3D Face Reconstruction

***Abstract*—This paper describes the methods for stereo 3D face reconstruction. The 3D mesh is a point cloud created from the disparities between images of the same face from varying angles. The results show that the methods allow a good mesh generation.**

***Keywords*—3D face reconstruction, depth estimation, disparity mapping, stereo vision**

I. Introduction

3D face reconstruction is the process of creating a 3-dimensional face mesh from multiple 2-dimensional face images.

Through 3D reconstruction one may identify the 3D profile of any object as well as the 3D coordinate of any point on the profile. 3D object reconstruction is a broad scientific topic and a basic technology in many industries, including Computer Aided Geometric Design (CAGD), computer graphics, computer animation, computer vision, medical imaging, computational science, virtual reality and digital media.

This process can be accomplished through active and passive methods. Passive 3D reconstruction methods do not meddle with the rebuilt item; instead, they employ a sensor to detect the illumination reflected or radiated by the object's surface in order to deduce its 3D structure through image interpretation. On the contrary, active methods use an artificial energy source with known patterns to replace the second camera [3]. In passive 3D reconstruction, Monocular Vision methods use single images to construct 3D meshes: It oversegments each image into tiny chunks and deduces the 3D location and orientation of every chunk at the same time, using shading, texture and silhouettes to create the 3D mesh [9]. In contrast, Stereovision uses the disparities between points belonging to the same object created by the perspective discrepancies between a stereo pair of pictures [10]. Furthermore, we can differentiate between methods with single images as input and multiple images as input: In binocular vision we retrieve the depth of a point seen on two photos by using two images of the same scene obtained from slightly different points of view. First, a matching set of points in both photos is identified. The depth of a point on the pictures can then be determined using the triangulation method [4]. In parallax motion, the camera's position in relation to the scene gives vital signals about depth perception. Objects in close proximity to the camera move faster than objects further away. Structure from motion is the process of extracting 3D structures from video [7]. In image blurring, the blur/sharpness along the occlusion boundaries is used to assess depth [8]. In silhoute vision, background subtraction is used to segment the silhouettes of the target items in each image. The silhouettes are returned to a common 3D space, with projection centers identical to the camera positions. A cone-like volume is created by back-projecting a silhouette. The optical hull of the target 3D object is formed by the intersection of all the cones, which is frequently processed in the voxel representation [1]. Atmospheric scattering is a technique based on the notion that minuscule particles in the atmosphere modify the power and direction of light when it passes through it. Objects closer to the camera appear crisper, while those further away appear fuzzy [12]. In Shape from shading, an object's surface normal is calculated dependent on how it reflects light. In this approach, the amount of light reflected by an object's surface is determined by its orientation and was first used by Woodham in 1980. When the input is a image, it is referred to as shape from shading, and B. K. Horn P. investigated it in 1989. Since then, photometric stereo has been used to a wide range of conditions, including non-Lambertian surface finishes and extended light sources [11].

This paper will be a demonstration of 3d reconstruction by using the passive, multi image stereo vision technique to reconstruct a 3D face mesh. Section II discusses the materials used and details the steps taken to reconstruct the 3D face mesh. Section III displays the results, while Section IV interprets these results and discuses limitations. Finally Section V concludes the paper with a recap of the overall implementation and the importance of the work..

II. Methods and Materials

A. Materials

The project has been fully performed using Matlab R2021b, with a Stereo Camera Calibrator Application of Computer Vision toolbox used for the image pre-processing steps. The input for the project includes three subjects, each of whom has three different sets of pictures taken in variate angles covering the subjects' face features, including left-camera, middle-camera and right-camera images. Two of these three subjects also include different sets with different face expression (total 11 sets, each of this has 3 camera viewpoints). The input data also comprises two sets of checker-board pictures for calibrations; each set consists of 20 pictures from all three camera. In this project, we will use the 1st calibration folder to calibrate our images.

