Computer Vision – Assignment 1.

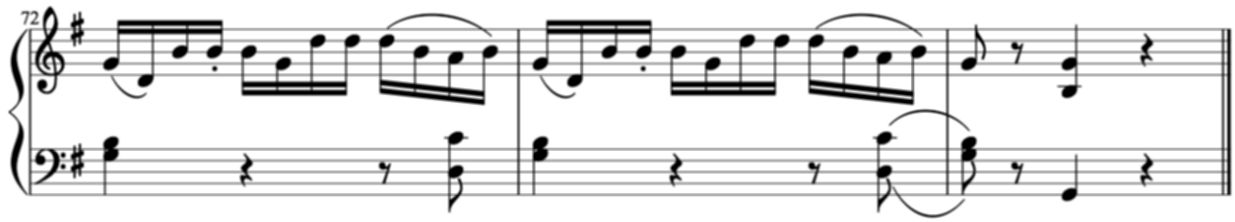
Team Members:
Hao Peng
Venkatesh Raizaday
Abhishek Mehra

Question 1:
We were successfully able to compile the code and execute it.

Question 2:
Convolution was implemented successfully. We have applied brute force method here, and used the filter as was initialized in the skeleton code. The code runs successfully.

NOTE: We have modified this code to calculate hamming distance for part 4. The below output is before the change. (the unchanged part is also kept commented in our code.)

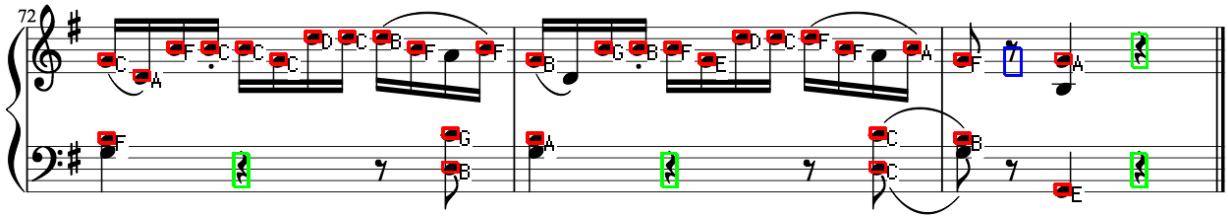
Output Image:



Question 3:
Separable convolution is a faster method of implementing a 2D (nD) convolution by splitting the kernel and having multiple 1D convolution. We were successfully able to implement this part as well. Here in the code we have used a 2D kernel and we were successfully able to convert it to 2 → 1D convolutions.

Question 4:
We have tweaked the convolve general function to calculate the hamming distance between the templates and the image as required. We then compare this hamming distance to a set threshold value and mark these as the points of interest.
Key: For every new image the threshold value differs. (Magic Number ☺). Hack used: So we find out the minimum hamming score and then add the threshold to it, so that we get at least some points of interest.

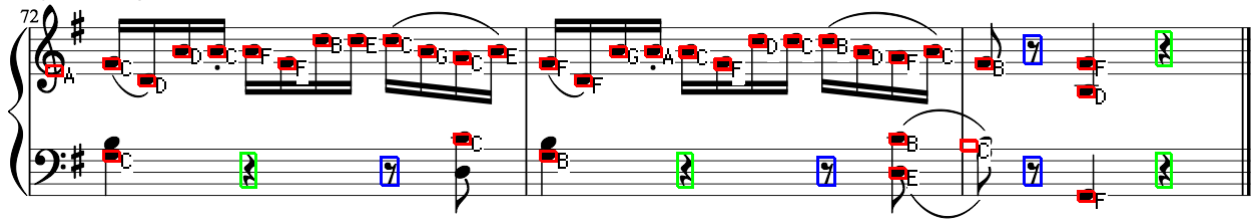
Output Image



Question 5:

For this part, we have implemented Sobel operator and constructed the edge map. Our code calculates the minimum distance for all the nearest neighbors for template matches. Our code runs runs with a better time complexity than brute force and finishes execution in/under 1 minute.

