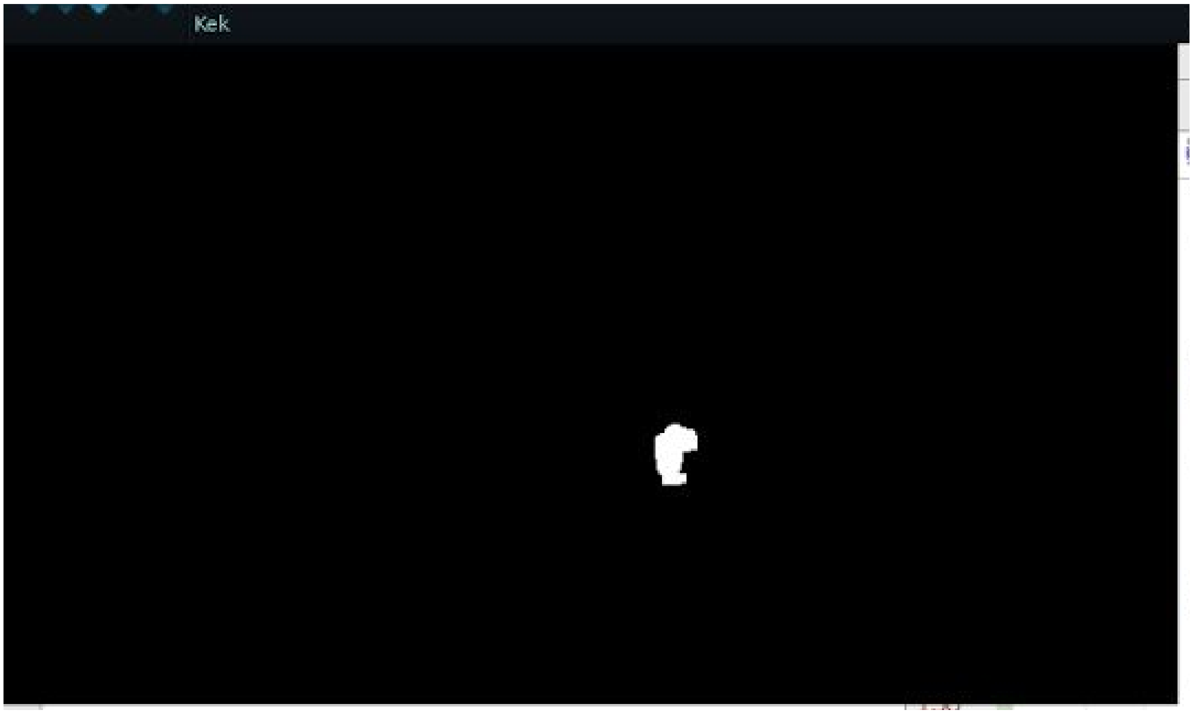
CS4053 - Computer Vision
Assignment 3
Julius Merecas (12313689)

High level description

Change Detection

Look at every 5th frame of the video this is to aid the median background model to do a better job at spotting differences. Blur each of the frames to get rid of noise. For each of the 5th frames calculate two median background models. One of the background models uses a learning speed of 1.005 and the other uses 1.05. This difference in learning speeds will make it so that one MBM says the item is not part of background while the other one says it is part of the background.

