

HW_03_Shah_Niyati

This homework asks us to process a video and create an output image such that non overlapping movements are captured in a single frame.

Algorithm:

User the background subtraction method to separate background and foreground.

Apply noise removal to remove unwanted foreground noise.

Find the biggest contour to that can be the ROI (region of interest) from the foreground.

If the ROI overlaps with the previous ROI.

then copy the ROI to a background image.

Else ignore the frame.

Repeat for all frames.

Apply noise removal to remove jagged edges from the so formed output image.

Things Tried:

BackGround subtraction Model:

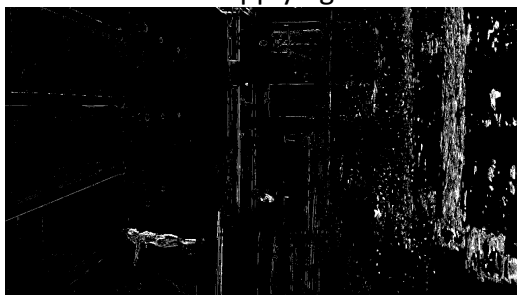
I have used the MOG2 background subtraction method. I got an error while trying MOG method of background subtraction. I dint work on it a lot to see how to resolve it as MOG2 method worked really well.

Here I checked each frame of the video and updated the background model formed by the MOG2 method And find the foreground image mask of the frame. (ie the part where the previous frame and current frame differ and the rest of the background is set to black)

Noise Removal:

I tried using the dilation and erosion on the frames to try and eliminate as much noise as possible. But found that just a median blur and thresholding did the trick and did not erode a lot of the actual movement that I needed in the image.

Noise seen after applying the MOG2 method.



I also tried using the Gaussian blur but it also eroded some parts (the inside of the body where it is black). Gaussian and erosion/dilation caused issues in the later part as other unimportant regions like the movement of the water made a larger contour than the swimmers body.

Gaussian Blur:

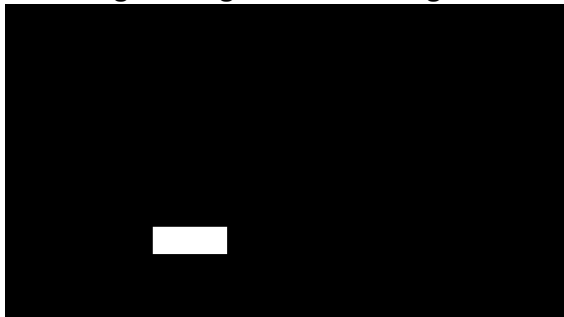


Median Blur and thresholding:



Find Largest Contour / ROI:

Using the findcontour method, I found the largest contour in the frame by its area and filled a bounding rectangle around it as given below:



This rectangle will now help to see if two contours overlap or not.
This rectangle is copied to boundRect Mat object.

Determining the correct ROI for the final output:

Part A: Selection of Correct Motion Change

As mentioned earlier, I applied median blur so that unwanted changes like small but many changes in the movement of water in the diver video and changes due to sunlight in the

pedestrian video can be eliminated and the actual change (the diver and the pedestrian) can be selected as the region of interest.

Without the noise removal, the below image was formed which selected the wrong motion change.



Part B: Selection of non overlapping ROI:

1. Check if contours overlap:

After finding the biggest contour I intersected the previous contour with the current contour and checked if any there was any overlap between them or not.

This method gave a lot of false positives, as the contour formed were not necessarily complete and even if there was an overlap, it did not find it easily.

Other issues were that many noise changes which were bigger then the actual change (diver, pedestrian) were not captured correctly.

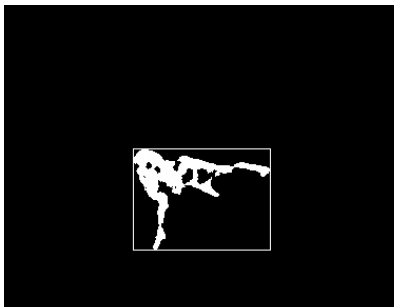
2. Check if bounding rectangles overlap

After finding the biggest contour I drew a bounding rectangle around it, and then checked if the bounding rectangle of the previous frame overlapped with the current frame or not. Again this method was not helpful when with the noise change.

To check overlap,

1. Checked if the point was inside the rectangle
2. Checked intersection of lines of the rectangle
3. Compared leftmost and rightmost points for intersection

Due to some reason, I still got a few false positives, and there were some conditions (described later in Part C) that I discovered later on that were not met.



(Cropped to show the bounding rectangle)

3. Check if filled bounding rectangles overlap



Here I copied the actual foreground in a separate object, calculated the largest contour and then drew a bounded rectangle and filled it in completely.

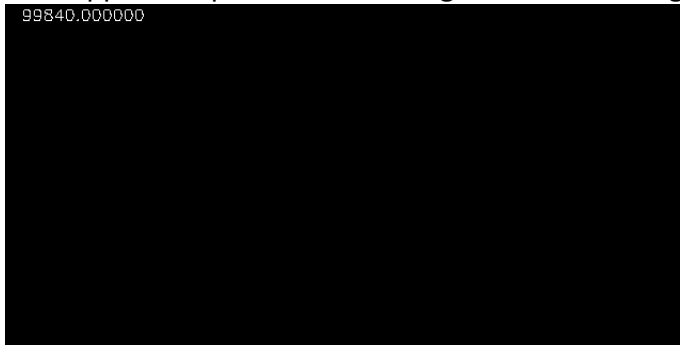
Now when I checked the overlap between them, I could easily find it.

Part C: Selection of precise non overlap

After finding the overlap of images, I saw that in between there is some change due to sunlight/clouds that creates a very huge change in that frame due to light change, this creates noise in the output image as the foreground now consists of new changes that overlap the professor walking. To overcome this, I tried applying a median blur to the image before the foreground retrieval but that ruined created more noise for the background subtraction model.

Other thing I checked was the range of area of the actual ROI (diver/pedestrian) and then only selected those rectangles whose area was between them. (Area selected 1000 and 13000).

But in the pedestrian video, I found that one area which was noise was in the range and it overlapped the pedestrian causing noise in the image,



One of the noise frame.

To avoid this, I tracked a continuous history of the selected movement by overlapping each change that made it into the output image. This was done by the “OR” operator.

But this method did not work, I also tried adding, binary OR and bitwise OR operator, but somehow could not get the continuous overlap which could have avoided this noise in the background.



Adding the selected frames to output:

After using the above methods I selected the frame which did not overlap and copied the foreground image to the output image by using the copyTo function. Where the foreground mask is “Anded” with the actual frame and the part remaining is copied to the output image. Then I update the history object to the current boundRect object.

In this way the process is repeated for each frame.

Other Issues Encountered:

Here when I run the video for the diver image, somehow the after updating the first image, it stops checking the history and I am unable to find a reason why.

I tried just to check if the 1st frame of it somehow differs from the pedestrian video, but it doesn't not.

As described earlier that the two issues, one of diver video and other of pedestrian video caused my program to not work correctly.

As a Final submission, I have created a new program, one where I have preselected the frames and used the copyTo function to add them in the output frame and also done noise removal to get a good image.

Output of both can be seen below:

Diver Video:



Pedestrain Video

681

