**Computer vision – HW4**

# 程式碼

|  |
| --- |
| **import** cv2  **import** numpy **as** np  **def** nothing**(**x**):** **pass**  cap **=** cv2**.**VideoCapture**(**0**)**  ### Create window and trackbar. ###  cv2**.**namedWindow**(**"Tracking"**)**  cv2**.**createTrackbar**(**"LOW" **,** "Tracking"**,** 0**,** 255**,** nothing**)**  cv2**.**createTrackbar**(**"HIGH"**,** "Tracking"**,** 255**,** 255**,** nothing**)**  **while** **True:**  frame **=** cv2**.**imread**(**'lena.jpg'**)**  gray **=** cv2**.**cvtColor**(**frame**,** cv2**.**COLOR\_BGR2GRAY**)** # BGR to GRAY.  ### Returns the current position of the specified trackbar. ###  LOW **=** cv2**.**getTrackbarPos**(**"LOW" **,** "Tracking"**)**  HIGH **=** cv2**.**getTrackbarPos**(**"HIGH"**,** "Tracking"**)**  ### Convert to numpy array. ###  l\_b **=** np**.**array**([**LOW**,**LOW**,**LOW**])**  u\_b **=** np**.**array**([**HIGH**,**HIGH**,**HIGH**])**  ### Calculate 2 images. ###  mask **=** cv2**.**inRange**(**frame**,** l\_b**,** u\_b**)** # Grab a specific range of colors.  res **=** cv2**.**bitwise\_and**(**gray**,** gray**,** mask**=**mask**)** # Perform "intersection" operation on two images.  ### Show image ###  cv2**.**imshow**(**"frame"**,** frame**)**  cv2**.**imshow**(**"mask"**,** mask**)**  cv2**.**imshow**(**"res"**,** res**)**  key **=** cv2**.**waitKey**(**1**)**  **if** key **==** 27**:**  **break**  cv2**.**destroyAllWindows**()** |

# 成果圖