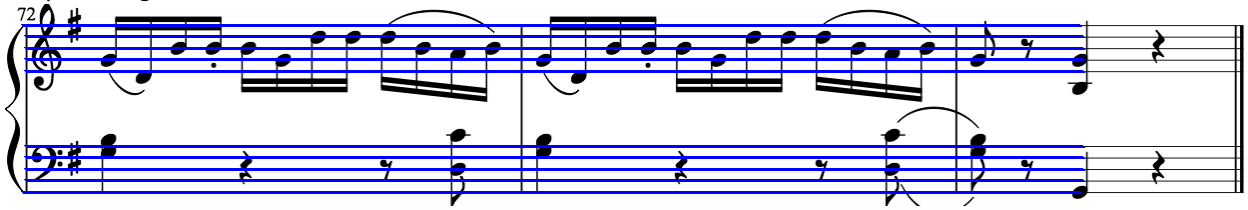
Output Image



Question 6:

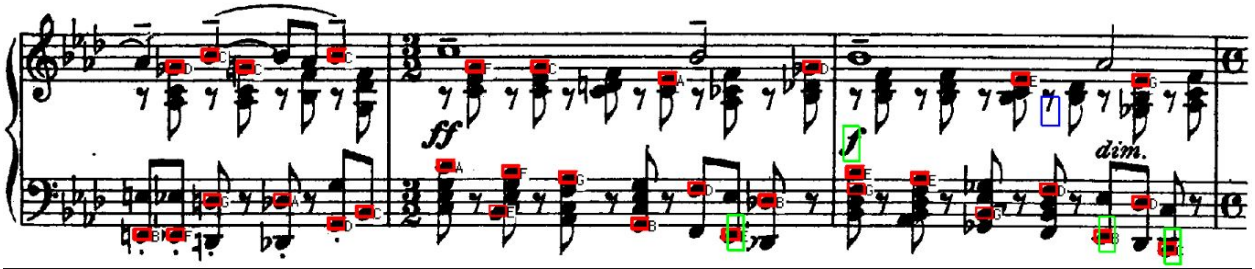
As per the assumption that the lines will be only horizontal and equidistant from each other we performed the Hough transform only on the bins of 0 degree. Whenever there were a bunch of points in the accumulator that exceeded the length threshold that meant that it was a staff.

Output Image



Difficulties Faced:

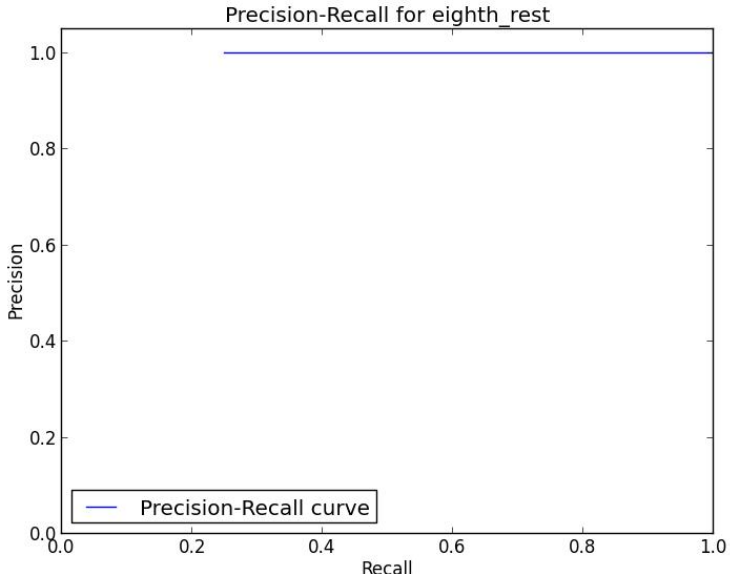
- We couldn't figure out an efficient way to rescale the image to the staff distance which we had calculated in step 6. We tried boosting up the size of the template to match the staff distance, but the results were not promising. Thus, we have rescaled the image using an application and the results are attached below.

