Name : Haoyuan Zhong

Part-1 : Estimate the albedo and surface normals

1) Insert the albedo image of your test image here:

2) What implementation choices did you make? How did it affect the quality and speed of your solution?

   I stacked the pixel values of all data points (x, y) in a matrix I and call lstsq function once. It is pretty fast.

   For height reconstruction, my implementation is based on dynamic programming. It averages all paths from (0, 0) that either moves right or down. It eliminates the noise caused by just averaging several random paths. The recovered image is smoother. It also reduces the integration error. The computational complexity is O(n^2). So, it is as much fast as average method. The whole reconstruction process can be finished in milliseconds level. This method works perfectly with B05 and B07. Their faces are reconstructed vividly in my opinion.

I also try to compute the paths from different start points, (h-1, w-1), (0, w-1). I notice that the recovered image for B01 and B02 is much better if the start point is (h-1, w-1), i.e. to integrate from left-down corner up to the right-upper corner.

3) What are some artifacts and/or limitations of your implementation, and what are possible reasons for them?

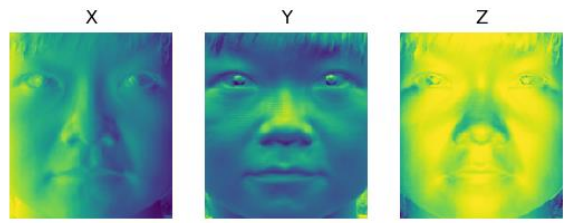
   The albedo near the nose is smaller than the skin in other area. The reason might be that the nose blocked some light from the other side which make the skin behind it darker.

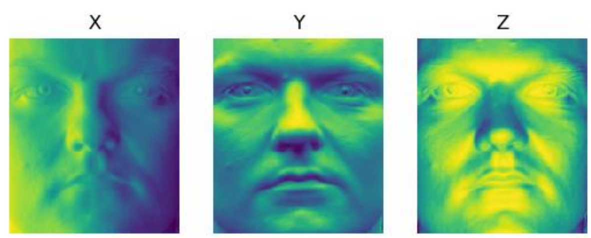
   For the 3D image, there are many waves in the face for row-first implementation, huge error for column-first implementation, or average integration. The is due to the error of the approximation to the integration.

   When I use random integration, I notice that the recovered image quality of B01 and B02 is not as good as B05 and B07. Both of the two faces have some 'waves' in the right side of their mouths. And the right-bottom corner's height seems larger than the other area. I guess it is because that I use integration paths from left-upper corner down to the right-bottom corner. As the integration path increases, the error accumulates.

   I further observed the albedo of all four images. I found B01 has more black spots all around the face and B02 has some stubble around its mouth. They both have artifacts around their mouths (skew and waves). So, I guess the skin condition and stubble might also be a factor effects the outcome.

4) Display the surface normal estimation images below:

