3D face reconstruction

Tongli Zhu, *t.zhu-2@student.utwente.nl,s3159396,Robotics*
Qingyi Xu, *q.xu-1@student.utwente.nl,s3238113,Robotics*
Oceane Tai, *o.c.p.h.thai@student.utwente.nl,s3316866,Computer Science*

***Abstract*—In this study, we introduce a methodology for 3D facial reconstruction through image preprocessing, single mesh generation, and dual-mesh fusion, leveraging binocular vision principles. We encounter and address challenges such as background noise and mesh quality, which inform our recommendations for advanced image processing and point cloud refinement techniques.**

***Keywords*—stereo vision, 3D face reconstruction, disparity mapping, 3D point cloud**

I. Introduction

WITH the integration of computer algorithms, modern medical imaging has advanced, enabling clearer visuals and improved diagnoses. Yet, transitioning from multiple 2D images to an accurate 3D model remains a challenge, aiming to create a true-to-life 3D representation of the subject.

Capturing the 3D geometry of an object is crucial in various fields like medical imaging, face recognition, and computer-aided design. The common method is through a 3D surface representation in a computer model, utilizing a 3D point cloud. However, this method lacks the ability to represent the connectedness among points.

This task requires developing a method for generating 3D surface meshes of subjects' faces, promising applications in face recognition and medical fields, particularly in evaluating facial paralysis due to neural disorders. The subtasks include camera calibration, compensation for non-linear lens deformation, stereo rectification, global color normalization, stereo matching for disparity maps, background detection, identification of unreliable disparities, merging of 3D surface meshes using the Iterated Closest Points (ICP) algorithm, and mesh quality assessment.

This paper demonstrates the 3D reconstruction process using the passive, multi-image stereo vision technique to create a 3D facial mesh. Section II details the materials and procedures; Section III presents the results; Section IV interprets the outcomes and discusses limitations; Section V concludes by recapping the overall execution and emphasizing the significance of this work.

II. Methods and Materials

A. Materials

The primary dataset used in the project is detailed in Table I.

TABLE I: Material list

Items	Contents
Software	Matlab R2023b(Stereo Camera Calibrator Application)
Subjects	1 (each with three sets of images captured at various angles)
Calibration Data	2 sets of checker-board images (20 images/set from all 3 cameras)
Calibration Used	1st calibration folder

B. Methods

The reconstruction process can be categorized into three distinct stages: Image Preprocessing, Single Mesh Generation, and Dual-Mesh Fusion. An illustrative representation of this methodology is depicted in the Fig. 1. The crux of the process lies in the generation of the single mesh.

Central to our approach is the understanding that two photographs of the same object, captured from differing perspectives, contain a plethora of corresponding points. Post image preprocessing, for any set of corresponding points (u, v) from both photographs, binocular vision principles ascertain that after rectification, these points align on an identical horizontal axis. Leveraging the properties of similar triangles, the disparity between two sets of corresponding points can be determined. Subsequently, this facilitates the extraction of depth information, essentially quantifying the Z of the point within the world coordinate system.

Utilizing both intrinsic and extrinsic camera parameters, the coordinates (u, v) can be transposed to the camera coordinate system, and thereafter, mapped to spatial coordinates (X, Y, Z) within the global frame of reference. Upon processing all corresponding points via the aforementioned steps, a point cloud is synthesized. The culmination of this procedure involves the amalgamation of two point clouds, thereby yielding a three-dimensional facial representation.

a.Camera Calibration: Camera calibration is a critical procedure performed to determine the intrinsic and extrinsic parameters of imaging devices. In this study, the well-established Zhang's method is employed for calibration, leveraging checkerboard patterns to facilitate precise feature extraction [1].

The calibration parameters can be broadly categorized into two primary groups:

1) **Single Camera Parameters:** These pertain to both the intrinsic and extrinsic parameters of an individual camera. Notably, the intrinsic parameters play a pivotal role in rectifying nonlinear lens distortions which might skew the captured imagery.
2) **Stereo Calibration Parameters:** The emphasis here lies on discerning the rotation and translation matrices

