

# BÀI TẬP THỰC HÀNH 4





## KHẢO SÁT BỘ DỮ KHÁC – VÍ DỤ KHUÔN MẶT

GV: Lê Đình Duy - Nguyễn Tấn Trần Minh Khang

Họ và tên: Nguyễn Anh Tú  
MSHV: CH1601020





- Q1 - Hãy viết function hiển thị ảnh có thứ tự là n (n là tham số) cùng label tương ứng trong tập huấn luyện (train) của tập dữ liệu THE DATABASE OF FACE. Paste code vào bài thực hành và lập bảng cho biết kết quả khi chạy với n=1, 50, 100, 200.

```
function showNImageTrain(n)
%Q1- Hien thi anh co thu tu n va label tuong ung tap Train
imgTrainAll = importdata('imgTrainImagesAll.mat');
lblTrainAll = importdata('lblTrainLabelsAll.mat');
imgTrain = imgTrainAll(:,n);
figure;
img2D = reshape(imgTrain,112,92);
imshow(img2D);
strLabelTrain = num2str(lblTrainAll(n));
title(strLabelTrain);
end
```

n	1	50	100	200
Image + Label	<div>1</div> 	<div>8</div> 	<div>15</div> 	<div>29</div> 

- Q2 - Hãy viết function hiển thị ảnh có thứ tự là n (n là tham số) cùng label tương ứng trong tập test của tập dữ liệu THE DATABASE OF FACE. Paste code vào bài thực hành và lập bảng cho biết kết quả khi chạy với n=1, 50, 100, 119.

```
function showNImageTest(n)
%Q2- Hien thi anh co thu tu n va label tuong ung tap Test
imgTestAll = importdata('imgTestImagesAll.mat');
lblTestAll = importdata('lblTestLabelsAll.mat');
imgTest = imgTestAll(:,n);
figure;
img2D = reshape(imgTest,112,92);
imshow(img2D);
strLabelTest = num2str(lblTestAll(n));
title(strLabelTest);
end
```

n	1	50	100	119
Image + Label	1 	17 	34 	40 

- Q3 - Hãy viết function thống kê số lượng các ảnh tương ứng với các label trong tập huấn luyện (train) của tập dữ liệu THE DATABASE OF FACE. Paste code vào bài thực hành và lập bảng kết quả khi chạy (nên xuất dưới dạng csv để tiện import thành bảng).

```
function countTrainImages()
%Q3- Thống kê số lượng ảnh tương ứng với các label trong tập Train
lblTrainAll = importdata('lblTrainLabelsAll.mat');
rArray = [];
imgNumber = 1;
numLabel = size(lblTrainAll, 2);
while(imgNumber<=40)
    countImgs = 0;
    for i = 1: numLabel
        if(lblTrainAll(i) == imgNumber)
            countImgs = countImgs + 1;
        end
    end
    rArray = [rArray,[imgNumber,countImgs]'];
    imgNumber = imgNumber+1;
end
csvwrite('Question3.csv',rArray);
end
```

1	7	11	7	21	7	31	7
2	7	12	7	22	7	32	7
3	7	13	7	23	7	33	7
4	7	14	7	24	7	34	7
5	7	15	7	25	7	35	7
6	7	16	7	26	7	36	7
7	7	17	7	27	7	37	7
8	7	18	7	28	7	38	7
9	7	19	7	29	7	39	7
10	7	20	7	30	7	40	7

- Q4 - Hãy viết function thống kê số lượng các ảnh tương ứng với các label trong tập test của tập dữ liệu THE DATABASE OF FACE. Paste code vào bài thực hành và lập bảng kết quả khi chạy (nên xuất dưới dạng csv để tiện import thành bảng).

```
function countTestImages()
%Q4- Thống kê số lượng ảnh tương ứng với các label trong tập Test
lblTestAll = importdata('lblTestLabelsAll.mat');
rArray = [];
imgNumber = 1;
numLabel = size(lblTestAll, 2);
while(imgNumber<=40)
    countImgs = 0;
    for i = 1: numLabel
        if(lblTestAll(i) == imgNumber)
            countImgs = countImgs + 1;
        end
    end
    rArray = [rArray, [imgNumber, countImgs]'];
    imgNumber = imgNumber+1;
end
csvwrite('Question4.csv', rArray);
end
```