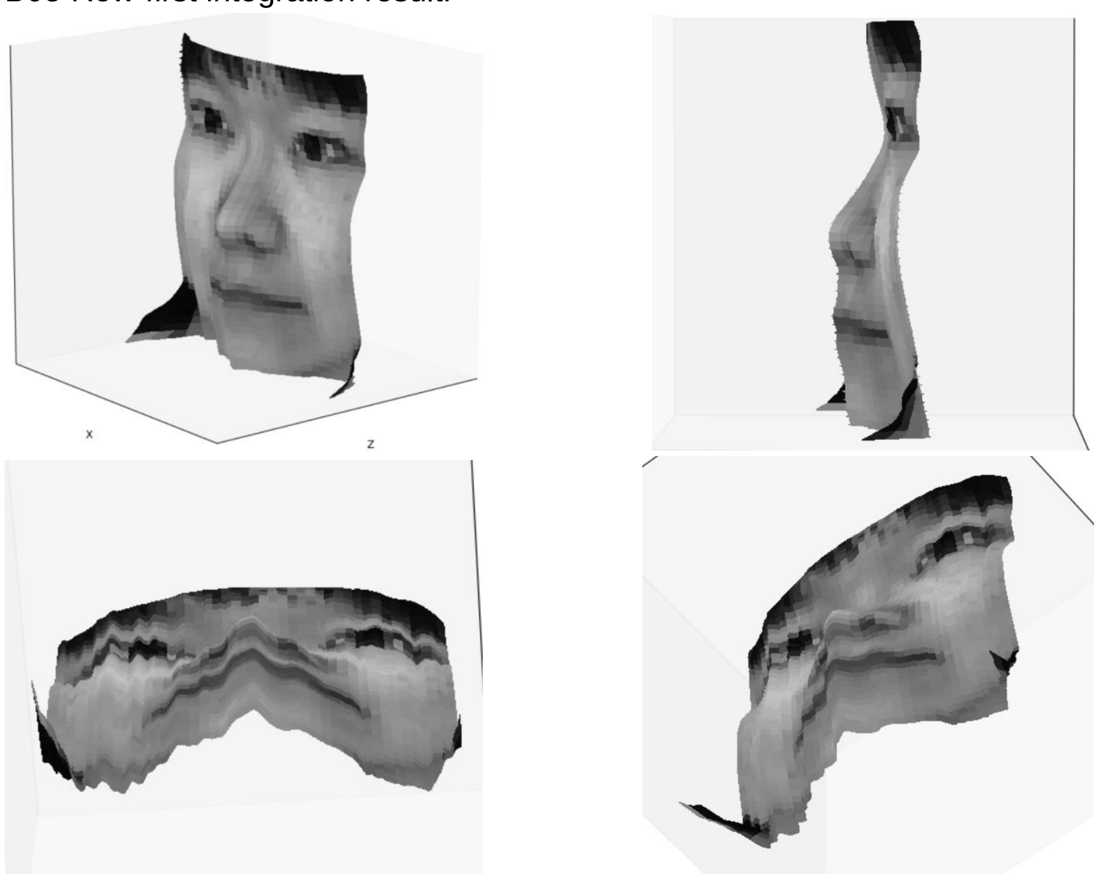




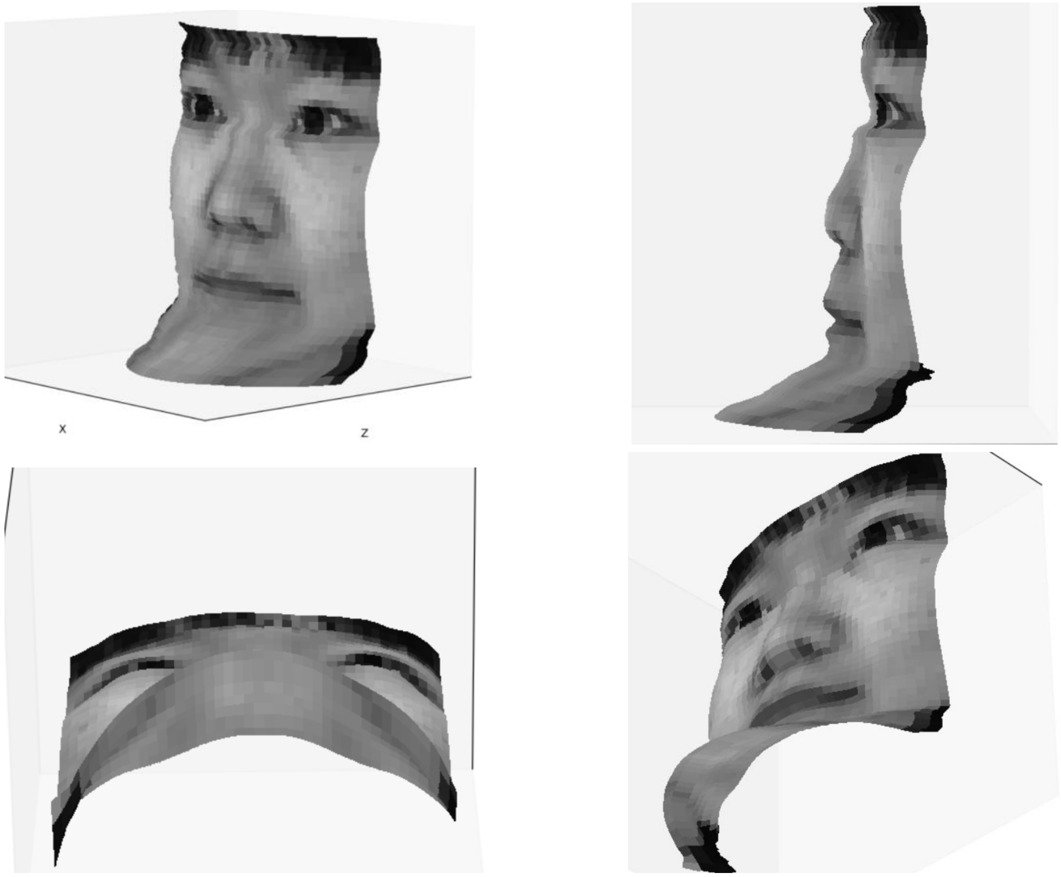
Part-2 : Compute Height Map

5) For every subject, display the surface height map by integration and display. Select one subject, list height map images computed using different integration method and from different views; for other subjects, only from different views, using the method that you think performs best. When inserting results images into your report, you should resize/compress them appropriately to keep the file size manageable -- but make sure that the correctness and quality of your output can be clearly and easily judged.

