

```

1 %=====
2 %LOADS THE VIDEO INTO MEMORY
3 %=====
4 pkg load image
5 pkg load communications
6
7 file = "/Users/charlesdavi/Desktop/Image_Sequence/Image_Sequence/Training_Set/"
8
9 total_num_directories = 24; %the number of videos
10
11 num_images = 10; %the number of frames per video
12
13 for i = 1 : total_num_directories
14
15     directory = [file int2str(i) '/'];
16
17     for j = 1 : num_images
18
19         image_array_training{i,j} = imread([directory num2str(j) '.jpg']);
20
21     endfor
22
23 endfor
24
25 file = "/Users/charlesdavi/Desktop/Image_Sequence/Image_Sequence/Testing_Set/"
26
27 total_num_directories = 20; %the number of videos
28
29 num_images = 10; %the number of frames per video
30
31 for i = 1 : total_num_directories
32
33     directory = [file int2str(i) '/'];
34
35     for j = 1 : num_images
36
37         image_array_testing{i,j} = imread([directory num2str(j) '.jpg']);
38
39     endfor
40
41 endfor
42
43 %=====
44 %TRAINING LOOP
45 %=====
46 gap_size = 3000; %this determines the resolution of the frames - a larger gap leads to
faster prediction
47
48 I = image_array_training{1,1};
49 x = double((I(:, :, 1) + I(:, :, 2) + I(:, :, 3))/3) > 80); %converts image to a binary
matrix_type
50 x = x(:)'; %reshapes the image as a vector
51 M = size(x,2);
52
53 x = x(1:gap_size:M);
54
55 N = size(x,2);
56
57 x(N+1) = 0;
58
59 counter = 1;
60
61 %these are randomized initialized values necessary to kick off the loop, but they don't
provide any information about the actual data
62 video_dataset = [];
63 video_dataset(1,:) = rand(1, num_images + 1);
64 video_dataset(1,num_images + 1) = 0;
65
66 video_dataset(2,:) = rand(1, num_images + 1);
67 video_dataset(2,num_images + 1) = 1;

```

```

68 %
69 %these are randomized initialized values necessary to kick off the loop, but they don't
provide any information about the actual data
70 img_dataset = [];
71 img_dataset(1,:) = rand(1,N+1);
72 img_dataset(1,N+1) = 0;
73 img_dataset(2,:) = rand(1,N+1);
74 img_dataset(2,N+1) = 1;
75
76 num_directories = 24;
77
78 video_classifiers = [];
79
80 for i = 1 : num_directories
81
82     directory = [file int2str(i) '/'];
83     video_vector = [];
84
85     tic;
86     for j = 1 : num_images
87
88         I = image_array_training{i,j};
89
90         x = double(((I(:,:,1) + I(:,:,2) + I(:,:,3))/3) > 80); %converts image to a binary matrix
91         x = x(:)'; %reshapes the matrix into a row vector
92
93         x = x(1:gap_size:M);
94
95         x(N+1) = (i>12); %stores the classifier for the video (directories 1 to 12 are 0)
96
97         [IGNORE video_vector(j)] = find_NN(x, img_dataset,N); %predicts the class of the new frame
98         img_dataset = [img_dataset; x]; %adds the new frame to the dataset
99
100     endfor
101
102     video_vector(num_images + 1) = x(N+1); %stores the classifier for the video
103
104     [IGNORE video_classifiers(counter)] = find_NN(video_vector, video_dataset,num_images);
105     video_dataset = [video_dataset; video_vector]; %stores the prediction vector in the dataset
106     counter = counter + 1;
107     toc
108
109 endfor
110
111 %=====
112 %TESTING LOOP
113 %=====
114
115 num_directories = 20
116
117 for i = 1 : num_directories
118
119     directory = [file int2str(i) '/'];
120     video_vector = [];
121
122     tic;
123     for j = 1 : num_images
124
125         I = image_array_testing{i,j};
126
127         x = double(((I(:,:,1) + I(:,:,2) + I(:,:,3))/3) > 80); %converts image to a binary matrix
128         x = x(:)'; %reshapes the matrix into a row vector
129
130         x = x(1:gap_size:M);
131
132         x(N+1) = (i>10); %stores the classifier for the video (directories 1 to 10 are 0)
133
134         [IGNORE video_vector(j)] = find_NN(x, img_dataset,N); %predicts the class of the new frame
135         img_dataset = [img_dataset; x]; %adds the new frame to the dataset
136

```

```
137 endfor
138
139 video_vector(num_images + 1) = x(N+1); %stores the classifier for the video
140
141 [IGNORE video_classifiers(counter)] = find_NN(video_vector, video_dataset,num_images);
142 video_dataset = [video_dataset; video_vector]; %stores the prediction vector in the dataset
143 counter = counter + 1;
144 toc
145
146 endfor
147
148 %=====
149 %ACCURACY
150 %=====
151 video_dataset(1:2,:) = []; %this deletes the initial entry which was a placeholder to
start the loop
152 accuracy_vector = (video_classifiers' == video_dataset(:,num_images + 1));
153 accuracy = sum(accuracy_vector)/44
154
```