B. Methods

In stereo vision, 3D depth information is extracted from multiple digital images. Through the varying angles between images, the depth can be calculated through triangulation. Figure 1 showcases this process in a setting where the two images are fully aligned. This means that the two image planes marked in red are parallel to the baseline b and their optical axis is orthogonal to the baseline. It is then possible to calculate

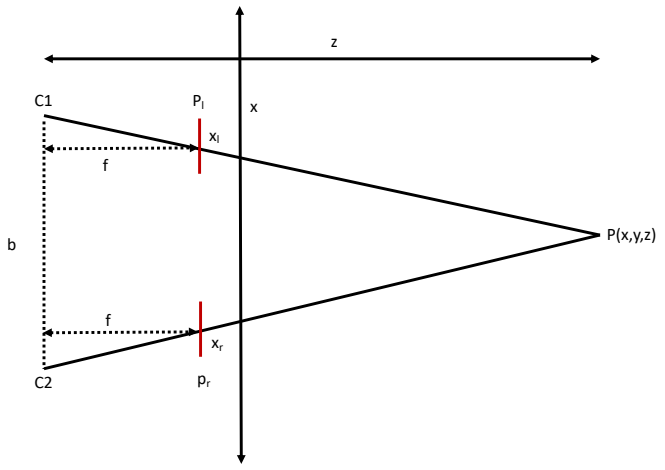

Fig. 1: Fully aligned stereo vision system in geometric representation. $P(x, y, z)$: 3D point in scene, p_l, p_r: projected point on image plane, f: focal length, b baseline, C_1, C_2: focal point of camera. Note that the Y–axis is orthogonal to the page.

the depth z by placing a triangle between the two camera focal points C_1 and C_2 and the actual 3D point P in the scene:

$$\frac{z}{f} = \frac{x}{x_l} = \frac{x-b}{x_r} \tag{1}$$

$$d = x_l - x_r = \frac{fb}{z} \tag{2}$$

where d is the disparity of the point p between the two images. We can construct a 3D point cloud or surface mesh, using this depth information. This process relies on the mathematical subbranch of epipolar geometry.

To extract the depth information, it is necessary to first fully align the cameras in a process called stereo rectification. Then we can estimate the disparities with Stereo Matching, yielding us a disparity map that can then be used to create a point cloud and a surface mesh. Lastly, it is possible to obtain the final result by merging two point clouds generated from different images. The overview diagram of our 3D face reconstruction method is shown in figure 2. The process of the project can be divided into 3 sections: Pre-processing, Points cloud generations and ICA Merging. Details of each of these steps can be found in the following sections.

1) Background removal: Firstly, all the important information for the reconstruction will be chosen from the pictures before continuing with the analyzing steps. K-means background removal has been used in this study, as a method to discard all the non-relevant features from our input pictures. With this method, the images are dissected into multiple parts: Background (Yellow), Skin (Beige), Eye and Shirt (Dark black) (Figure 3). Color-based segmentation with k-means clustering is used to identify the primary colours in the images. The brightness dimension of the pictures will not be considered in this process, as the pictures will be converted in lab color space format.