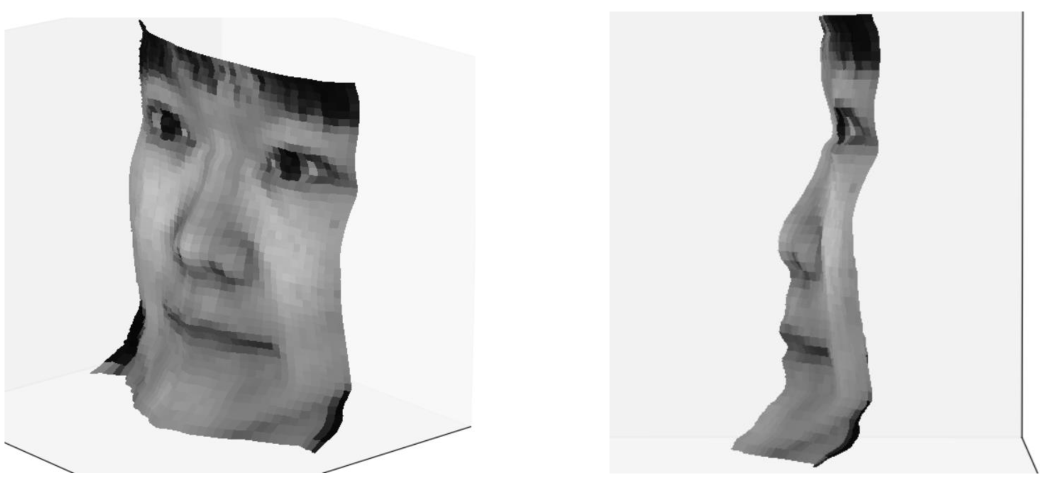
B05 Row-first integration result:

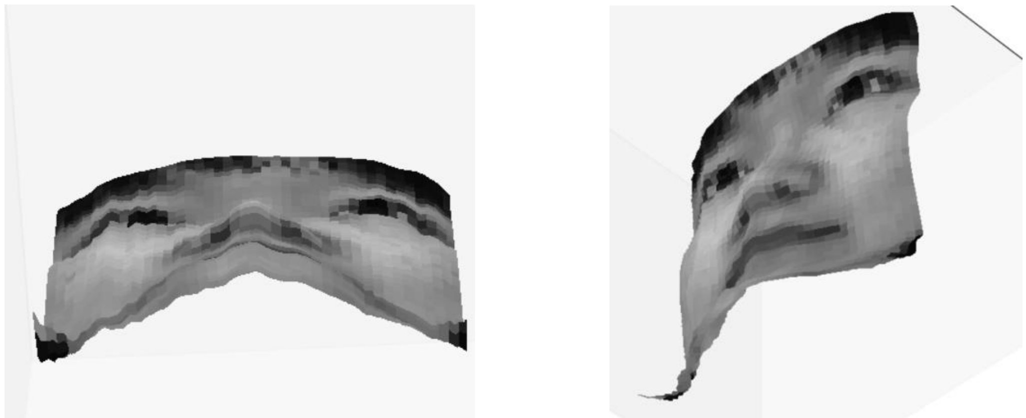


B05 Column-first integration:

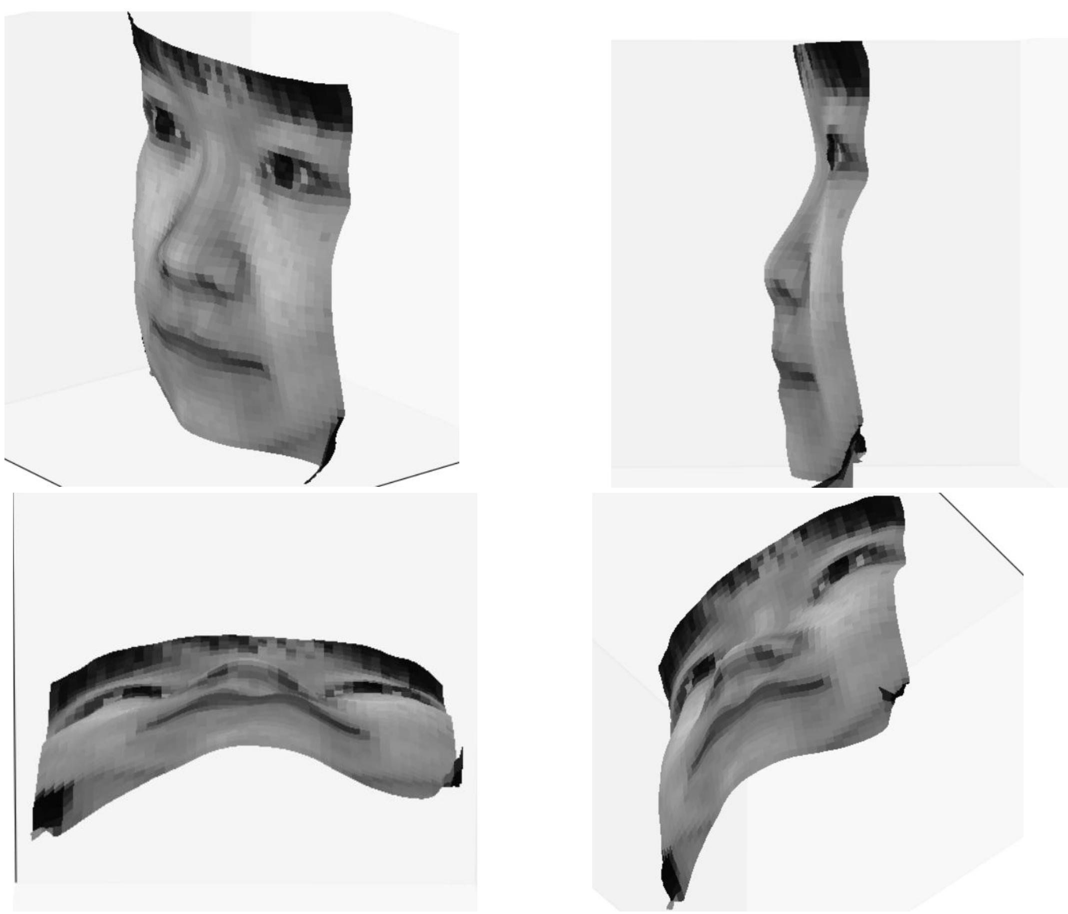


B05 Average integration:

