Computer Vision

Alessia Paccagnella

December 3, 2019

1 Shape from Silhouettes

1.1 Silhouette extraction

To extract silhouettes from the provided images, it was used a thresholding technique.
Only the pixels that have a higher value than the silhouette threshold are considered. For this reason, there might be some areas with pixels of higher intensities that are not of our interest, or some areas with pixels of lower intensities (like shadowed-areas on the statue) that may be our interest to extract. This is one of the problems of thresholding evaluation.
I did a lot of trials, choosing the value of 110 as it was giving me the best result compared to other thresholds that were extracting too many parts of the image or too less.

Figure 1: On the left, pixels of intensity 1 and 0 are shown. On the right, pixel that are higher than the threshold are shown with original intensity.

1.2 Volume of interest

The problem here is to define the bounding box for the statue. After several trials I choose

`bbox = [0.2 -0.2 -1.8;2.7 1.6 2.5].`

As asked in the assignment, I chose a grid of `64 x 64 x 128`.
On the following page results are shown.

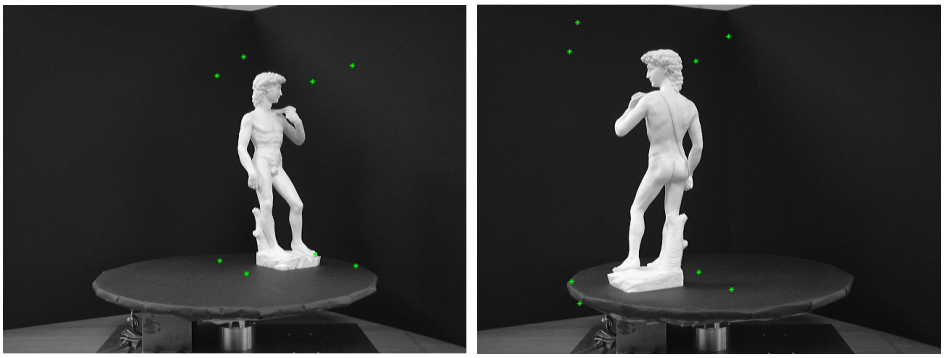

Figure 2: Some of the images from cameras. The box corners are displayed in green

1.3 Visual hull

As described in the assignment, in order to complete this task I iterated over every point `(x,y,z)` of the volume and transformed it into 3D coordinates with the `T` matrix given. I projected the points with the projection matrix into the image plane, for every image. I checked whether the point projected into the silhouette and in that case I increased the value of the voxel by 1.
The provided code extracted an iso-surface from the volume based on the threshold, which I did not changed. Here results can be seen.

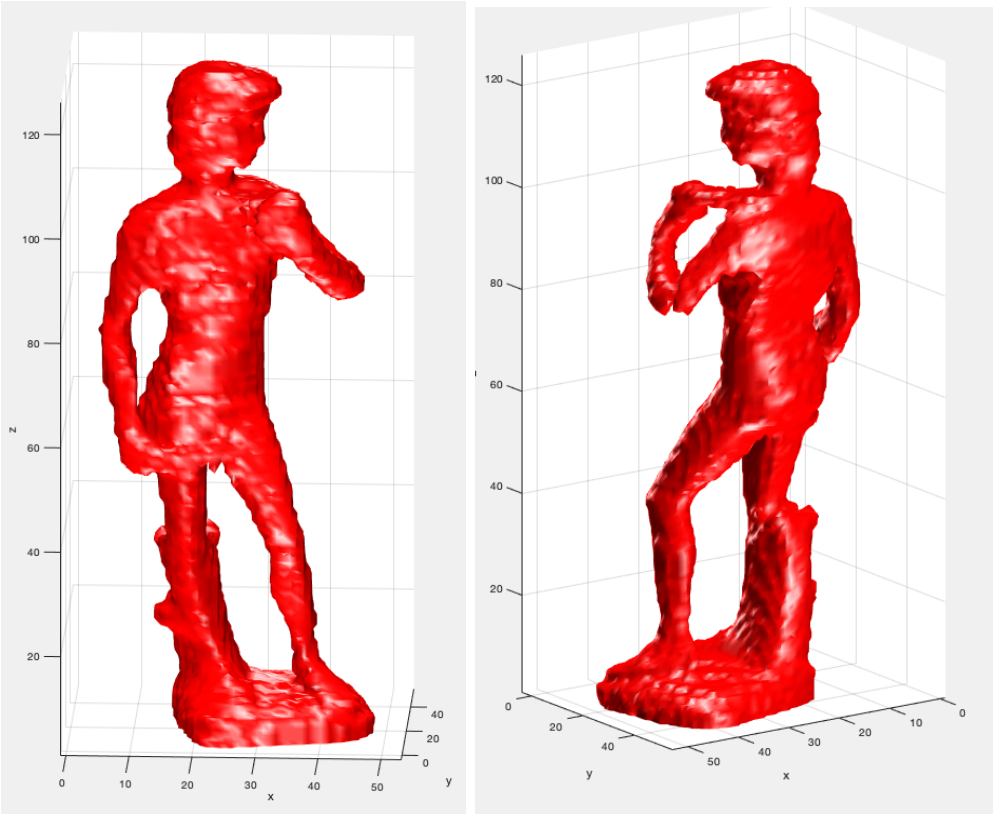

Figure 3: 3D iso-surface

1.4 Improvements

The silhouette is the contour of an object in an image and it gives us information about the 3D shape of the object.
Given a single silhouette image, we project the 3d coordinates of the volume into a 2d space using the camera parameters, and we check if the the silhouettes contains those points. Only voxels that lie inside all silhouette volumes remain part of the final shape.
I can think of several drawbacks of this approach.
The first drawback of this technique that I notice could be the use of thresholding algorithms, cause this affects the accuracy of the silhouettes extraction. In our case, the intensity of the pixels of the statue has a very different value than the ones of the rest of the scene, as they are white while the background is very dark. However, if we did not have a situation where the contrast was this evident, the thresholding technique would not have been optimal, leading to a probably bad extraction.
This method also suffers the visibility problem: it is needed that every camera that we use contains the object without it being occluded. In fact it can only extracts object that are in the intersection of the camera view cones. Moreover, the result also depends on the number of views that we have. If we have a low number of cameras, the approximation could have a low accuracy level.
Another problem of this approach is the reconstruction of some ghost objects which do not contain real objects, especially when there are multiple real objects in the scene. A possible solution to this drawback is proposed in the paper ***Silhouettes Fusion for 3D Shapes Modeling with Ghost Object Removal***, where it discusses a new theoretical approach that identifies and removes ghost objects. This technique leads to more accurate reconstructed shapes.
A possible improve of the *Shape From Silhouettes* approach could be the calculation of depths by combining methods used in stereo geometry. This way we could change the value of the voxels when needed. Also, as the paper ***3D Reconstruction by Combining Shape from Silhouette with Stereo*** says, this could lead to a significant improvement of visual hull even if the number of cameras was limited.