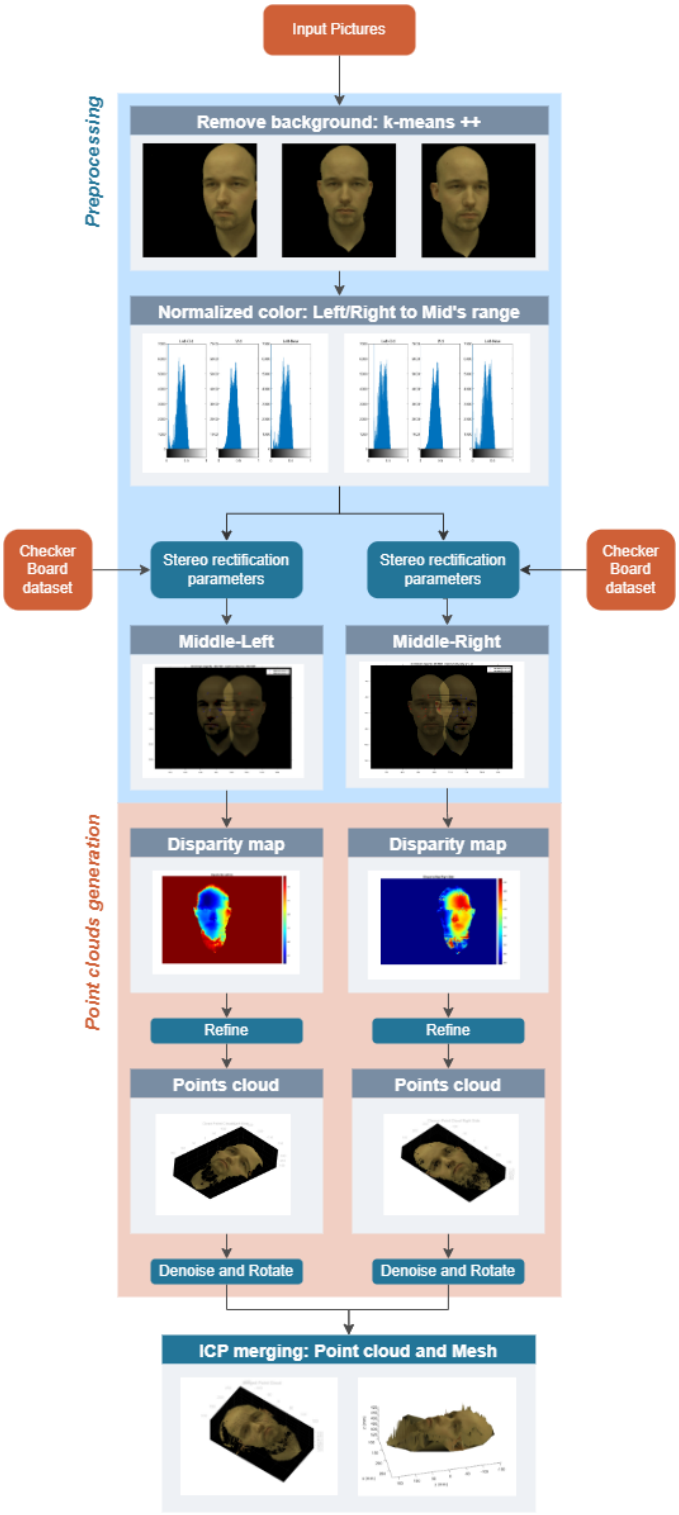

Fig. 2: The overall scheme of 3D face reconstruction method used in this project

Algorithm 1 k-means++ clustering face segmentation

1: Select an observation uniformly at random from the data set, X. The chosen observation is the first centroid, and is denoted c1.
2: Compute distances from each observation to c1. Denote the distance between cj and the observation m as $d(x_m, c_j)$
3: Select the next centroid, c2 at random from X with probability $\frac{d^2(x_m,c_1)}{\sum_{j=1}^{n} d^2(x_j,c_1)}$
4: **while** not converge **do**
5: **for** every i, set **do**
6: $c^{(i)} = argmin_j ||h^{(i)} - \mu_j||^2$
7: **end for**
8: **for** each j, set **do**
9: $\frac{d^2(x_m,c_p)}{\sum_{\{h;x_h \in C_p\}} d^2(x_h,c_p)}$
10: **end for**
11: **end while**

Algorithm 2 Basic rescursive flood-fill algorithm

1: **procedure** FLOOD-FILL(node)
2: **if** node is inside area to fill **then**
3: fill node
4: **for** all cardinal directions **do**
5: perform FLOOD-FILL on node one step into cardinal direction of current node
6: **end for**
7: **end if**
8: return
9: **end procedure**

K-means clustering is usually performed with the Lloyd algorithm, but we choose for a faster alternative implementation: k-means++[2], a clustering algorithm (see Algorithm 1) that chooses centroids randomly with probability:

$$\frac{d^2\left(x_m, c_p\right)}{\sum_{\{h;x_h \in C_p\}} d^2\left(x_h, c_p\right)} \tag{3}$$

where $d(x_h, c_p)$ is the distance between centroid and observation. After the division into 3 clusters, we manually select which sections of the image are background and shirt. Once the the shirt and background have been removed, the eyes and chin edges, which is similar and classified in the same cluster as the shirt, will be added back again to the pictures. The hole filling will be carried out with the flood-fill algorithm (see Algorithm 2 for a basic recursive implementation). This process will be repeated for each of the three images.

2) Global colour normalization: Stereo matching assumes the Constant Brightness Assumption to support key-point detection. This assumption states that corresponding points from two images have the same level of brightness. This assumption, however, might not always hold true considering different angels and lighting conditions from separate cameras. Therefore, it is necessary to normalize the brightness of all our images before stereo matching process. We will achieve

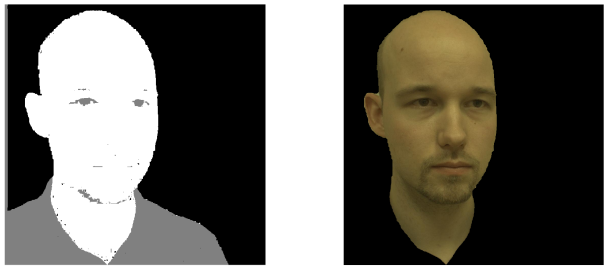

Fig. 3: Left - The left-camera picture is divided into 3 clusters depending on the color hue using k-means ++ method. Right - The result picture after background removal.