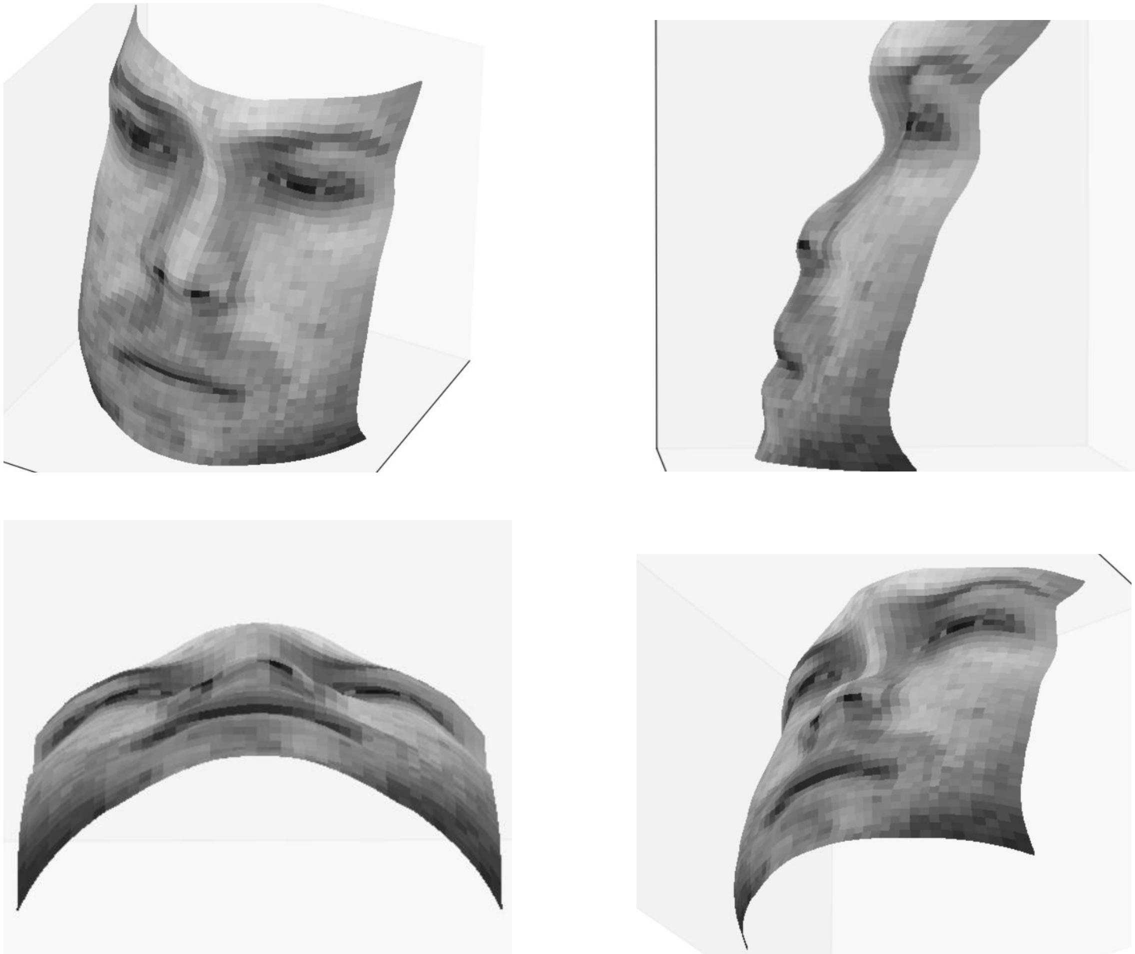




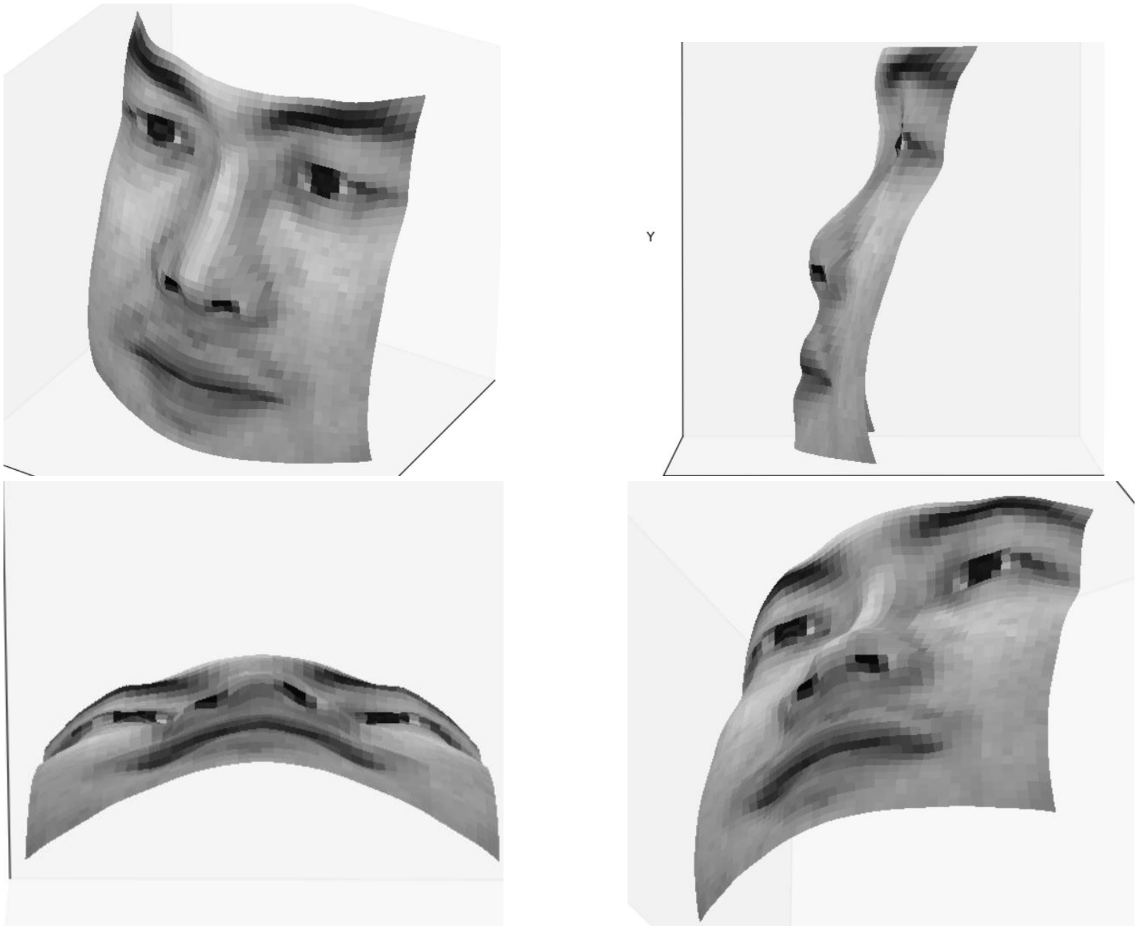
B05 Random integration: start from (0,0)



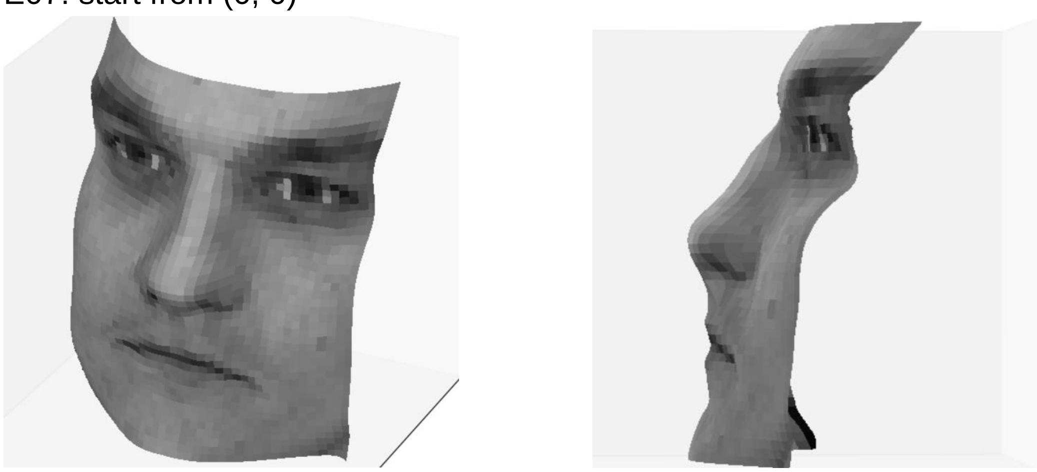
B01: start from (h-1, w-1), and average the sum of height and fliplr(height)

