

Object and its effects eliminate

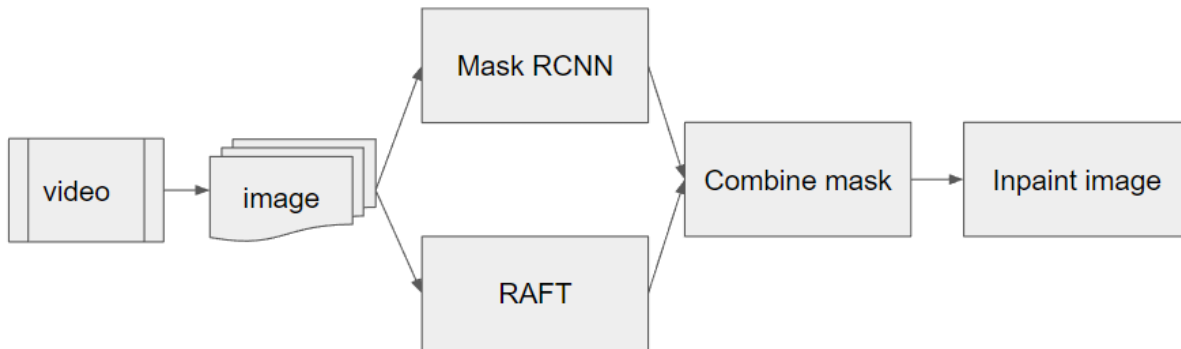
Member

- 數據所 310554031 葉詠富
- 數據所 310554037 黃乾哲

Code

- https://github.com/frankye1000/CV_Final_Project

Workflow



Mask R-CNN

- use the Mask RCNN to select the object.
- The training data is "PennFudanPed" which is containing images that are used for pedestrian detection in the experiments reported.



Since the video background is clear so the final result can select the people easily. We use the result to do the threshold, if pixel is not including in the object segment, we set black. Otherwise, we set white



RAFT

- Flow to RGB to Grayscale
 - Use the pretrain model raft-thing to detect the flow from people and shadow
 - convert flow to rgb images

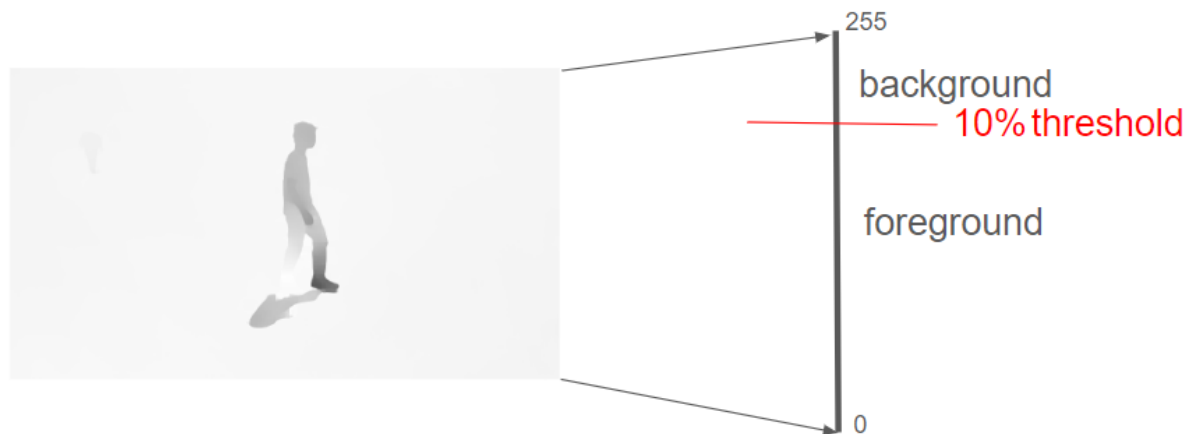


- convert rgb image to grayscale image



RAFT Mask by Threshold

- Mapping grayscale images onto a 1D vector
- Set the threshold to create RAFT Mask.
- The threshold is set with percentiles.
- For example: 10% of the data is less than the threshold as the foreground, greater than the threshold as the background



- threshold 2% vs 10%
 - 2% covers less shadows, but reduces the effect of optical flow in nearby scenes
 - 10% covers more shadows, but increases the effect of the optical flow of nearby scenes
 - We choose 10%, hoping to cover more shadows