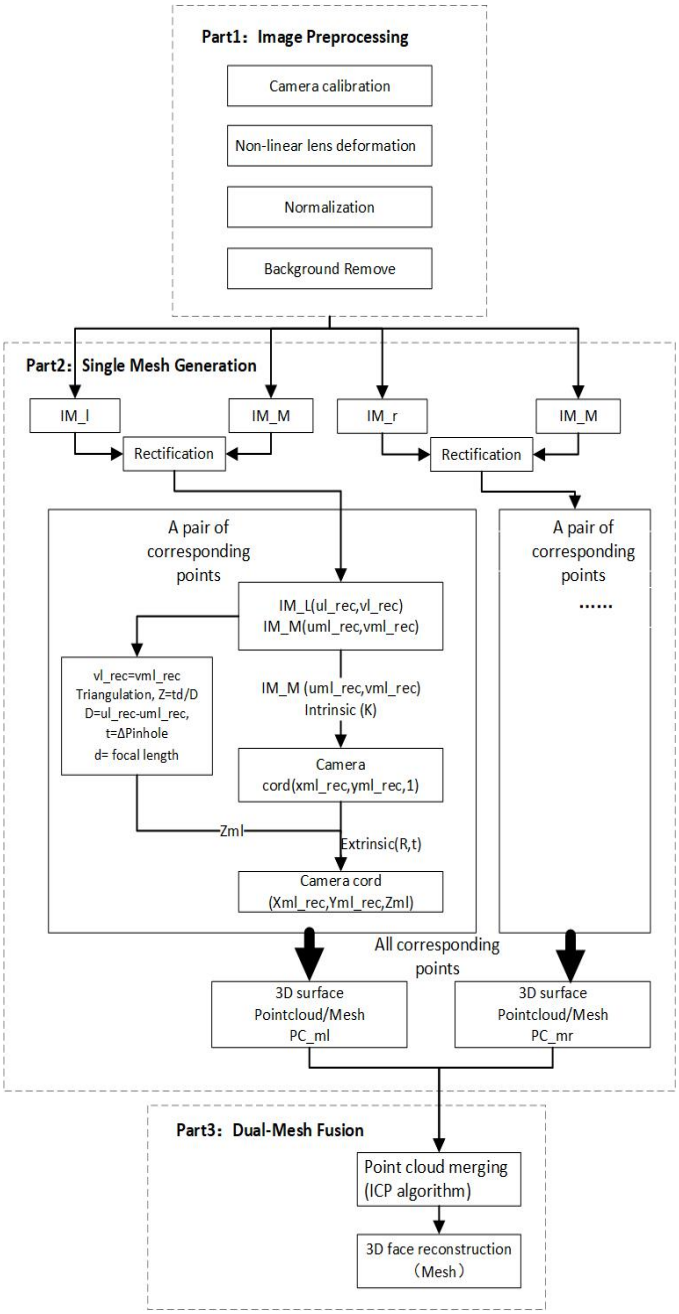

Fig. 1: Subtask flow chart

linking the camera pair. This is imperative for determining the fundamental matrix, F. Given a point x in the left camera and its corresponding point x' in the right camera, the fundamental matrix conforms to the relationship:

$$x'^T F x = 0 \tag{1}$$

The derived relationship facilitated by the fundamental matrix offers robust constraints, proving instrumental in binocular point correspondence during subsequent stereo matching.

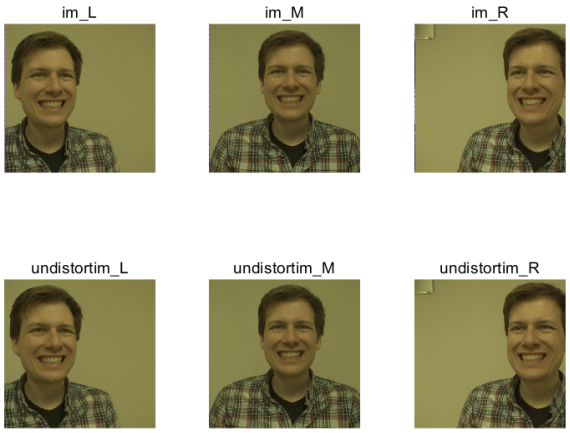

Fig. 2: Comparison chart of compensating nonlinear lens distortion

b. Compensating for non-linear lens deformation : Due to the inherent physical attributes of a lens, the process of projection may engender distortion. The mathematical delineation of distortion can be articulated through the subsequent equations:

$$x_d = x(1 + k_1 r^2 + k_2 r^4) + 2p_1 xy + p_2(r^2 + 2x^2) \tag{2}$$

$$y_d = y(1 + k_1 r^2 + k_2 r^4) + 2p_2 xy + p_1(r^2 + 2y^2) \tag{3}$$

where x_d, y_d denote the positions of distorted points in the image, x, y represent the positions of undistorted points, k_1, k_2 are the radial distortion coefficients, p_1, p_2 are the tangential distortion coefficients, and r is the radius from the center of the image to a point x, y.

To rectify this distortion, an inverse distortion model is employed to map the distorted pixel positions back to their original, undistorted positions. In instances where distortion correction results in pixel positions that do not correspond to integer coordinates in the image, interpolation methods are utilized for compensation.

By observing the compensated image Fig. 2, we found that there is not much difference from the original image. This proves that the original image was shot with a high-quality lens, and the distortion introduced by the lens is very small.

c. Normalization: A normalization procedure is employed to align the statistical characteristics across images $\text{Im\_left}, \text{Im\_middle}, \text{Im\_right}$. Initially, the mean ($\text{mean}_i$) and standard deviation (std_i) of each image are computed as:

$$\text{mean}_i = \frac{1}{N} \sum \text{Im}_i \tag{4}$$

$$\text{std}_i = \sqrt{\frac{1}{N} \sum (\text{Im}_i - \text{mean}_i)^2} \tag{5}$$

where N is the total number of pixels, and i indexes the images. The image Im_middle serves as a reference with its mean and standard deviation denoted as referenceMean and referenceStd respectively. The normalized images J_1, J_2, J_3 are then calculated using the formula:

$$J_i = \left(\frac{\text{referenceStd}}{\text{std}_i} \right) \cdot (\text{Im}_i - \text{mean}_i) + \text{referenceMean} \quad (6)$$