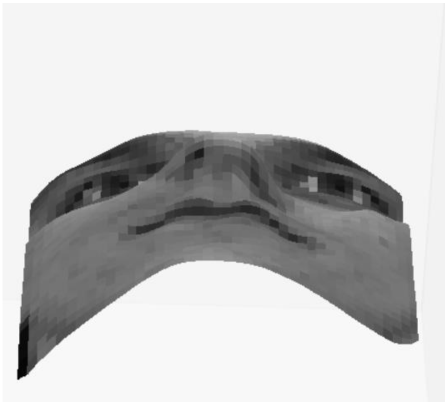


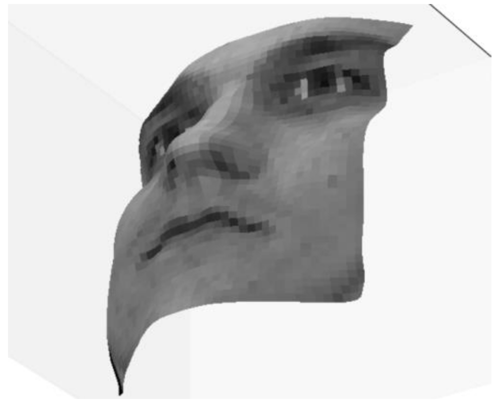
B02: start from (h-1, w-1), and average the sum of height and fliplr(height)



E07: start from (0, 0)



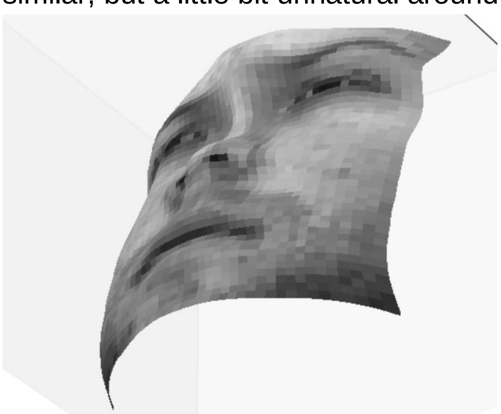


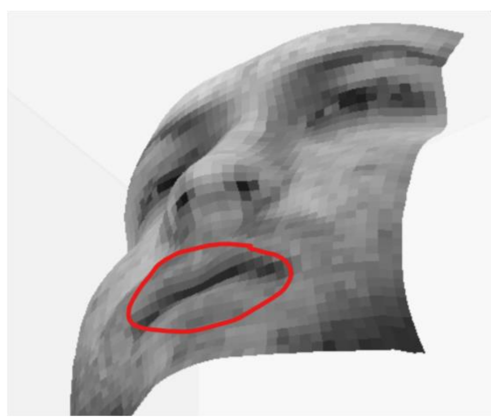


6) Which integration method produces the best result and why?

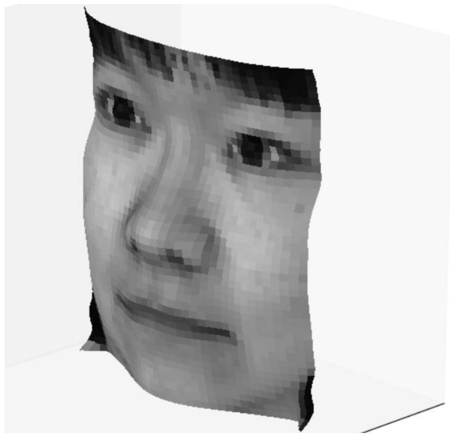
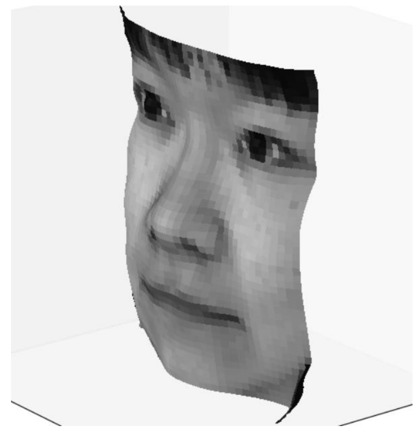
Randomly integrating multiple paths produces the best result. It decreases the integration error. So, the 3D image becomes smoother. It also has less moves. Some details in the recovered image (like eyes) become more natural.

For B01 and B02, I average all paths started from (h-1, w-1), and average the output height and fliplr(height). I think the average is necessary because the face should be symmetric. As the integration path becomes longer, error accumulates and I notice the far side (if it integrates started from (0, 0), then right-bottom side is far side, vice versa) tends to be higher. And I also notice the waves near the mouth can be eliminated by averaging the left and right half. I decide to integrate from (h-1, w-1) simply because I want the area nearby the mouth has less error. The outcome by starting from (0, 0) is similar, but a little bit unnatural around the mouth.





Similarly, it works similarly for B05 and B07. But since the result for them before averaging is good, I cannot observe much difference.