1	3	11	3	21	3	31	3
2	3	12	3	22	3	32	3
3	3	13	3	23	3	33	3
4	3	14	3	24	3	34	3
5	3	15	3	25	3	35	3
6	3	16	3	26	3	36	3
7	3	17	3	27	3	37	3
8	3	18	3	28	3	38	3
9	3	19	3	29	3	39	3
10	3	20	3	30	3	40	3

- Q5 - Hãy viết function trả về kết quả nhận dạng của ảnh trong tập test có thứ tự là n (n là tham số, nằm trong đoạn [1, 10000]). Paste code vào bài thực hành và lập bảng kết quả khi chạy với n = 5, 50, 100.

```
function RecognitionTestImages(n)
%Q5 - Function trả về kết quả nhận dạng của ảnh trong tập Test có thứ
%tu n ([1,10000]).
imgTestAll = importdata('imgTestImagesAll.mat');
lblTestAll = importdata('lblTestLabelsAll.mat');
imgTrainImagesAll = importdata('imgTrainImagesAll.mat');
lblTrainLabelsAll = importdata('lblTrainLabelsAll.mat');
```

```

Mdl = fitcknn(imgTrainImagesAll',lblTrainLabelsAll);
imgTest = imgTestAll(:,n);
lblPredictTest = predict(Mdl,imgTest');
lblImageTest = lblTestAll(n);
fprintf('\nBan dau: %s',num2str(lblTestAll(n)));
fprintf('\nDu doan: %s',num2str(lblPredictTest));
if(lblPredictTest==lblImageTest)
    fprintf('\n Ket qua dung');
else
    fprintf('\n Ket qua sai');
end
end

```

<b>n</b>	<b>5</b>	<b>50</b>	<b>100</b>
<b>Ban đầu</b>	2	17	34
<b>Dự đoán</b>	2	17	34
<b>Kết quả</b>	Đúng	Đúng	Đúng

- Q6 - Hãy viết function với tham số đầu vào n là thứ tự của ảnh trong tập test - sau đó hiển thị ảnh tương ứng - rồi hiển thị kết quả nhận dạng - rồi cho biết kết quả nhận dạng là đúng hay sai khi so khớp với label của tập test.

```

function RecognitionShowTestImages(n)
%Q6 - Nhap n, hien thi anh va ket qua nhan dang
imgTestAll = importdata('imgTestImagesAll.mat');
lblTestAll = importdata('lblTestLabelsAll.mat');
imgTrainImagesAll = importdata('imgTrainImagesAll.mat');
lblTrainLabelsAll = importdata('lblTrainLabelsAll.mat');
Mdl = fitcknn(imgTrainImagesAll',lblTrainLabelsAll);
imgTest = imgTestAll(:,n);
lblPredictTest = predict(Mdl,imgTest');
lblImageTest = lblTestAll(n);
figure;
img2D = reshape(imgTest,112,92);
imshow(img2D);
strLabelImage = 'Ban dau: ';
strLabelImage = [strLabelImage,num2str(lblTestAll(n)),'.'];
strLabelImage = [strLabelImage,'Du doan: '];
strLabelImage = [strLabelImage,num2str(lblPredictTest),'.'];
if(lblPredictTest == lblImageTest)
    strLabelImage = [strLabelImage,' Ket qua dung.'];
else
    strLabelImage = [strLabelImage,' Ket qua sai.'];
end
title(strLabelImage);
end

```

- Q7 - Hãy viết function đếm số lượng các ảnh có label là n (n là tham số) bị nhận dạng sai theo thuật toán knn. Paste code của function đã chạy được vào bài thực hành và lập bảng kết quả khi chạy với n= 1...40.

```
function CountWrongImages(n)
%Q7- Dem so luong anh co label n bi nhan dang sai
imgTrainImagesAll = importdata('imgTrainImagesAll.mat');
lblTrainLabelsAll = importdata('lblTrainLabelsAll.mat');
Mdl = fitcknn(imgTrainImagesAll', lblTrainLabelsAll);
imgTestImagesAll = importdata('imgTestImagesAll.mat');
lblTestLabelsAll = importdata('lblTestLabelsAll.mat');
nNumTestImgs = size(imgTestImagesAll,2);
count = 0;
for nNumber = 1 : nNumTestImgs
    img = imgTestImagesAll(:,nNumber);
    lblPredictTest = predict(Mdl, img');
    if(lblPredictTest ~= lblTestLabelsAll(nNumber))
        if(lblTestLabelsAll(nNumber) == n)
            count = count + 1;
        end
    end
end
fprintf('\n%d',count);
end
```

