COL780: Computer Vision

Assignment 1: Background Subtraction

We present here the details for each of the methods we implemented for background subtraction. We also provide the details on results obtained and links to generated mask videos.

1. Baseline

Methods Used: In Baseline, we used K Nearest Neighbour (KNN) method to do Background Subtraction. For each frame of the video, firstly, we computed the foreground mask by using the KNN method with no shadow detection. Then, we removed the small holes inside the foreground objects by using the morph_closing (dilation followed by erosion) method of Morphological Transformations. Lastly, we removed the noise in the obtained foreground mask by using the morph_opening (erosion followed by dilation) method of Morphological Transformation.

We attempted to design our own background subtractor from scratch using the Adaptive Gaussian Mixture Model. Although its performance was worse compared to the method mentioned above, we have provided the code for this approach as well. And here, we would like to discuss the same. We followed the approach used in the paper Adaptive background mixture models for real-time tracking. Broadly, we maintained a distribution of multiple Gaussians for each one of the pixels in our frame. For every new frame, we checked if our new value for a particular pixel belonged to any one of the previously existing Gaussians in the maintained mixture of distributions (we defined specific thresholds for classifying pixels). If it did, we simply added this value to the distribution and based on the gaussian it belonged to, we assigned it to be a foreground or background pixel. If it didn't, we removed the least weighted Gaussian distribution from the mixture and added a new Gaussian distribution that could accommodate the new pixel value.

Mean IOU score: 0.7048

Link of generated output masks: here

2. Illumination Changes

Methods Used: In the Illumination part, we used the Frame Differencing Method for doing Background Subtraction. For each frame of the video, firstly, we blurred the frame by using a 3×3 kernel and increased the brightness of each pixel of the image by converting it into HSV space and then reconverting to RGB space. This reduced the brightness difference of pixels between consecutive frames. Then, we computed the foreground mask by applying a threshold on the difference in values of pixels between consecutive frames. Lastly, we removed the small holes inside the foreground objects and noise in the foreground mask by using the morph_close and morph_open method of Morphological Transformations.

Mean IOU score: 0.3880

Link of generated output masks: here

3. Camera Shake (Jitter)

Methods Used: After observing that none of the in-built background subtractors worked well on their own, we tried to process our images by applying multiple filters/transformations. We tested our score for multiple approaches. The best performance was observed with the following: we first applied a 5×5 Gaussian blur on the input image. We then used a K nearest neighbour based background subtraction to extract the foreground while neglecting shadows. The initial burring of the image made the camera shake less noticeable to the KNN background subtractor. Once we had our foreground mask, we further applied two morphological transformations (MORPH_CLOSE followed by MORPH_OPEN) using 5 x 5 kernels. This helped us remove noise from the generated mask and gave us finer results.

Mean IOU score: 0.5724

Link of generated output masks: here

4. Changing Background

Methods Used: After observing that none of the in-built background subtractors worked well on their own, we tried to process our images by applying multiple filters/transformations. We tested our score for multiple approaches. The best performance was observed with the following: For each frame of the video, firstly, we applied a 5×5 and then a 3×3 median blur on it. Then, we computed the foreground mask by using the KNN method with no shadow detection. Lastly, we removed the small holes inside the foreground objects and noise in the foreground mask by using the morph_close and morph_open method of Morphological Transformations.

Mean IOU score: 0.4617

Link of generated output masks: here

5. Pan-tilt-zoom Videos

Methods Used: We attempted to design a background subtractor that worked on Pan-tilt-zoom videos. For this, we performed optical flow detection using consecutive frames of the image. Specifically, we used dense optical flow detection. We monitored the movement of each pixel present in the frame as optical flow vectors and analyzed the magnitude and angle of motion. Our method performed comparatively well for small amounts of change in zoom, but could not work for large changes as the pixels could no longer be tracked accurately. Overall, it did not perform well on the sample video, it was very prone to noise which further compromised its performance.

Mean IOU score: 0.0224

Link of generated output masks: here

Submitted by:

Rakshita Choudhary, 2019CS10389

Deepak, 2019MT10685