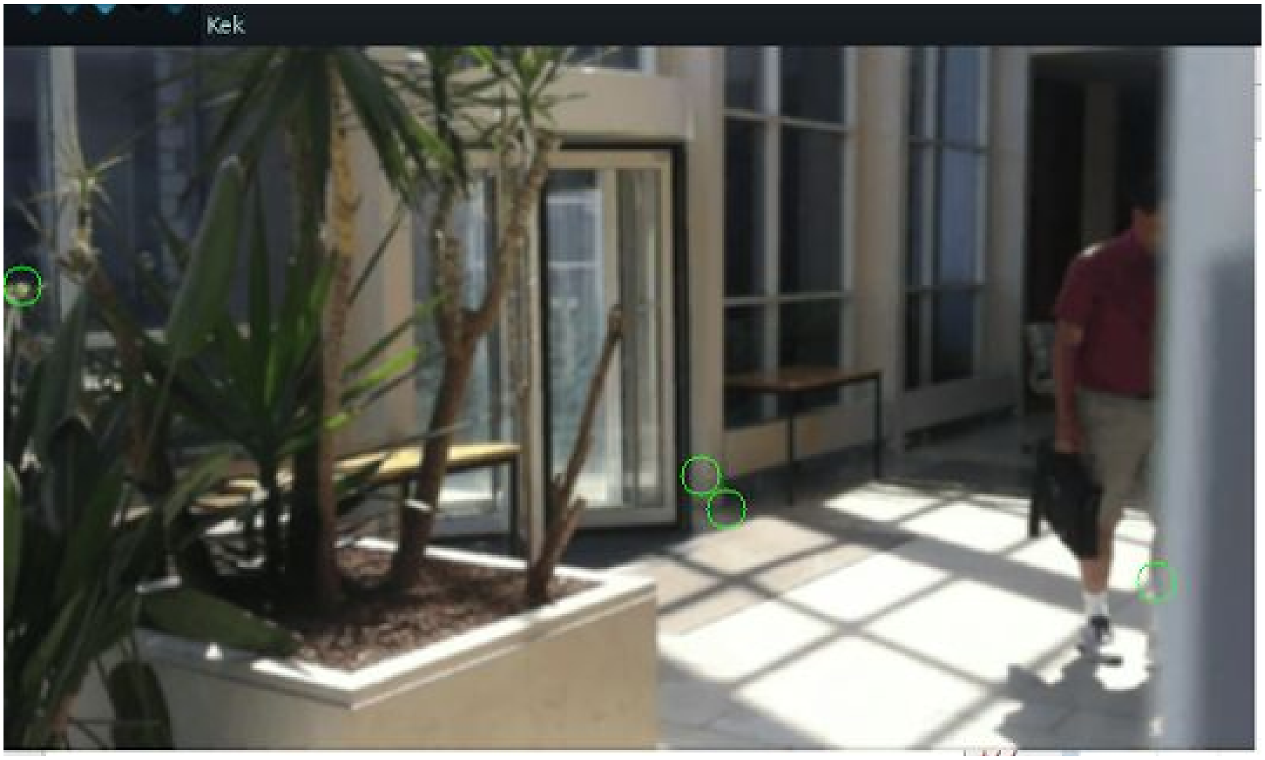
I then calculate the absolute difference of the two median backgrounds. This produces the objects which are part of the background in one but not the other MBM’s.



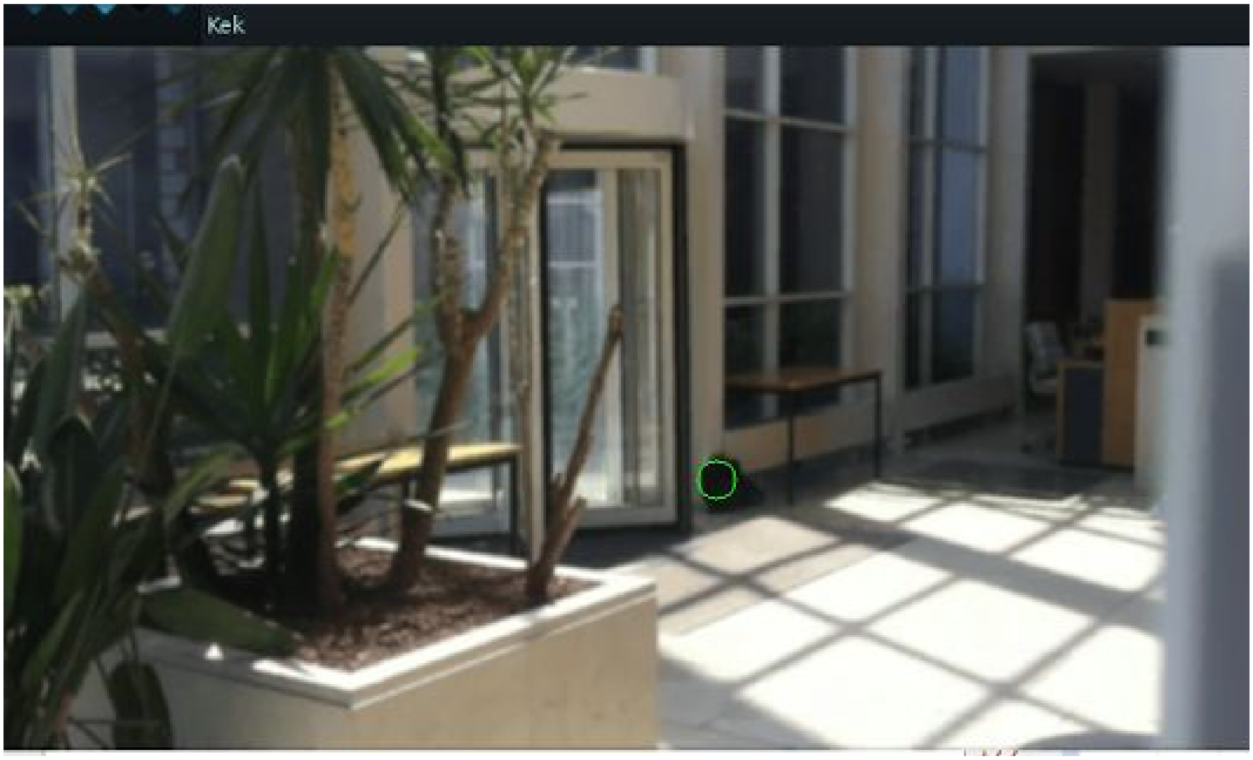
Above image shows the result of absdiff of the two median backgrounds

