Machine Vision Homework#1

110590017 陳姿安

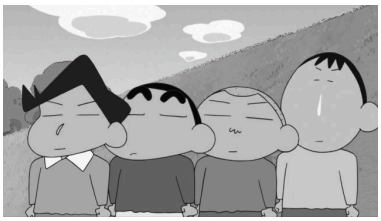
1.Image Quantization(binary, gray, index-color)

1-1. Convert the color image to the grayscale image

- Formula: (0.3 x R) + (0.59 x G) + (0.11 x B).

1. let R, G, B be the same value of each channel by formula.

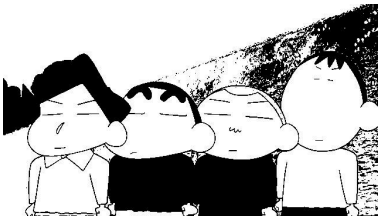




1-2. Convert the grayscale image to the binary image

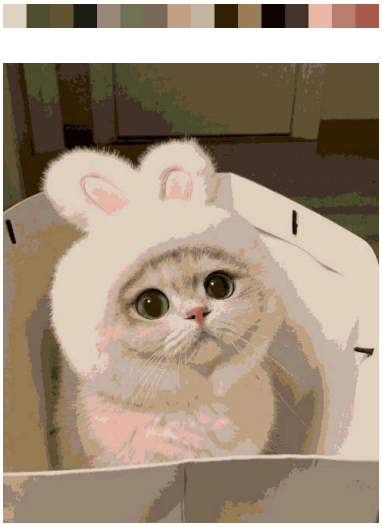
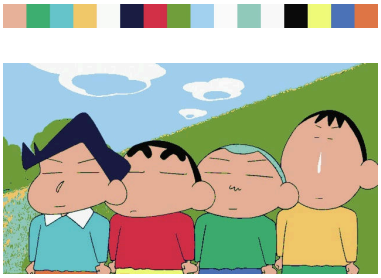
- Choose a appropriate threshold by yourself.
- if brightness > 127 then 255 else 0





1-3 Convert the color image to the index-color image

- Define your own colormap of 16 type colors.
- def index_color(img_path) return index_color
  1. 先擷取所有出現過得顏色對出現次數進行排序
  2. 使用排序後的 list 針對 rgb 進行分類
     1. r,g,b 三群進行分類
     2. 透過查看該顏色 rgb 的最大值進行分類
     3. 使用 deviation 變數將較為相近的顏色去除
     4. 若是總數低於 16 色則降低 deviation 再進行一次分類
  3. 分類後依序在各群取出顏色
  4. 取出後回傳最終結果
- def deaw_by_color(img_path, color, filename)
  1. 使用 find_colose(color, index_color)選擇原始圖片中最接近 index_color 的顏色
     1. 計算當前顏色與 index_color 內所有顏色的距離
     2. 回傳距離最近的顏色
  2. 將原始圖片中的顏色替換成 find_colose 回傳的結果

(b, g, r)
img1 = [(86, 175, 219), (0, 76, 0), (97, 48, 2), (150, 220, 249), (16, 78, 62), (161, 118, 66), (0, 5, 44), (0, 140, 32), (142, 84, 24), (198, 228, 239), (0, 12, 4), (57, 27, 26), (20, 44, 80), (114, 133, 131), (85, 71, 41), (1, 41, 228)],

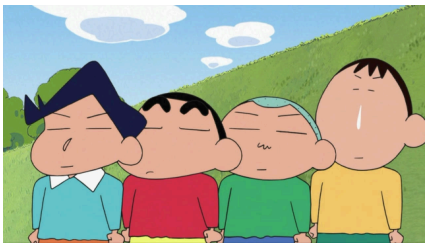
img2 = [(195, 210, 226), (48, 79, 78), (53, 83, 100), (22, 30, 29), (120, 134, 153), (88, 114, 113), (90, 104, 123), (131, 162, 193), (166, 180, 199), (5, 32, 52), (85, 125, 154), (0, 3, 14), (43, 51, 68), (171, 184, 234), (113, 125, 188), (77, 90, 169)],

img3 = [(152, 176, 232), (111, 173, 63), (202, 195, 101), (106, 198, 242), (250, 250, 248), (69, 29, 28), (73, 48, 210), (62, 158, 115), (239, 208, 163), (250, 250, 250), (189, 201, 146), (249, 248, 248), (10, 10, 10), (122, 250, 240), (184, 114, 78), (70, 118, 222)]

2.Resizing Image

2-1. Resizing image to 1 2 and 2 times without interpolation

1. 根據 ppt 給的圖示進行縮放



Resizing image to 1 2 and 2 times with interpolation

- You can use bilinear or bicubic interpolation
  1. 透過新舊圖片比例計算 x_origin, y_origin 獲得新圖片在原始圖片中的位置
  2. 使用 x0,y0,x1,y1 定位原始圖片出周圍四點
  3. 化簡 ppt 的公式進行計算
  4. 將算出的顏色給當前新圖片