7) Compare the average execution time (only on your selected subject, “average” here means you should repeat the execution for several times to reduce random error) with each integration method, and analyze the cause of what you’ve observed:

Integration method	Execution time
random	3329ns
average	503ns
row	10ns
column	10ns

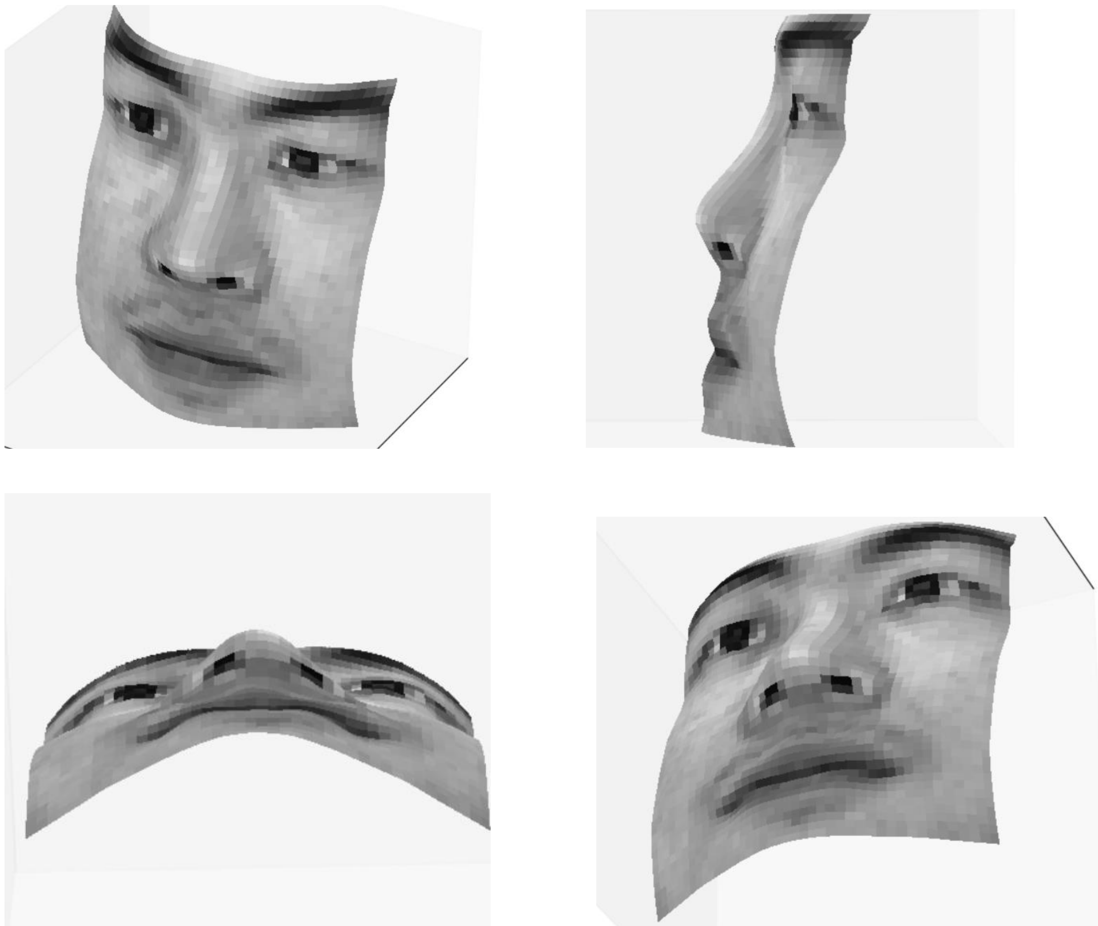
Part-3 : Violation of the assumptions

8) Discuss how the Yale Face data violate the assumptions of the shape-from-shading method covered in the slides.

   Firstly, the faces are not perfect lambertian objects. And the image received by camera is not orthographic projection. Moreover, the local shading model is not valid because some points (like the area nearby the nose) can receive the light reflected from nose.

9) Choose one subject and attempt to select a subset of all viewpoints that better match the assumptions of the method. Show your results for that subset.

   I delete all the viewpoints with A >= 90 or 30 <= A <= 60. I test it with B02, and get the almost the same result as before, but the amount of computation is much less.



10) Discuss whether you were able to get any improvement over a reconstruction computed from all the viewpoints.

    Compared to the previous result, we can see that it gets a little higher nose. The profile also has some subtle differences. His jaw is also not that prominent.