Once that is done I get the connected components of each of the objects whose area is greater than 25. This ensures that very small objects are left out, such as noise.

I then look for contours that have a centre, between the current and previous frame, within a 5px box. This returns to me a list of contours that were present the previous and current frame (An intersection of the two sets with some possibility of error is a good way to think of this). These are my objects which are added or removed from the scene.



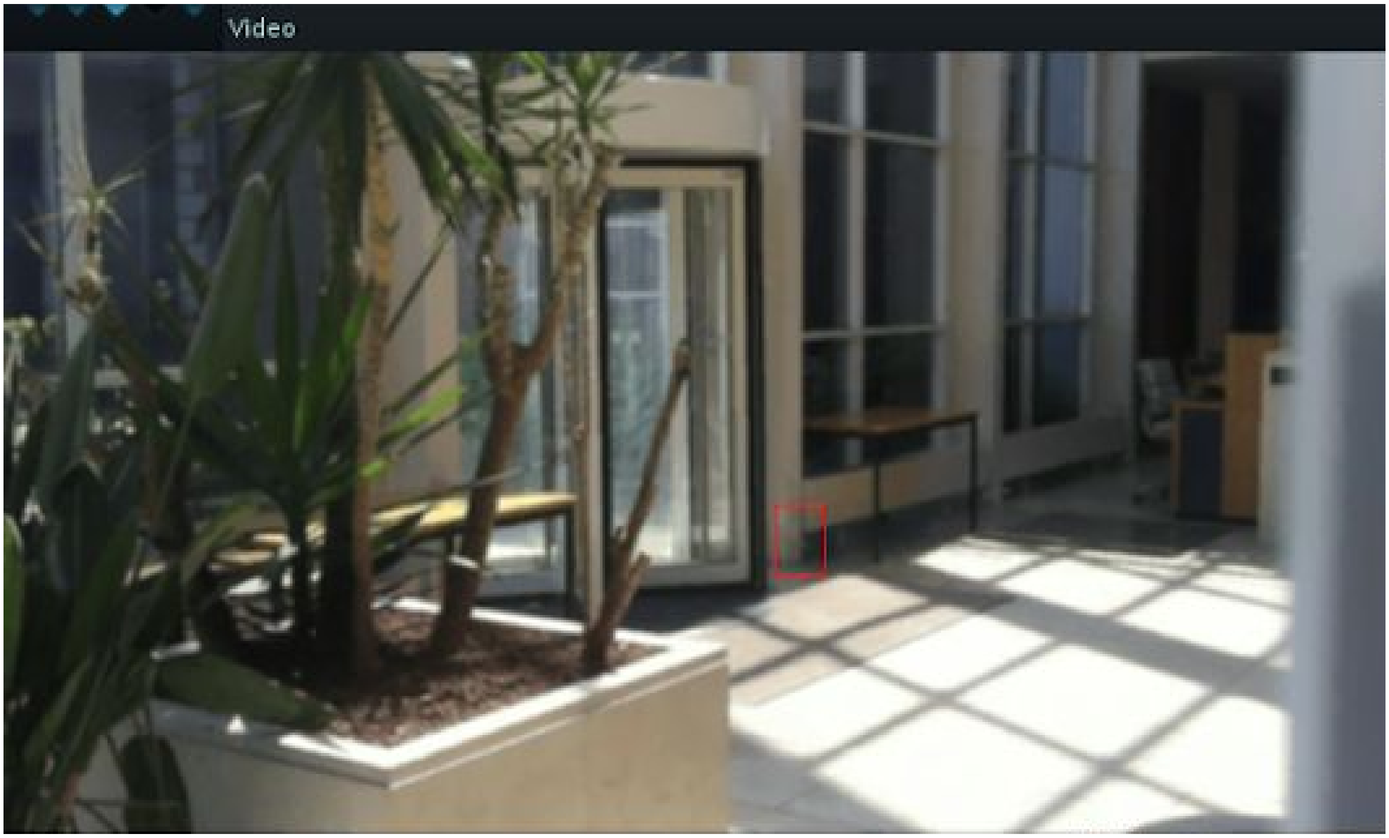
Above Image shows the centres of contours found by the program