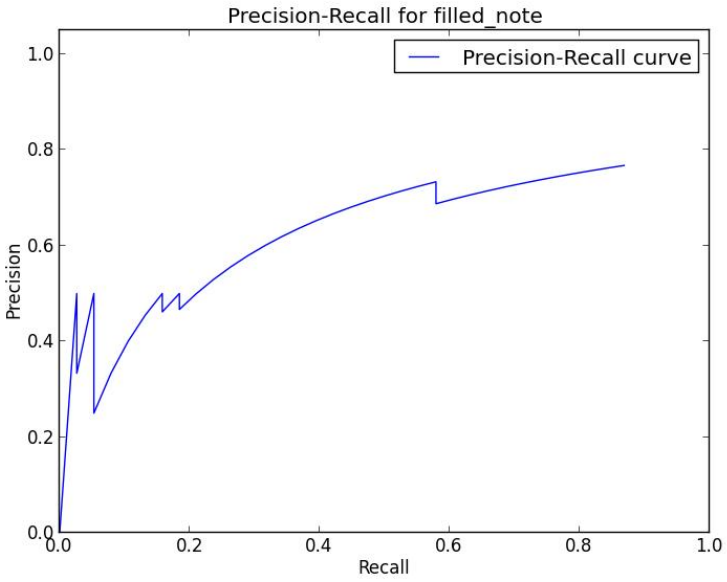


RESULT:

Image1:

Mean average precision for filled_note = 0.576431
Mean average precision for eighth_rest = 1.000000
Mean average precision for quarter_rest = 0.000000
OVERALL Mean average precision = 0.525477





