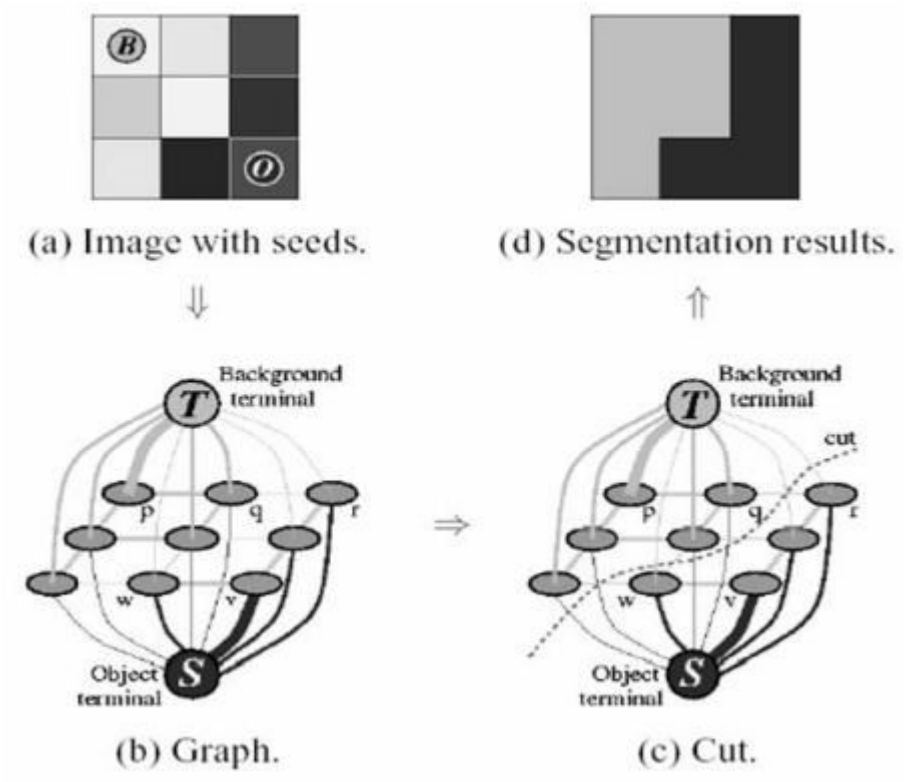
Lab03

GrabCut Algorithm (40%)

Background subtraction
Threshold
Connected component (60%)

https://docs.opencv.org/3.4/d8/d83/tutorial_py_grabcut.html

GrabCut Algorithm



GrabCut Algorithm

1. 讀取圖片

GrabCut Algorithm

2. 使用滑鼠指定矩形區域

cv2.selectROI()

GrabCut Algorithm

```
b_Model = np.zeros((1,65),np.float64)
f_Model = np.zeros((1,65),np.float64)
```

演算法參數初始化

rect: (左上x, 左上y, 寬, 高)

```
mask_new, b_model, f_model=cv2.grabCut(img, None, rect, b_Model, f_Model, iter_num, cv2.GC_INIT_WITH_RECT)
```

```
mask = np.where((mask_new==0)|(mask_new==2),0,1).astype('uint8')
img = img*mask[:,:,np.newaxis]
```

用mask的方式留下前景物體

GrabCut Algorithm

Demo: test.jpg

example: GrabCut for 15 iterations

Background subtraction
Threshold
Connected component
(60%)

輸入一段影片

輸出成果

Background subtraction

↓

Threshold

↓

Connected component

↓

框出前景

0. 讀取影片

影片(video)是由連續的影像(image)組成，組成影片的影像稱為影格(frame)，影片播放時會不斷呈現新的影格，影格間的時間稱作更新頻率(frame rate)。

- cap = cv2.VideoCapture(filename)
- cap.isOpened()
  - 檢查影片是否被成功讀取
- ret, frame = cap.read()
  - 不斷讀取來源影格，把資訊寫進frame
- cv2.imshow("frame", videoFrame)
  - 顯示影片
- cv2.waitKey(33)
  - 等待幾毫秒再讀取下一禎

1. Background Subtraction

- backSub = cv2.createBackgroundSubtractorMOG2()
  - 創造一個BackgroundSubtractor
- fgmask = backSub.apply(frame)
  - 用在frame上面

2. Threshold

- shadowval = backSub.getShadowValue()
  - 找出shadow的值，default = 127，0: background，255: foreground
- ret, nmask = cv2.threshold(fgmask, shadowval, 255, cv2.THRESH_BINARY)
  - 輸出圖片　　輸入圖片　threshold最小值　最大值　　method

3. Connected Component

- Two-Pass Algorithm:

Pass 1:

• Perform label assignment and label propagation.

• Construct the equivalence relations between labels when two different labels propagate to the same pixel.

• Apply resolve function to find the transitive closure of all equivalence relations.

Pass 2:

• Perform label translation

3. Connected Component

- 找connected component時，要同時計算相連區域面積
- 若相連區域面積 > T，找出相連區域最外圍的四個邊

4. 框出物體

- opencv畫矩形
  - cv2.rectangle(影像, (頂點座標), (對向頂點座標), 顏色, 線條寬度)