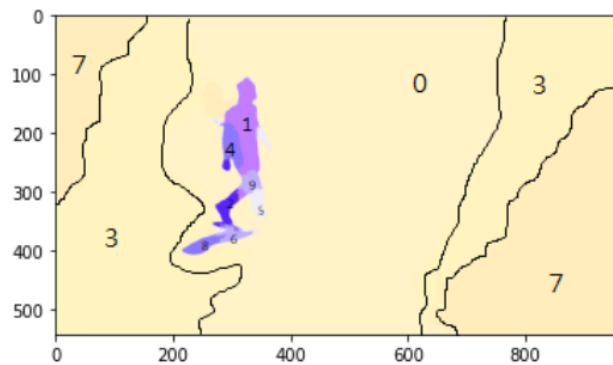
2%



10%

RAFT Mask by KMeans

- Divide the rgb optical flow map into 10 groups using Kmeans
- Sort by the number of pixels in the group
- If the difference between the front and the back is 5 times, it is the dividing point between the foreground and background.
- Example: $90505/4247=21 > 5$

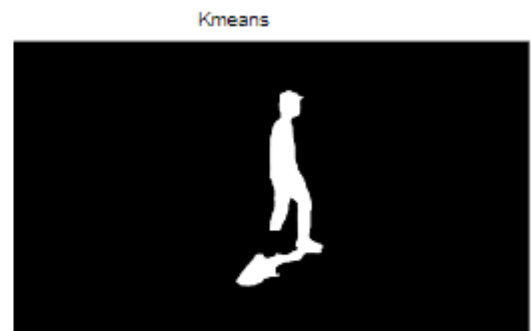


group id	n	
5	605	foreground
6	714	foreground
8	754	foreground
2	879	foreground
9	1288	foreground
4	2822	foreground
1	4247	foreground
0	90505	background
3	199233	background
7	221093	background

dividing point

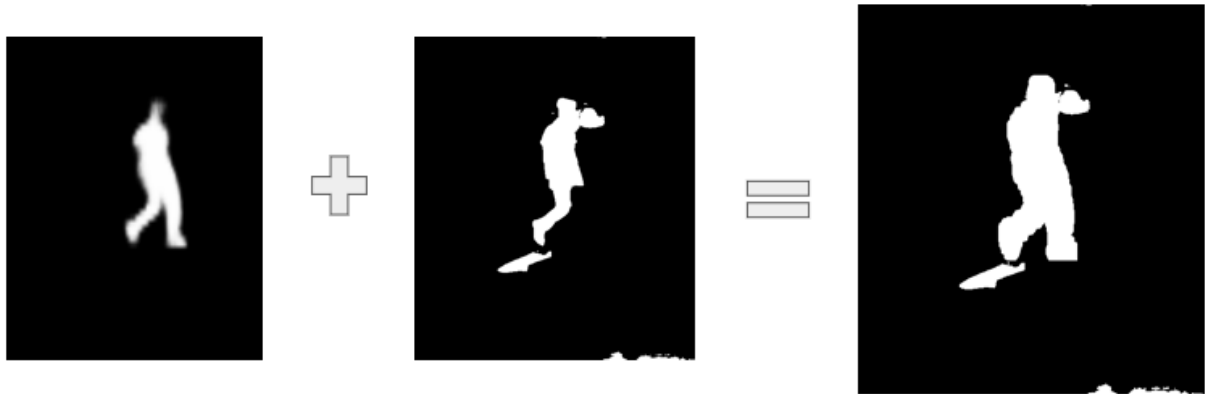
Threshold v.s. KMeans

- We can find that the use of KMeans can better connect the relationship between people and shadows to generate masks.
- Eliminate the effects of background.
- the people behind the original were kept.

