

```

1   import numpy as np
2   import pandas as pd
3   import cv2
4   from matplotlib import pyplot as plt
5   import matplotlib.cm as cm
6   import matplotlib.gridspec as gridspec
7   from pylab import imread
8   from skimage.color import rgb2gray
9   import pywt

```

```

11  def ShowImage(ImageList, nRows = 1, nCols = 2, WidthSpace = 0.00, HeightSpace = 0.00):
12      from matplotlib import pyplot as plt
13      import matplotlib.gridspec as gridspec
14
15      gs = gridspec.GridSpec(nRows, nCols)
16      gs.update(wspace=WidthSpace, hspace=HeightSpace) # set the spacing between axes.
17      plt.figure(figsize=(20,10))
18      for i in range(len(ImageList)):
19          ax1 = plt.subplot(gs[i])
20          ax1.set_xticklabels([])
21          ax1.set_yticklabels([])
22          ax1.set_aspect('equal')
23
24          plt.subplot(nRows, nCols,i+1)
25
26          image = ImageList[i].copy()
27          if (len(image.shape) < 3):
28              plt.imshow(image, plt.cm.gray)
29          else:
30              plt.imshow(image)
31              plt.title("Image " + str(i))
32              plt.axis('off')
33
34      plt.show()

```

```

74  # Read Image
75  image_color = imread("Sample05/balloon.jpg")
76  # Convert Image into Gray
77  image_gray = cv2.cvtColor(image_color, cv2.COLOR_RGB2GRAY)
78  # Display Image
79  ShowImage([image_color, image_gray], 1, 2)
80
81  import pywt
82  import pywt.data
83  from pywt import dwt2, idwt2
84  from pywt._doc_utils import wavedec2_keys, draw_2d_wp_basis
85
86  discrete_wavelets = ['db5', 'sym5', 'coif5', 'bior1.3', 'haar']
87
88  x = image_gray.astype(np.float32)
89  shape = x.shape
90
91  max_lev = 2      # how many levels of decomposition to draw
92  label_levels = 2 # how many levels to explicitly label on the plots

```

```

94 for i in discrete_wavelets:
95     fig, axes = plt.subplots(2, max_lev + 1, figsize=[14, 8])
96     for level in range(0, max_lev + 1):
97         if level == 0:
98             # show the original image before decomposition
99             axes[0, 0].set_axis_off()
100            axes[1, 0].imshow(x, cmap=plt.cm.gray)
101            axes[1, 0].set_title('Image')
102            axes[1, 0].set_axis_off()
103            continue
104
105            # plot subband boundaries of a standard DWT basis
106            draw_2d_wp_basis(shape, wavedec2_keys(level), ax=axes[0, level],
107                            label_levels=label_levels)
108            axes[0, level].set_title('{} level\ndecomposition'.format(level))
109
110            # compute the 2D DWT
111            c = pywt.wavedec2(x, i, mode='periodization', level=level)
112            # normalize each coefficient array independently for better visibility
113            c[0] /= np.abs(c[0]).max()
114            for detail_level in range(level):
115                c[detail_level + 1] = [d/np.abs(d).max() for d in c[detail_level + 1]]
116            # show the normalized coefficients
117            arr, slices = pywt.coeffs_to_array(c)
118            axes[1, level].imshow(arr, cmap=plt.cm.gray)
119            axes[1, level].set_title('Coefficients\n({} level)'.format(level))
120            axes[1, level].set_axis_off()

```

Image 0

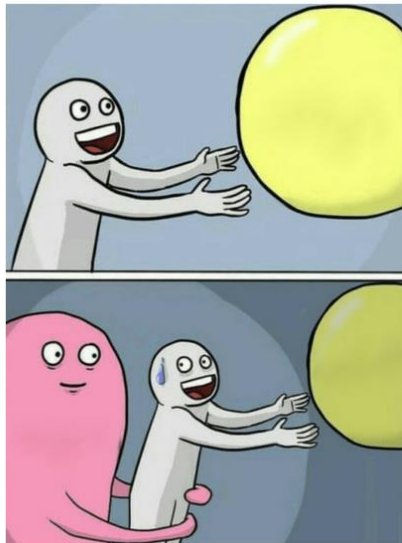


Image 1



1 level
decomposition

a	h
v	d

2 level
decomposition

aa	ah	h
av	ad	
v		d

Image



Coefficients
(1 level)