1	0	11	1	21	0	31	0
2	0	12	0	22	0	32	1
3	0	13	0	23	0	33	0
4	0	14	0	24	0	34	0
5	0	15	0	25	0	35	0
6	0	16	1	26	0	36	0
7	0	17	0	27	0	37	0
8	0	18	0	28	1	38	0
9	0	19	1	29	0	39	0
10	0	20	1	30	0	40	1

Q7\* - Tương tự bài tập Q7 - nhưng lập bảng confusion matrix.

```
function CountWrongImagesConfussionMatrix()
%Q7*Dem so luong anh co label n bi nhan dang sai va lap Confusion matrix
imgTrainImagesAll = importdata('imgTrainImagesAll.mat');
lblTrainLabelsAll = importdata('lblTrainLabelsAll.mat');
Mdl = fitcknn(imgTrainImagesAll', lblTrainLabelsAll);
imgTestImagesAll = importdata('imgTestImagesAll.mat');
lblTestLabelsAll = importdata('lblTestLabelsAll.mat');
CMatrix = zeros(40,40);
nNumTestImgs = size(imgTestImagesAll,2);
for nNumber = 1 : nNumTestImgs
    imgTest = imgTestImagesAll(:,nNumber);
```

```

        lblImageTest = lblTestLabelsAll(nNumber);
        lblPredictTest = predict(Mdl, imgTest');
        CMatrix(lblImageTest, lblPredictTest)
=CMatrix(lblImageTest, lblPredictTest)+1;
    end
    csvwrite('CMatrix.csv', CMatrix);
end

```

Đường dẫn Confussion Matrix ([CMatrix.csv](#)).

- Q8\*\* (Tương đương 50% đồ án môn học nếu kết hợp với Q7\*) - Hãy viết function tính độ chính xác của thuật toán knn với các tham số khác nhau của hàm knn (ví dụ số lượng nearest neighbors, độ đo distance). Paste code vào bài thực hành và lập bảng kết quả.

```

% Accuracy(1,1): tham so NumNeighbors, k = 1 ->94.167
% Accuracy(1,3): tham so NumNeighbors, k = 3 -> 90.833
% Accuracy(2,1): tham so Distance, do do minkowski -> 94.167
% Accuracy(3,1): tham so Distance, do do euclidean -> 94.167
function d = Accuracy(param,number)
    imgTrainImagesAll = importdata('imgTrainImagesAll.mat');
    lblTrainLabelsAll = importdata('lblTrainLabelsAll.mat');
    imgTestImagesAll = importdata('imgTestImagesAll.mat');
    lblTestLabelsAll = importdata('lblTestLabelsAll.mat');
    switch(param)
        case (1)
            Mdl = fitcknn(imgTrainImagesAll', lblTrainLabelsAll,
'NumNeighbors', number);
        case (2)
            Mdl = fitcknn(imgTrainImagesAll', lblTrainLabelsAll,
'Distance','minkowski');
        case (3)
            Mdl = fitcknn(imgTrainImagesAll', lblTrainLabelsAll,
'Distance','euclidean');
    end
    nNumTestImgs = size(imgTestImagesAll,2);
    count = 0;
    for nNumber = 1 : nNumTestImgs
        fprintf('\n%d ',nNumber);
        img = imgTestImagesAll(:,nNumber);
        lblPredictTest = predict(Mdl, img');
        if(lblPredictTest == lblTestLabelsAll(nNumber))
            count = count + 1;
        end
    end
    d = (count / nNumTestImgs) * 100;
    fprintf('Do chinh xac: %8.3f',d);
end

```

•

	K=1	K=3
--	-----	-----

NumNeighbors	94.167%	90.833%
--------------	---------	---------

	minkowski	euclidean
Distance	94.167%	94.167%