Fig. 4: Left - The right-camera picture before color normalized with the middle-camera picture. Right - The right-camera picture after color normalized with the middle-camera picture (become slightly darker). Both pictures are converted to black and white for easy visualization.

this global colour homogeneity by normalizing every channel of every picture according to its means and variances. These values are then scaled back according to a reference range in order for better visualization. The normalization formula for a channel x (R,G or B) in a specific picture is given as:

$$x = min(x_{ref}) + \frac{\mu(x)}{\sigma(x)} * (max(x_{ref}) - min(x_{ref})) \tag{4}$$

With this normalization, we will transform the 2-D image, returning an output image whose histogram approximately matches the histogram of the reference image. Each color channel of this image will be matched independently to the corresponding color channel of references image. The reference channels will be taken from the middle image. Thus, both left-camera and right-camera pictures are matched and scaled with the histogram of this picture.

3) Stereo rectification: Depth extraction is much simpler with aligned cameras; through stereo rectification, the images need to be rotated through projective transforms such that they are as if they had been taken by precisely aligned cameras.

For a fully aligned stereo set-up with a two cameras, the following is required:

- Rotate the cameras such that the orientation between cameras is 0: ${}^1\mathbf{R}_2 = \mathbf{I}$.

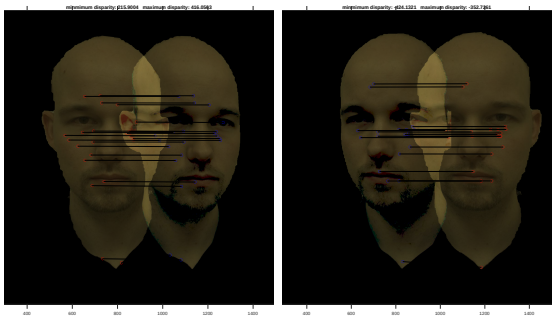

Fig. 5: Left - The left-camera picture is rectified according to the middle-camera picture. Right - The right-camera picture is rectified according to the middle-camera picture.

- Rotate the cameras such that the row directions coincide with the direction of the baseline.
- Change the two calibration matrices such that they are equal: $\mathbf{K}_1 = \mathbf{K}_2$.

For that, the cameras need to be calibrated; the relative position of the two cameras to each other must be exactly known. The intrinisc and extrinsic camera parameters are extracted with the help of checkerboard images (calibration images). Calibration is done using the method from Zhang et al. [13]. We transform the images such that both the left and right camera perspective are transformed to confirm to the perspective of the middle camera and we account for any non-linear lens deformation. We use visual inspection to ensure correct transformation as shown in Figure 5. A group of matching points will be detected by Speeded-Up Robust Features (SURF) method, and their epipolar lines will be sketch between the two pictures. We ensure that epipolar lines map to the same features, are parallel and horizontal, and their angles to the x axis will be considered as a criteria for the accuracy of our rectification.

4) Stereo matching: Given the fully aligned camera setup, we can calculate the disparities through dense stereo matching. Stereo matching describes the process of finding the corresponding point $p_1 = [nm]^T$ of image 1 in image 2 $p_2 = [uv]^T$. The epipolar constraint transforms this 2D search problem into a 1D search problem by utilizing the fact that p_2 must be located on the epipolar line of p_1. Since we rectified our images, the epipolar lines are horizontal and therefore the two corresponding points are on the same horizontal row. Let (n, m) and $(u, v) = (u, m)$ be corresponding points, then the disparity is calculated as

$$D = n - u \tag{5}$$

The disparities are found using the semi-global matching method (SGM) [5]. Dynamic programming approaches usually process the image row by row, missing the fact that disparities do not only have context in the direction of the row, but rather it has context in all directions. The semi-global matching method processes the disparities by considering it along multiple

