Siddhanth Patel
CS 558 - Computer Vision
Professor Xinchao Wang
9 May 2021

Homework Assignment 3 - Image Classification

Problem: Image Classification.

Source Code:

```
labels = ["coast", "forest", "insidecity"] #labels to use for checking which class

#Returns a histogram that contains number of pixels in the blue/green/red color channels in the bins
#Verifies that the pixels are counted exactly 3 times, otherwise returns None

def hist(pic, num_bins):
    histogram = []
    for i in range(num_bins):
        histogram += [0, 0, 0]
    for i in range(pic.shape[0]):
        for j in range(pic.shape[1]):
            for k in range(3):
                histogram[k*num_bins + int(pic[i][j][k] // (256/num_bins))] += 1

    #Verfication Step: makes sure that all pixels are counted exactly 3 times, once in each color channel
    if int(sum(histogram)/3) == len(pic)*len(pic[0]):
        return histogram
    else:
        return None

#classifies the histograms using "knn" nearest neighbors (findest the prediction)
def threeNNhist(test_hist, train_hist, knn, num_bins):
    print("Results: (" + str(num_bins) + " bins, " + str(knn) +" nearest neighbors) ")
    num_right = 0
    for test in test_hist:
        # Find the hist in the train_hist array that has smallest dist to current img
        mindist = [[-1,0] for i in range(knn)]
```

```
        for train in train_hist:
            a = np.array(test[1])
            b = np.array(train[1])
            dist = np.linalg.norm(a-b) # distance function using numpy between train[1] and test[1]
            for k in range(knn):
                if dist < mindist[k][0] or mindist[k][0] == -1:
                    mindist.insert(k, [dist, train[0]]) #assigns to the test image the label of the training
image that has the nearest representation
                    break

        label = []
        for i in range(knn):
            label += [mindist[i][1]]

        #check which is the best label for the image
        count = [0,0,0]
        for i in range(len(label)):
            for j in range(3):
                if label[i] == labels[j]:
                    count[j] += len(label)-i
        maxval = max(count)
        for i in range(3):
            if count[i] == maxval:
                best = labels[i]
                if best == test[0]:
                    num_right += 1
        #print statement to see classifications
        print("Test image " + test[2] + " of class " + test[0] + " has been assigned to class "+ best +
".")
    #print statement to see accuracy of classifer
    print("Accuracy of classifier: " + str(num_right) + "/12 right.")



if __name__ == "__main__":
    #gets all the images in the ImClass folder that have term "train" or the term "test" and store
them
    train_pics = glob.glob('ImClass/*train*.jpg')
    test_pics = glob.glob('ImClass/*test*.jpg')
```

```
    # For bins = 8 and nearest neighbors = 1 (change these numbers to desired bins # and nearest neighbors #)
    knn_arr = [1, 3]
    bins_arr = [8, 4, 16, 32]

    for knn in knn_arr:
        for num_bins in bins_arr:
            if num_bins == 8 or knn == 3:
                train_hist = []
                test_hist = []

                #for each training pics store histogram in train_hist array
                for file in train_pics:
                    pic = cv2.imread(file)
                    hist1 = hist(pic, num_bins)
                    for i in range(len(labels)):
                        if labels[i] in file:
                            if hist1 != None:
                                train_hist.append([labels[i], hist1])

                #for each testing pics store histogram in test_hist array
                for file in test_pics:
                    pic = cv2.imread(file)
                    hist2 = hist(pic, num_bins)
                    for i in range(len(labels)):
                        if labels[i] in file:
                            if hist1 != None:
                                test_hist.append([labels[i], hist2, file])

                print("------------------------------------------------------------")
                threeNNhist(test_hist, train_hist, knn, num_bins)
                print("------------------------------------------------------------")
```

Output:

```
------------------------------------------------------------
Results: (8 bins, 1 nearest neighbors)
Test image ImClass\coast_test1.jpg of class coast has been assigned to class coast.
Test image ImClass\coast_test2.jpg of class coast has been assigned to class coast.
Test image ImClass\coast_test3.jpg of class coast has been assigned to class coast.
```

Test image ImClass\coast_test4.jpg of class coast has been assigned to class coast.
Test image ImClass\forest_test1.jpg of class forest has been assigned to class forest.
Test image ImClass\forest_test2.jpg of class forest has been assigned to class forest.
Test image ImClass\forest_test3.jpg of class forest has been assigned to class forest.
Test image ImClass\forest_test4.jpg of class forest has been assigned to class forest.
Test image ImClass\insidecity_test1.jpg of class insidecity has been assigned to class forest.
Test image ImClass\insidecity_test2.jpg of class insidecity has been assigned to class forest.
Test image ImClass\insidecity_test3.jpg of class insidecity has been assigned to class insidecity.
Test image ImClass\insidecity_test4.jpg of class insidecity has been assigned to class insidecity.
Accuracy of classifier: 10/12 right.

