


電腦視覺 作業五

指導老師 傅楸善

學生 蔡宇晴

學號 R08945050

	原圖	Dilation
圖 片		
程 式 碼	<pre> 10 def dilation_(img): 11 12 kernel = np.array([[0,1,1,1,0],[1,1,1,1,1],[1,1,1,1,1],[1,1,1,1,1],[0,1,1,1,0]]) 13 h,w = img.shape 14 img = np.pad(img,(2,2)) 15 temp_ = np.zeros((h+4,w+4)) 16 # 答案放置temp_,maximum filter kernel 設置完成 17 for height in range(2,h+2): 18 for width in range(2,w+2): 19 slice_range = img[height-2:height+3,width-2:width+3] 20 max_value = np.max(slice_range * kernel) 21 temp_[height,width] = max_value 22 temp_ = np.uint8(temp_) 23 temp_ = temp_[2:h+2,2:w+2] </pre> <ol style="list-style-type: none"> 1. 建立一個 35553 的八角形，在影像中平移時，選擇在八角形範圍內的極大值作為更新點 (i,j) 於 temp_。 2. 輸出 temp_ 	
描 述	<p>影像整體亮度明顯提高、白色區域明顯變大，從髮尾部分可以看出。圖片中的白點顆粒變的明顯。</p> <p>影像整體變亮、變得平滑。</p>	

	原圖	Erosion
圖 片		
程 式 碼	<pre> 30 def erosion(img): 31 32 kernel = np.array([[256,1,1,1,256],[1,1,1,1,1],[1,1,1,1,1],[1,1,1,1,1],[256,1,1,1,256]]) 33 h,w = img.shape 34 img = np.pad(img,(2,2)) 35 temp_ = np.zeros((h+4,w+4)) 36 for height in range(2,h+2): 37 for width in range(2,w+2): 38 slice_range = img[height-2:height+3,width-2:width+3] 39 min_value = np.min(slice_range * kernel) 40 temp_[height,width] = min_value 41 temp_ = np.uint8(temp_) 42 temp_ = temp_[2:h+2,2:w+2] </pre> <p>1.建立一個 35553 的八角形，在影像中平移時，選擇在八角形範圍內的極小值作為更新點 (i,j) 於 temp_。</p> <p>2.輸出 temp_</p> <p>備註：為了不讓 kernel 的數值影響最後更新數值，因此把 kernel 四端點設為 256。</p>	
描 述	<p>影像整體亮度明顯降低、黑色區域明顯變大，從髮尾部分可以看出，且眼睛部分的眼白明顯變少了。一些小白點也被去除。</p> <p>影像整體變暗、變得平滑。</p>	

	原圖	Opening
圖 片		
程 式 碼	<pre> 49 def opening(): 50 try: 51 img = cv2.imread("./lena.bmp",cv2.IMREAD_GRAYSCALE) 52 except: 53 print("No image file") 54 e = erosion(img) 55 d = dilation_(e) 56 cv2.imwrite("./opening_lena.jpg",d) </pre> <p>先做 erosion，在做 dilation。</p>	
描 述	<p>Opening 和 Erosion 都有去除白點的功能，但不同的是，Opening 只對小於 kernel 範圍的白色區域進行去除，因此相較於 Erosion，Opening 能保留較多原圖訊息。</p> <p>在 Opening 中，髮尾亮處消失很多，但眼白部分則和原圖差不多;而 Erosion 則是眼白及髮尾都被侵蝕。</p> <p>影像整體變暗（但 Erosion 較明顯）變得平滑。</p>	

	原圖	Closing
圖 片		
程 式 碼	<pre> 64 def closing(): 65 try: 66 img = cv2.imread("./lena.bmp", cv2.IMREAD_GRAYSCALE) 67 except: 68 print("No image file") 69 d = dilation_(img) 70 e = erosion(d) </pre> <p>先做 dilation，在做 erosion。</p>	
描 述	<p>Closing 和 Dilation 都使影響整體變亮許多，與 Dilation 不同的是，Closing 保留較多原圖資訊，例如眼白處，Closing 與原圖差不多，但 Dilation 會使眼白區擴大，影像整體變亮（但 Dilation 較明顯）、變得平滑。</p>	