PCA (Calls: 1, Time: 192.957 s)

Generated 13-Jun-2023 17:00:54 using performance time.
Script in file C:\Users\Pietro\Documents\DELFT\DS\courseworks\C\PCA.m
Copy to new window for comparing multiple runs

Parents (calling functions)

No parent

Lines that take the most time

Line Number	Code	Calls	Total Time (s)	% Time	Time Plot
44	`C = (1/m) * (M*M');`	1	97.570	50.6%	
16	`image =imread(fullfile(fl(i).folder …`	5854	45.273	23.5%	
51	`[Vec, D_val] = eigs(C, num_of_eigenv…`	1	19.983	10.4%	
32	`M = D - D_means;`	1	8.831	4.6%	
61	`plot(cumsum(explained_variance_tot))`	1	7.500	3.9%	
All other lines			13.799	7.2%	
Totals			192.957	100%	

Children (called functions)

Function Name	Function Type	Calls	Total Time (s)	% Time	Time Plot
imread	Function	5855	43.384	22.5%	
eigs	Function	1	18.378	9.5%	
newplotwrapper	Function	1	3.556	1.8%	
imshow	Function	17	2.261	1.2%	
fullfile	Function	5855	1.668	0.9%	
bar	Function	1	1.449	0.8%	
subplot	Function	15	1.075	0.6%	
clf	Function	1	0.773	0.4%	
rescale	Function	25	0.290	0.2%	
ylim	Function	1	0.203	0.1%	
get_file_list	Function	1	0.194	0.1%	
metaDim>metaDim.metaDim	Class method	1	0.160	0.1%	
hold	Function	1	0.122	0.1%	
mean	Function	1	0.107	0.1%	
varNamesDim>varNamesDim.varNamesDim	Class method	1	0.054	0.0%	

table.table>table.table	Class method	1	0.019	0.0%	
rowNamesDim>rowNamesDim.rowNamesDim	Class method	1	0.012	0.0%	
trace	Function	2	0.007	0.0%	
Self time (built-ins, overhead, etc.)			119.247	61.8%	
Totals			192.957	100%	

Code Analyzer results

No Code Analyzer messages.

Coverage results

Show coverage for parent folder

Total lines in function	106
Non-code lines (comments, blank lines)	50
Code lines (lines that can run)	56
Code lines that did run	56
Code lines that did not run	0
Coverage (did run/can run)	100.00 %

Function listing

Time	Calls	Line	
< 0.001	1	2	base folder = 'data';
< 0.001	1	3	inputFolder = 'chest-rays';
< 0.001	1	4	outputFolder = inputFolder+" processed 150";
		5	
0.195	1	6	fl = get file list(base folder, outputFolder);
		7	
		8	%%
		9	%get size of images
0.069	1	10	example image = imread(fullfile(fl(1).folder , fl(1).name));
< 0.001	1	11	[h, w, d] = size(example image);
		12	
		13	%construct data matrix
< 0.001	1	14	D = zeros(w*h*d, length(fl));
< 0.001	1	15	for i = 1:numel(fl)
45.273	5854	16	image =imread(fullfile(fl(i).folder , fl(i).name));
0.068	5854	17	[h, w, d] = size(image);
0.640	5854	18	x = double(reshape(image, w*h*d, 1))/255;
1.768	5854	19	D(:, i) = x;
0.004	5854	20	end
		21	%%

```
                22
                23  %constuct mean vector
0.138       1   24  D means = mean(D, 2);
                25
                26  %get average image
0.004       1   27  average image =  uint8(reshape(D means, h, w, d)*255);
                28  %imshow(average image)
                29
                30  %%
                31
8.831       1   32  M = D - D means;
0.245       1   33  m = length(M);
0.016       1   34  d length = length(D);
                35
0.206       1   36  image reference =  uint8(reshape(D(:,1), h, w, d)*255);
0.028       1   37  image reference diff =  uint8(reshape(M(:,1), h, w, d)*255);
                38  %imshow(image reference)
                39  %imshow(image reference diff)
                40
                41  %%
                42
                43  %compute covariance matrix
97.570      1   44  C =  (1/m) * (M*M');
                45
                46  %C = (1/m) * (M'*M);
                47
                48  %%
                49  %compute eigenvalues and eigenvectors
0.025       1   50  num of eigenvec = 10;
19.983      1   51  [Vec, D val] = eigs(C, num of eigenvec);
                52
                53  %%
                54
                55  %plot explained variance
0.086       1   56  explained variance tot = diag(D val)/trace(C);
0.003       1   57  explained variance tot sum = sum(explained variance tot);
0.005       1   58  explained variance = diag(D val)/trace(D val);
                59
3.489       1   60  figure;
7.500       1   61  plot(cumsum(explained variance tot))
0.245       1   62  ylim([0 1])
0.161       1   63  hold on
1.609       1   64  bar(explained variance tot)
                65
                66  %%
```

```
                67  %plot eigenVectors as images
 0.004          1  68  num1 = ceil(num of eigenvec/2);
< 0.001         1  69  num2 = 2;
 0.860          1  70  clf
< 0.001         1  71  for i=1:num of eigenvec
 1.129         10  72      subplot(num1, num2, i);
 0.005         10  73      em = Vec(:, i);
 0.347         10  74      image = uint8(reshape(rescale(em, 0, 255), h, w, d));
 1.974         10  75      imshow(image);
 0.001         10  76  end
                77  %%
                78  %get eigenvectors weights for image
< 0.001         1  79  image index = 3800;
                80
 0.004          1  81  weights image = Vec' * (M(:, image index)- D means);
                82
                83  %show original image
 0.136          1  84  imshow(reshape(D(:, image index), h, w, d));
                85
                86  %%
                87
                88  %plot projected eigenvectors of image
< 0.001         1  89  for i=1:5
 0.075          5  90      subplot(num1, num2, i);
 0.003          5  91      em = Vec(:, i);
 0.003          5  92      p1x= M(:, image index)'*em*em;
 0.007          5  93      image = uint8(reshape(rescale(p1x, 0, 1), h, w, d)*255);
 0.203          5  94      imshow(image);
< 0.001         5  95  end
                96
                97  %%
                98  %new image = zeros(size(D means));
                99
               100  %reconstruct image from eigenvectors combination
< 0.001         1 101  new image = D means;
               102  %weights image = [0,0,0,0,0]; if want to do it manually
< 0.001         1 103  for i = 1:length(Vec(1,:))
 0.003         10 104      new image = new image + weights image(i)*Vec(:,i);
 0.005         10 105      new image = rescale(new image,0,1);
 0.001         10 106  end
 0.030          1 107  imshow(uint8(reshape(new image, h, w, d)*255));
```