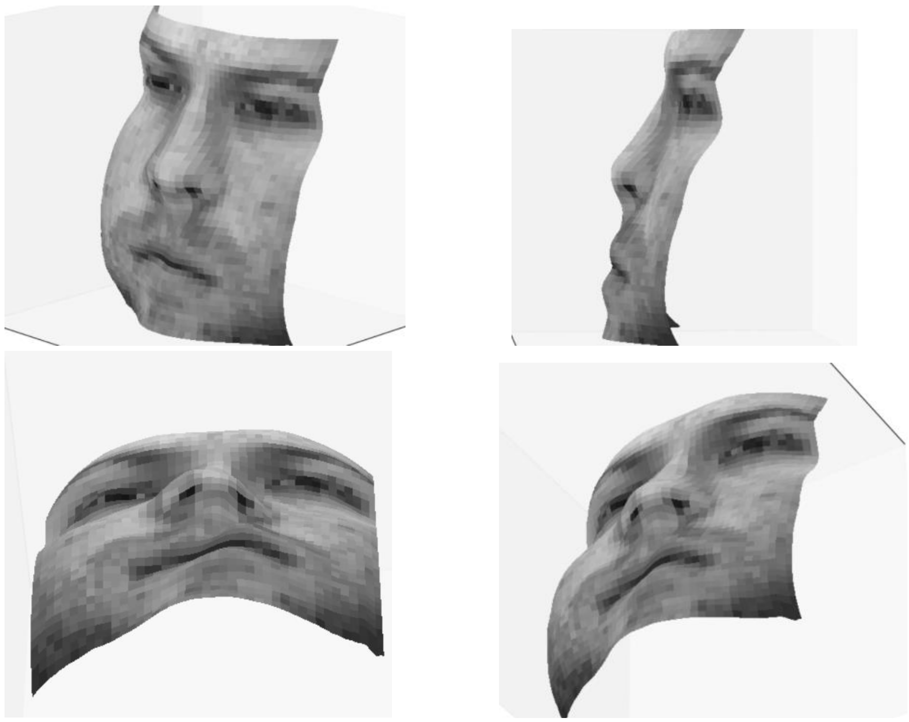
    It hard to tell whether it is am improvement or not since we are not given the exact outlook of this man. But I think his nose and jaw are more natural.

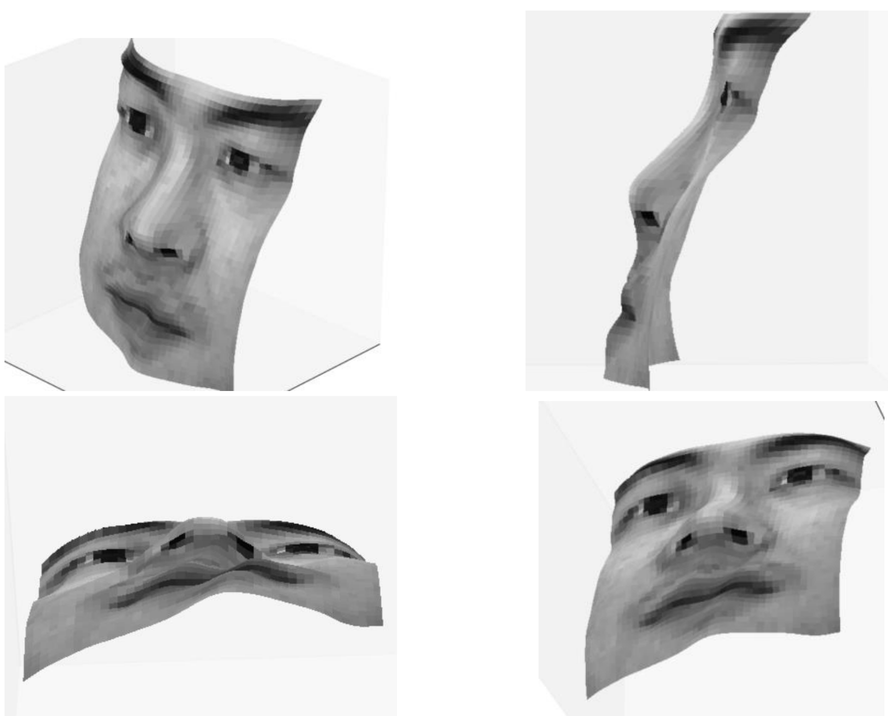
Part-3 : Bonus

Post any extra credit details/images/references used here.

As illustrate above, I implement the random integration with DP that averages all paths move bidirectionally (right and down if start point is (0, 0) or left and up if start point is (h-1,w-1)). It is very fast. The outcome is good as well. I do not use random start point because I found that all random start points at (0, x) or (x, 0) can be decomposed by integrating from (0, 0) to (0, x) or (x, 0) first and then from the intermediate point to the target point. Similarly, any integration started from (h-1, x) or (x, w-1) can be seen as the integration started from (h-1, w-1).

The initial result I got for B05 and BO7 is great, shown as above. But it not good for B01 and B02.





Then, I tried to integrate from different start points, like (0, w-1), (h-1,w-1). The later one works better for these two faces. Noticed the waves and asymmetricity near the mouth, I tried to correct fx by making it symmetric. I also tried to modify fy in some different ways. But none of them works. Finally, I realize I can correct the height output by averaging the sum of itself and its left-right flipped matrix. The result is pretty good.