1 level
decomposition

Coefficients
(2 level)



2 level
decomposition

Image



Coefficients
(1 level)



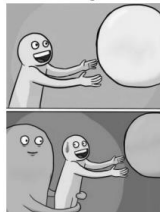
1 level
decomposition

Coefficients
(2 level)



2 level
decomposition

Image



Coefficients
(1 level)



Coefficients
(2 level)



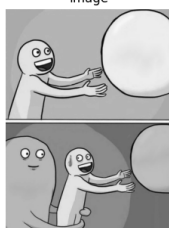
1 level
decomposition

a	h
v	d

2 level
decomposition

aa	ah	h
av	ad	
v		d

Image



Coefficients
(1 level)



1 level
decomposition

Coefficients
(2 level)



2 level
decomposition

a	h
v	d

aa	ah	h
av	ad	
v		d

Image



Coefficients
(1 level)



Coefficients
(2 level)



```

125 def my_func(img, wavelet, de_level):
126     image_color = imread("Sample05/" + img)
127     image_gray = cv2.cvtColor(image_color, cv2.COLOR_RGB2GRAY)
128     ShowImage([image_color, image_gray], 1, 2)
129
130     titles = ['Approximation', 'Horizontal detail',
131              'Vertical detail', 'Diagonal detail']
132     coeffs2 = dwt2(image_gray, wavelet)
133     LL, (LH, HL, HH) = coeffs2
134
135     cmaplist = [cm.gray, cm.jet, cm.rainbow, cm.viridis, cm.cubehelix, cm.RdBu]
136     fig = plt.figure(figsize=(20,10))
137     for i, a in enumerate([LL, LH, HL, HH]):
138         ax = fig.add_subplot(2, 2, i + 1)
139         ax.imshow(a, interpolation="nearest", cmap=cmaplist[5])
140         ax.set_title(titles[i], fontsize=10)
141         ax.set_xticks([])
142         ax.set_yticks([])
143
144     fig.tight_layout()
145     plt.show()
146
147     x = image_gray.astype(np.float32)
148     shape = x.shape
149
150     max_lev = de_level      # how many levels of decomposition to draw
151     label_levels = de_level # how many levels to explicitly label on the plots
152
153     fig, axes = plt.subplots(2, max_lev + 1, figsize=[14, 8])
154     for level in range(0, max_lev + 1):
155         if level == 0:
156             # show the original image before decomposition
157             axes[0, 0].set_axis_off()
158             axes[1, 0].imshow(x, cmap=plt.cm.gray)
159             axes[1, 0].set_title('Image')
160             axes[1, 0].set_axis_off()
161             continue
162
163             # plot subband boundaries of a standard DWT basis
164             draw_2d_wp_basis(shape, wavedec2_keys(level), ax=axes[0, level],
165                             label_levels=label_levels)
166             axes[0, level].set_title('{} level\ndecomposition'.format(level))
167
168             # compute the 2D DWT
169             c = pywt.wavedec2(x, wavelet, mode='periodization', level=level)
170             # normalize each coefficient array independently for better visibility
171             c[0] /= np.abs(c[0]).max()
172             for detail_level in range(level):
173                 c[detail_level + 1] = [d/np.abs(d).max() for d in c[detail_level + 1]]
174             # show the normalized coefficients
175             arr, slices = pywt.coeffs_to_array(c)
176             axes[1, level].imshow(arr, cmap=plt.cm.gray)
177             axes[1, level].set_title('Coefficients\n({} level)'.format(level))
178             axes[1, level].set_axis_off()
179
180     plt.tight_layout()
181     plt.show()

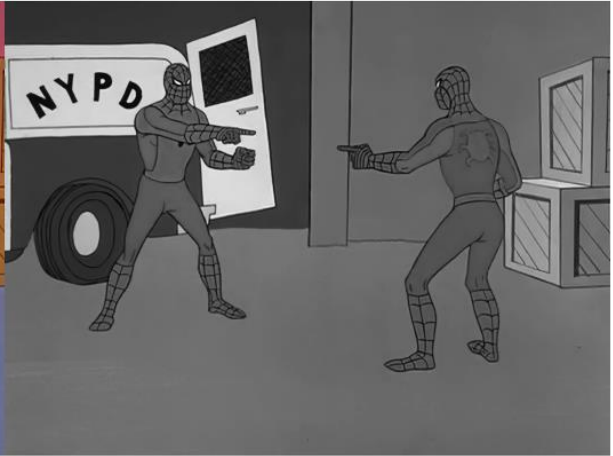
```

```
183 my_func("spider.jpg", "bior1.3", 3)
```

