
    [Num_of_Train , one] = size(val);
    a=round(Num_of_Train*ratio);
    b=randperm(a,a);% Selecting random Train Points
    for i=1:1:a
        Train_Points(i+t,1)=row(b(i));% Storing row, col and tag of Train_Points
        Train_Points(i+t,2)=col(b(i));
        Train_Points(i+t,3)=paviaU_gt(row(b(i)),col(b(i)));
    end
    t=t+a;
end
% KNN CLASSIFICATION
distance=0;
counter=size(Train_Points,1); %Number of Total Train Data
neighbors=zeros(1,knn);
tagged=zeros(spat1,spat2); %Classified Image
for x=1:spat1 %Calculating Euclidean Distance
    for y=1:spat2
        for z=1:counter
            for band=1:spec
                distance = distance + (paviaU(x,y,band) -
paviaU(Train_Points(z,1),Train_Points(z,2),band))^2;
            end
            dist(z,1)=sqrt(distance);
            distance=0;
        end
        [v , index]=sort(dist(:,1));
        for k=1:knn
            neighbors(k)=Train_Points(index(k),3);
        end
        tagged(x,y)=mode(neighbors); % Selecting the most frequent class tag
    end
end
end

```

```

figure;
subplot(1,2,1); imagesc(tagged); title('KNN Classified Hyperspectral Image');
subplot(1,2,2); imagesc(paviaU_gt);title('Groundtruth');

%CALCULATING CLASSIFICATION SUCCESS RATE
true_positive=0;
false_positive=0;
for x=1:spat1
    for y=1:spat2
        if(paviaU_gt(x,y)~=0)
            if(tagged(x,y)==paviaU_gt(x,y))
                true_positive = true_positive+1;
            else
                false_positive = false_positive+1;
            end
        end
    end
end
success_rate= true_positive*100 / (true_positive + false_positive);
print=['Success Rate of KNN Classification = %',num2str(success_rate)];
disp(print)

```