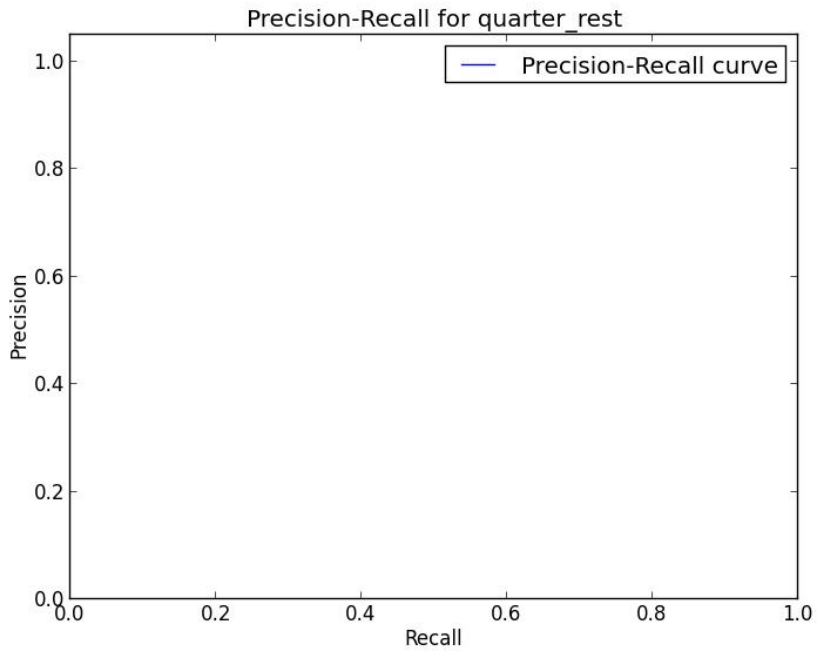
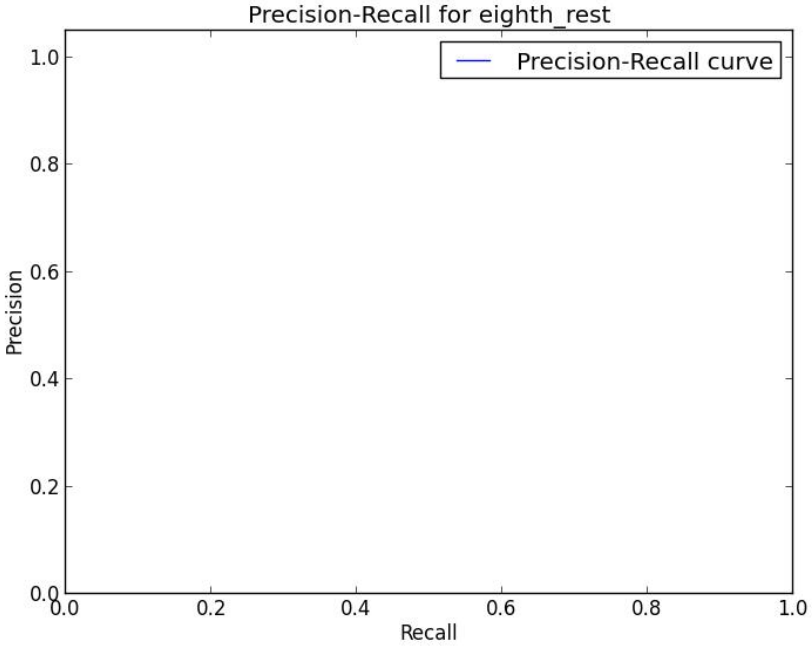
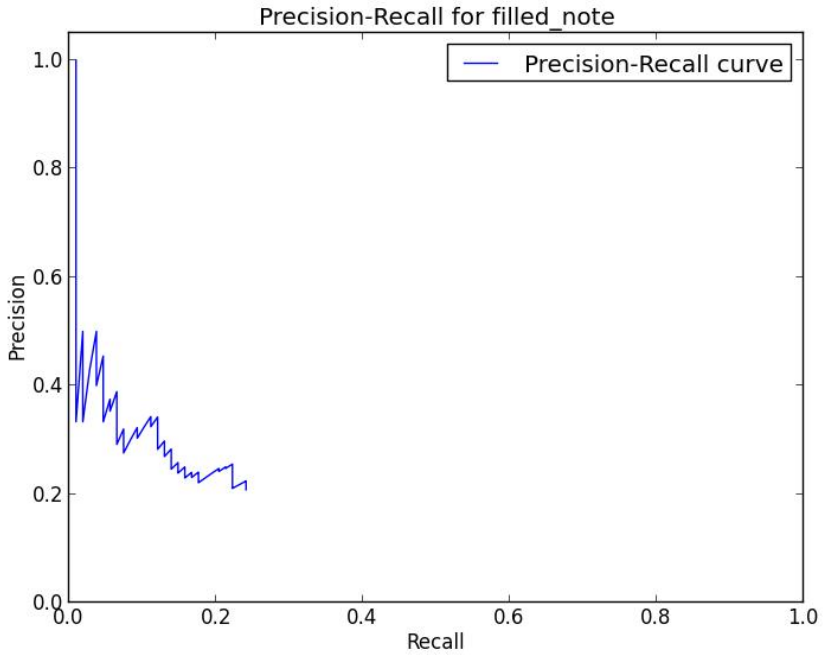


Image4:

Note: We have a lot of detections (as can be seen from the image above) but the score doesn't reflect that. We tried to analyze why this was occurring but couldn't figure it out.

Mean average precision for filled_note = 0.285403
Mean average precision for eighth_rest = 0.000000
OVERALL Mean average precision = 0.142701





Future Improvements:

- Rescaling of template images
- Automated way of figuring out thresholds (Magic Numbers ☺)
- Maybe come up with better algorithms to have a more sound system of OMR.