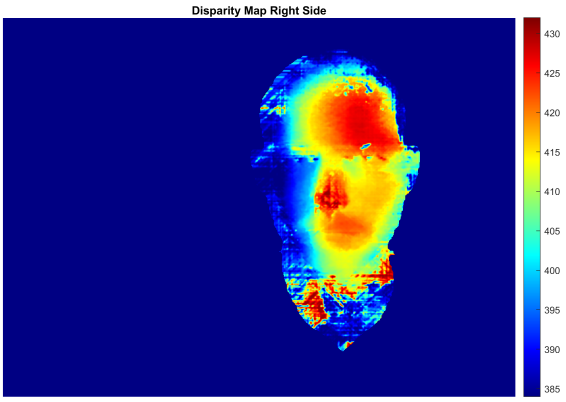

Fig. 6: Disparity Map using SGM

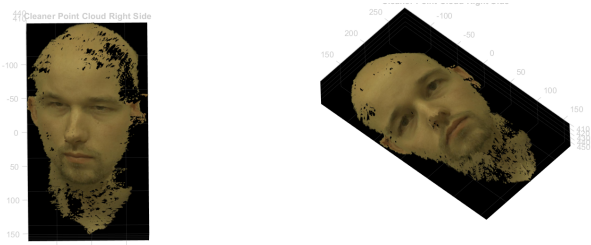

Fig. 7: Point cloud reconstructed from right-to-middle camera's disparity map

paths from different directions, choosing the disparity with the lowest cost. More specifically it tries to minimize

$$G[n, m] + \gamma \sum_{(i,j)\in N(n,m)} T_{i,j}[D(n, m), D(i, j)] \tag{6}$$

in which the first term $G[n, m]$ is the cost associated with creating a consistent solution with the observed image, and the second term ensuring consistency and smoothness within the neighbourhood of this pixel defined as $N(n, m)$. The disparity map associated with SGM yields a smoother disparity map compared to other dynamic programming approaches.

The disparity map is further refined by smoothened with a Gaussian filter and interpolating missing points/ holes with a Gaussian mask. Every unreliable points, such as "hidden points" with value of Nan or background area (corresponding to middle-camera image), will be also removed from the disparity map. Figure 6 showcases the disparity map after these refining steps.

5) 3D Point reconstruction: Finally, from a disparity map, a 3D reconstruction of the images can be made. The first step is to construct a point cloud from the disparity map. This point cloud is fist denoised to remove existing outliers using the average of the 4 nearest neighbours. With two disparity maps (left to middle and right to middle camera), we can

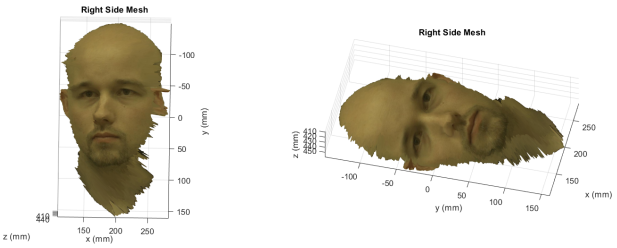

Fig. 8: 3D Surface mesh constructed from right-to-middle camera's point cloud

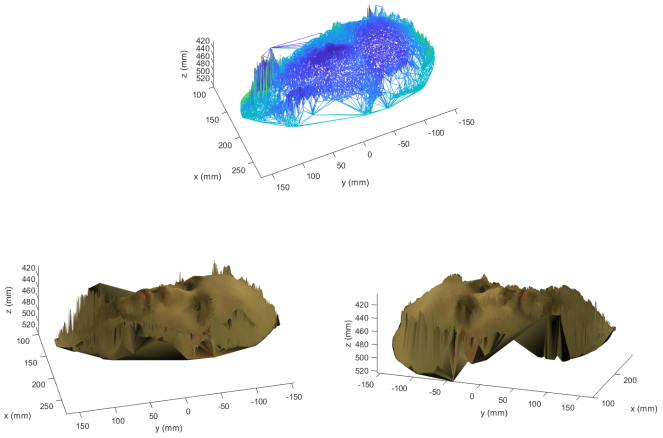

Fig. 9: Final 3D Surface mesh constructed from the two point clouds. Both side of the subject's face can be clearly seen on the graph.

reconstruct two points clouds that contain information from the two sides of the subject's face. One example of these point clouds reconstruction can be found in figure 7. We also created Delaunay meshes from these point clouds that are described by a set of adjacent triangles that together form the mesh. One such triangle is defined by 3 corners with their 3D position also defined as vertices. Given a list of these vertices and a connectivity list, it is possible to create a surface mesh. The surface mesh for the point cloud is also given in figure 7.