Above image shows the union of the centres of contours found by the program

The fact that I look for contours bigger than some size and ones that overlap with the previous frames contours allows me to get rid of Ken walking around in the image (Since his connected components move around too much)

I was unable to determine accurately when an object is added or removed to the image. I kept getting slight flickering of the objects.



My program tells me that it found the following objects in the first video:

```
((355,207)(378,240)) ((355,208)(378,241)) ((355,208)(378,241)) ((355,208)(378,240))
((355,208)(378,240)) ((355,208)(378,240)) ((355,208)(378,240)) ((356,214)(362,234))
((356,214)(362,234)) ((355,208)(379,241)) ((355,208)(378,240)) ((355,208)(378,240))
((355,208)(378,240)) ((355,208)(378,240)) ((357,208)(378,240)) ((357,208)(378,240))
((361,209)(378,222)) ((361,209)(378,222))
```

And in the second video:

```
((290,262)(362,326)) ((290,262)(362,326)) ((332,301)(362,321)) ((332,301)(352,320))
((332,301)(352,320))
```

Dice coefficients are as follows:
Object appears in video 1:
Coeff: 0.0149837
Coeff: 0.0154519

```
Coeff: 0.0154519
Coeff: 0.0149873
Coeff: 0.0149873
Coeff: 0.0149873
Coeff: 0.0149873
Coeff: 0.00257152
Coeff: 0.00257152
Object disappears in video 1:
Coeff: 0.0161486
Coeff: 0.0149873
Coeff: 0.0149873
Coeff: 0.0149873
Coeff: 0.0149873
Coeff: 0.0143158
Coeff: 0.0143158
Coeff: 0.00473076
Coeff: 0.00473076

Object appears in video 2:
Coeff: 0.077901
Coeff: 0.077901
Coeff: 0.0104991
Coeff: 0.00666223
Coeff: 0.00666223
Video ends
```

I used the following formula to calculate the coefficient: where C is the intersection of the two bounding boxes (ground truth and my programs thoughts on where the object is), A and B are the two object bounding boxes.

$$QS = \frac{2C}{A+B} = \frac{2|A \cap B|}{|A| + |B|}$$

Object removal vs. abandonment discrimination

Didn't do this part. Would have compared histograms of the image cropped with the best fit rectangle and the first frame of the video cropped with the same rectangle (except 5px bigger each way). If the histograms are similar then the object has been removed (as the backgrounds have not gained or lost any details) if they are different then the object has been abandoned.