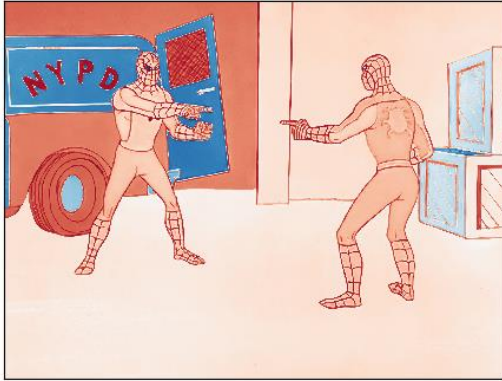
Image 0



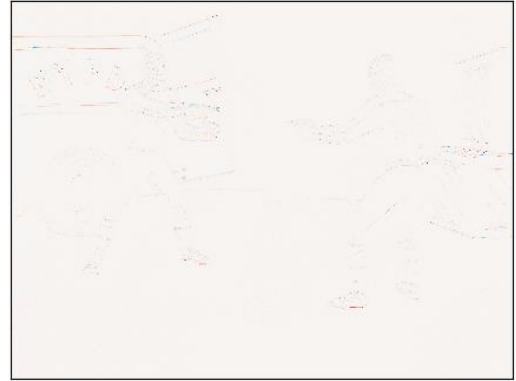
Image 1



Approximation



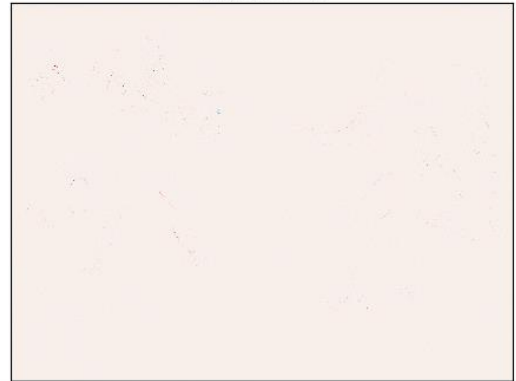
Horizontal detail



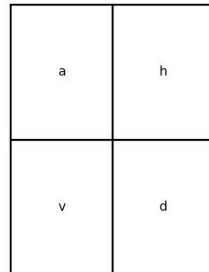
Vertical detail



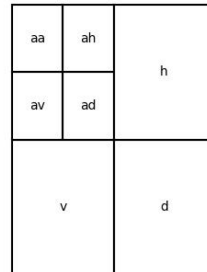
Diagonal detail



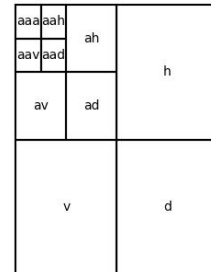
1 level
decomposition



2 level
decomposition



3 level
decomposition



Coefficients
(1 level)

Coefficients
(2 level)

Coefficients
(3 level)

Image