From these two refined point clouds from left-middle and right-middle camera, we will merge them together using the Iterative Closest Point (ICP) algorithm. To increase the efficiency of this matching algorithm and reduce calculation cost, we down-sample 1% of the point clouds before merging. Furthermore, for ICP, we also initiate a suitable rotation angle between these two clouds, deriving from rotation calculation we retrieved from the stereo rectification. The result for this merging is presented in mesh structure as shown in figure 9. It should be noted that the resolution of this mesh can be adjusted for a clear visualization and understand the robust structure of the result.

6) Performance evaluation: For the performance evaluation of 3D face reconstructions, no gold standard measure exists. But because we have two separate 3D point clouds from 3 stereo images, it is possible for us to compare them against each other. This would yield a measure that considers the performance based on how many points both point clouds have in common and consider them as correctly identified points. More specifically, the evaluation protocol is as follows:

- Reconstruct 3D point cloud from middle-right image pair
- Reconstruct 3D point cloud from middle-left image pair
- Align both 3D point clouds with ICP algorithm
- Calculate the root mean squared error (RMSE)

The RMSE is calculated between the aligned point clouds. More specifically, it is defined as:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(p_i^1 - p_i^2)^2} \tag{7}$$

where p_i^1 and p_i^2 are corresponding points in their respective pointcloud and n is the total number of points.

Another parameter that we can use to evaluate the quality of the mesh is using the smallest angles in every cell of the mesh. If there is none-smoothened surface in the mesh, there is more propability of the mesh creating a thin sliver triangle (2 sharp acute angles). This is also motivation for the Delaunay triangulation method used in generating the mesh, to avoid any bad quality cells creation. The criteria of a good mesh, thus will be decided basing on how many cells in this mesh is sliver triangles, or if their acute angles is less than a certain threshold. The angles of all the triangles in the mesh can be calculated using the vertices locations. The formula can be written as:

$$accuracy = 1 - \frac{N(faces\ with\ angle\ <\ threshold)}{N(faces)} \tag{8}$$

For our study, we choose triangles with angles smaller than $\pi/6$ non-satisfied (perfect triangles have all the angles equal to $\pi/3$).

III. Results

For the execution of this project, the 3 stereo images of subject 4 were used. Additionally, subject 4 has a neutral facial expression in all 3 stereo images. The stereo images were recorded from the an perspective on the left side of the face, the middle part of the face and the right side of the face. We followed the steps described in the method section and came to a final result. The final results of the 3D face reconstruction can be observed in figure 9. Figure 9 represents the application of the surface mesh on the merged point clouds of the two stereo pairs middle-right and middle-left. Executing our evaluation protocol by constructing both point clouds, aligning them with ICP and calculating the RMSE, we receive a performance value of

$$RMSE = 11.94 \tag{9}$$

The number of satisfied triangles over the number of all the cells were estimated as:

$$accuracy = 54.88\% \tag{10}$$

This reflects that more than half our grids is smooth and satisfied for the face reconstruction, while still 46.22 % of the mesh is not qualified.

IV. Discussion

This section judges the results. The background removal is generally well done, however the edges along the face are not very smooth. This is because there is not much texture at these edges, making it very difficult to accurately process them. This makes it very difficult to achieve accurate disparity measurements. As a result the depth estimation along the edges of the faces is not very accurate. Concurrently, the depth at the center of the face is estimated much better, possibly because of the better texture in the center of the face. As there is no gold standard for evaluating the quality or performance of our model, we came up with own evaluation frameworks. Judging the RMSE value of 11.94 is rather difficult without a comparison, but intuitively it gives us the average distance of corresponding points in the point cloud. We have millimeter units and therefore one could translate this measure as the average distance between corresponding points in the mesh. An average distance of 12 millimeter can be considered as an acceptable performance for applications that do not require high precision and given that this error rate is inside the allowed tolerance. On the other hand, this accuracy might not be acceptable for high precision application for example in medical analysis. The second evaluation metric using minimal angles of the mesh's cells, overall, correctly reflect our intuitive observation. Half of the mesh is considered rough and accurate but the other half, especially on our focus side of the face, is rather smooth. However, similar to RMSE value, further comparison with other data set will be more preferable.