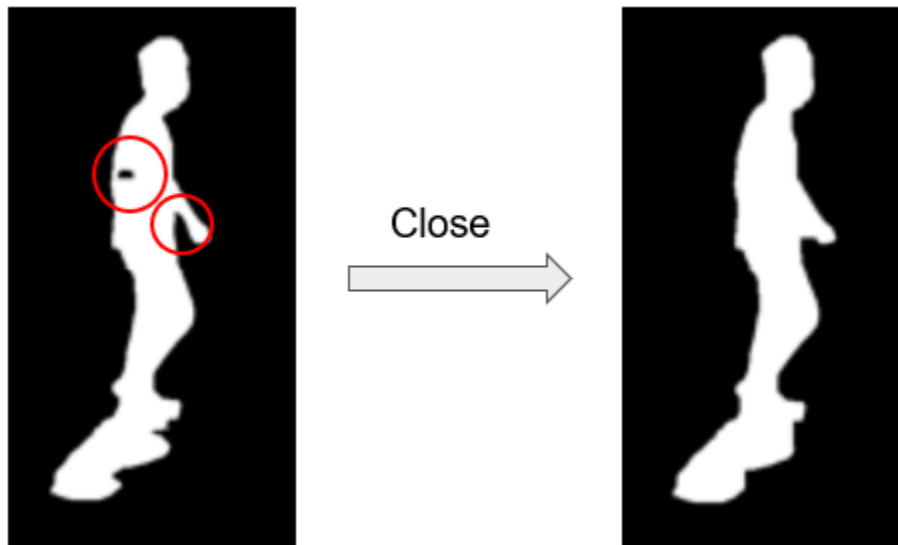


Combine Mask

- Mask RCNN is good at detecting the object and RAFT is good at detecting the shadow.
- Combine the both advantage to produce the mask. if the pixel in one of the masks is white, the pixel in the final mask will be white.

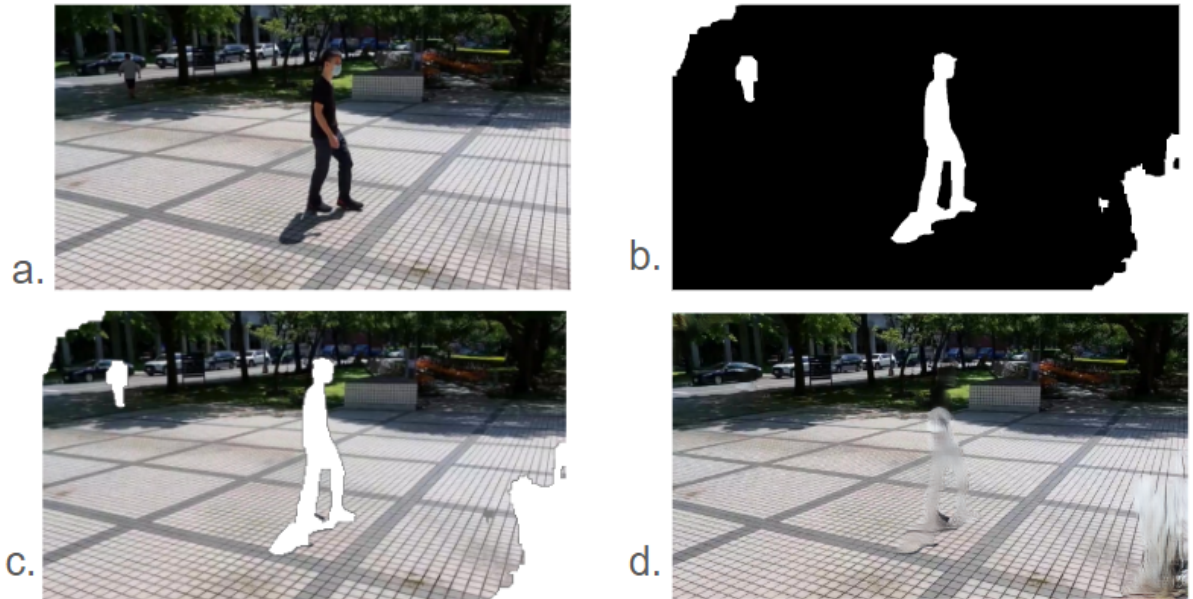


There are some small parts does not be detected. We use the close operation to eliminate them. One of the advantages is close operation can make the mask bigger, and it can cover the whole part of an object and it shadow



Deepfillv2

- We use pretrain model Deepfillv2 to inpainting disappear part by the mask



Discussion

- The mask almost cover the whole object and its effect.
- RAFT can not detect the whole shadow, it may cause by the action of the object.
- Inpainting result is not nature, it need to retrain a personal model for this video's background.

Teacher's Advice

- You can refer to the background picture of the frame before and after to inpaint disappear part by the mask.

Reference

- Mask R-CNN
 - https://www.youtube.com/watch?v=5VLI_gbpocE&t=1s&ab_channel=WilsonHo
 - <https://zhuanlan.zhihu.com/p/142757151>
 - https://pytorch.org/tutorials/intermediate/torchvision_tutorial.html
- RAFT
 - <https://github.com/princeton-vl/RAFT>
 - <https://arxiv.org/pdf/2003.12039.pdf>
- Deepfillv2
 - <https://github.com/nipponjo/deepfillv2-pytorch>
- KMeans
 - <https://www.thepythoncode.com/article/kmeans-for-image-segmentation-opencv-python>
 - <https://www.geeksforgeeks.org/image-segmentation-using-k-means-clustering/>
 - https://docs.opencv.org/4.x/d1/d5c/tutorial_py_kmeans_opencv.html