Results: (8 bins, 3 nearest neighbors)
Test image ImClass\coast_test1.jpg of class coast has been assigned to class coast.
Test image ImClass\coast_test2.jpg of class coast has been assigned to class coast.
Test image ImClass\coast_test3.jpg of class coast has been assigned to class coast.
Test image ImClass\coast_test4.jpg of class coast has been assigned to class coast.
Test image ImClass\forest_test1.jpg of class forest has been assigned to class forest.
Test image ImClass\forest_test2.jpg of class forest has been assigned to class forest.
Test image ImClass\forest_test3.jpg of class forest has been assigned to class forest.
Test image ImClass\forest_test4.jpg of class forest has been assigned to class forest.
Test image ImClass\insidecity_test1.jpg of class insidecity has been assigned to class forest.
Test image ImClass\insidecity_test2.jpg of class insidecity has been assigned to class forest.
Test image ImClass\insidecity_test3.jpg of class insidecity has been assigned to class insidecity.
Test image ImClass\insidecity_test4.jpg of class insidecity has been assigned to class insidecity.
Accuracy of classifier: 10/12 right.

Results: (4 bins, 3 nearest neighbors)
Test image ImClass\coast_test1.jpg of class coast has been assigned to class coast.
Test image ImClass\coast_test2.jpg of class coast has been assigned to class coast.
Test image ImClass\coast_test3.jpg of class coast has been assigned to class coast.
Test image ImClass\coast_test4.jpg of class coast has been assigned to class coast.
Test image ImClass\forest_test1.jpg of class forest has been assigned to class forest.
Test image ImClass\forest_test2.jpg of class forest has been assigned to class insidecity.
Test image ImClass\forest_test3.jpg of class forest has been assigned to class forest.
Test image ImClass\forest_test4.jpg of class forest has been assigned to class insidecity.
Test image ImClass\insidecity_test1.jpg of class insidecity has been assigned to class forest.
Test image ImClass\insidecity_test2.jpg of class insidecity has been assigned to class insidecity.
Test image ImClass\insidecity_test3.jpg of class insidecity has been assigned to class insidecity.

Test image ImClass\insidecity_test4.jpg of class insidecity has been assigned to class insidecity.
Accuracy of classifier: 11/12 right.

Results: (16 bins, 3 nearest neighbors)
Test image ImClass\coast_test1.jpg of class coast has been assigned to class coast.
Test image ImClass\coast_test2.jpg of class coast has been assigned to class coast.
Test image ImClass\coast_test3.jpg of class coast has been assigned to class coast.
Test image ImClass\coast_test4.jpg of class coast has been assigned to class coast.
Test image ImClass\forest_test1.jpg of class forest has been assigned to class forest.
Test image ImClass\forest_test2.jpg of class forest has been assigned to class forest.
Test image ImClass\forest_test3.jpg of class forest has been assigned to class forest.
Test image ImClass\forest_test4.jpg of class forest has been assigned to class forest.
Test image ImClass\insidecity_test1.jpg of class insidecity has been assigned to class forest.
Test image ImClass\insidecity_test2.jpg of class insidecity has been assigned to class forest.
Test image ImClass\insidecity_test3.jpg of class insidecity has been assigned to class forest.
Test image ImClass\insidecity_test4.jpg of class insidecity has been assigned to class insidecity.
Accuracy of classifier: 9/12 right.

Results: (32 bins, 3 nearest neighbors)
Test image ImClass\coast_test1.jpg of class coast has been assigned to class coast.
Test image ImClass\coast_test2.jpg of class coast has been assigned to class coast.
Test image ImClass\coast_test3.jpg of class coast has been assigned to class coast.
Test image ImClass\coast_test4.jpg of class coast has been assigned to class coast.
Test image ImClass\forest_test1.jpg of class forest has been assigned to class forest.
Test image ImClass\forest_test2.jpg of class forest has been assigned to class forest.
Test image ImClass\forest_test3.jpg of class forest has been assigned to class forest.
Test image ImClass\forest_test4.jpg of class forest has been assigned to class forest.
Test image ImClass\insidecity_test1.jpg of class insidecity has been assigned to class forest.
Test image ImClass\insidecity_test2.jpg of class insidecity has been assigned to class forest.
Test image ImClass\insidecity_test3.jpg of class insidecity has been assigned to class forest.
Test image ImClass\insidecity_test4.jpg of class insidecity has been assigned to class insidecity.
Accuracy of classifier: 9/12 right.