Future work might use extended texture enhancement methods to improve the edge processing. Also the incorporation of information about the facial features could yield higher accuracy related to the background removal and the matching corresponding points in the stereo images. Furthermore, the disparity estimation could be improved with face feature detection with methods such as by Kazemi et al. [6]. Finally, a more sophisticated method to detect outliers inside the point cloud or mesh could reduce the inconsistencies that can be observed around the edge but also imply less texture and points that define the point cloud.

V. Conclusion

Concluding, we have reconstructed a 3D facial surface mesh with the help of three input Stereo images. From these, we have built two point clouds after applying preprocessing, stereo rectification and stereo matching to receive the depth information about this scene encoded in the stereo images. These point clouds are merged and a surface mesh is created. Lastly, we evaluated the reconstruction with the help of the RMSE and considered our surface mesh to still have room for improvement especially regarding detecting outliers at the edges and a more accurate estimation of points along the edge of the face. A reconstructed 3D model can be used in applications related to facial recognition or medical analysis such as grading the degree of facial paralysis.

References

[1] "Informatics in control, automation and robotics: selected papers from the International Conference on Informatics in Control, Automation and Robotics 2006". In: Lecture notes in electrical engineering 15 (2008). Ed. by Juan Andrade-Cetto. Meeting Name: International Conference on Informatics in Control, Automation and Robotics OCLC: ocn243819193.

[2] David Arthur and Sergei Vassilvitskii. "k-means++: The Advantages of Careful Seeding". en. In: (), p. 11.

[3] A. Dipanda et al. "3-D shape reconstruction in an active stereo vision system using genetic algorithms". en. In: *Pattern Recognition*. Kernel and Subspace Methods for Computer Vision 36.9 (Sept. 2003), pp. 2143–2159. ISSN: 0031-3203. DOI: 10.1016/S0031-3203(03)00049-9. URL: https://www.sciencedirect.com/science/article/pii/S0031320303000499 (visited on 04/15/2022).

[4] "Frontiers — What is binocular disparity? — Psychology". In: (). URL: https://www.frontiersin.org/articles/10.3389/fpsyg.2014.00870/full (visited on 04/15/2022).

[5] Heiko Hirschmuller. "Accurate and efficient stereo processing by semi-global matching and mutual information". In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* 2 (2005), 807–814 vol. 2.

[6] Vahid Kazemi and Josephine Sullivan. "One millisecond face alignment with an ensemble of regression trees". In: (June 2014), pp. 1867–1874. DOI: 10.1109/CVPR.2014.241. URL: https://ieeexplore.ieee.org/document/6909637 (visited on 04/16/2022).

[7] Y. Lu et al. "A Survey of Motion-Parallax-Based 3-D Reconstruction Algorithms". In: *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)* 34.4 (Nov. 2004), pp. 532–548. ISSN: 1094-6977. DOI: 10.1109/TSMCC.2004.829300. URL: http://ieeexplore.ieee.org/document/1347305/ (visited on 04/15/2022).

[8] Jonathan A. Marshall et al. "Occlusion edge blur: a cue to relative visual depth". In: *Journal of the Optical Society of America A* 13.4 (Apr. 1, 1996), p. 681. ISSN: 1084-7529, 1520-8532. DOI: 10.1364/JOSAA.13.000681. URL: https://opg.optica.org/abstract.cfm?URI=josaa-13-4-681 (visited on 04/15/2022).

[9] Ashutosh Saxena, Min Sun, and Andrew Y. Ng. "3-D Reconstruction from Sparse Views using Monocular Vision". In: (Oct. 2007). ISSN: 2380-7504, pp. 1–8. DOI: 10.1109/ICCV.2007.4409219.

[10] "Stereo Vision - an overview — ScienceDirect Topics". In: (). URL: https://www.sciencedirect.com/topics/computer-science/stereo-vision (visited on 04/15/2022).

[11] Michael W. Tao et al. “Depth from shading, defocus, and correspondence using light-field angular coherence”. In: (June 2015), pp. 1940–1948. DOI: 10.1109/CVPR.2015.7298804. URL: http://ieeexplore.ieee.org/document/7298804/ (visited on 04/16/2022).

[12] Qingqing Wei. “Converting 2D to 3D: A Survey”. In: (2005).

[13] Z. Zhang. “A flexible new technique for camera calibration”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.11 (Nov. 2000). Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 1330–1334. ISSN: 1939-3539. DOI: 10.1109/34.888718.