This adjustment aligns the brightness and contrast of the images to match the reference, facilitating subsequent image processing tasks.

d. Background Extraction: Following the normalization phase on the images Im_left, Im_middle, Im_right, the algorithm transitions to the background extraction process, which involves the application of methods from Canny's edge detection [2] and the principles of image morphology, specifically erosion and dilation. This process involves a loop over the three images, where for each image, the subsequent steps are performed:

```
Algorithm: Background Extraction
Input: Im_left, Im_middle, Im_right
     Canny edge detection parameters
     Structuring element parameters
Output: Background removed images

1. for each image img in {Im_left,
   Im_middle, Im_right} do
2. Apply Canny edge detection on grayscale
    of img to obtain edge map edges
3. Perform morphological operations on
   edges using structuring element
   strel_params to obtain refined edge
   map edges_refined
4. Create binary mask mask from
   edges_refined
5. Apply mask to each color channel of img
    to obtain background image BG_img
6. Store BG_img in the list of background
   images
7. end for
8. return BG1, BG2, BG3
```

This structured approach enables the efficient extraction of background from each image, which sets a robust foundation for the subsequent stages of analysis and processing. The image after the operation can refer to Fig. 3

e. Image Rectification: Following the normalization phase, the next critical step is the rectification of the stereo images. Rectification is a process that geometrically aligns two stereo images such that corresponding points occur on the same pixel rows in both images. This alignment is crucial for simplifying the subsequent stereo correspondence problem.Following the stereo calibration procedure of the cameras, the foundational matrix and intrinsic calibration matrices K_1 and K_2 were obtained. Utilizing these parameters, a rotation transformation was applied, ensuring that the transformed calibration matrix of the second camera (K_2') is equivalent to

Fig. 3: After removing the background and background mask

Fig. 4: Rectification and inspection

the calibration matrix of the first camera (K_1), i.e., $K_2' = K_1$. Subsequently, a pairwise rectification was executed on the trio of images, yielding two sets of rectified image pairs: left-middle and right-middle. This rectification procedure assured the horizontal alignment of the epipolar lines across both images within each pair, thus providing a foundational basis for the subsequent computation of disparity maps, which could be seen on the Fig. 4.

f. Disparity Estimation using Semi-Global Matching (SGM): Post rectification, the next pivotal step entails the estimation of disparity between the rectified image pairs utilizing the Semi-Global Matching (SGM) algorithm. The essence of SGM lies in its capacity to balance between accuracy and computational efficiency which is paramount for real-time applications. It operates on the principle of minimizing a defined energy function that encapsulates both data and smoothness terms. The energy function is expressed as follows:

$$E(D) = \sum_p \left(C(p, D_p) + \sum_{q \in N(p)} P1 \cdot |D_p - D_q| + P2 \cdot \min(|D_p - D_q| - 1, 0) \right) \quad (7)$$

where:

- D represents the disparity map.
- p and q denote pixel positions, with q belonging to the neighborhood $N(p)$ of p.

- $C(p, D_p)$ is the matching cost at pixel p for disparity D_p.
- $P1$ and $P2$ are penalty terms that control the disparity map's smoothness [3].

Using the SGM algorithm to calculate the disparity map between the grayscale versions of the corrected image. The algorithm uses a specified disparity range to search for matching blocks.The calculated disparity map can be further refined by applying a mask in the background extraction step to retain only the disparity information in the face.The uniqueness threshold parameter in SGM helps to flag the estimated disparity value of a pixel as unreliable under certain conditions, thereby enhancing the robustness of the disparity map for accurate 3D reconstruction and analysis.

g. Disparity Map Refinement: Following the normalization and rectification stages, the disparity map obtained needs to be refined to enhance its reliability for further processing. The refinement process primarily aims at eliminating unreliable disparity values and smoothing the disparity map while preserving the disparities at edges.

Initially, unreliable disparity values are identified and set to zero. These unreliable values often arise due to occlusions or insufficient texture in the corresponding regions. A common approach to identify these unreliable values is to check for disparity values that are equal to zero or are not-a-number (NaN), which usually result from failed matching attempts.

Subsequently, a smoothing operation is performed to reduce noise in the disparity map while preserving the disparity values at edges. A common technique employed is the application of a Gaussian filter within a local window around each pixel. However, direct application of a Gaussian filter will smooth across edges, which is undesirable. Hence, an edge-preserving smoothing technique is adopted.

The essence of the smoothing operation can be captured by the following expression:

$$D_{\text{refined}}(i,j) = \frac{\sum_{m,n} D(i+m, j+n) \cdot W(m,n)}{\sum_{m,n} W(m,n)}, \tag{8}$$

where D represents the original disparity map, D_{refined} is the refined disparity map, (i, j) are the coordinates of the current pixel, and (m, n) represent the coordinates of the pixels within the local window around the current pixel. The weight $W(m, n)$ is computed based on the Gaussian function along with a measure of similarity between the current pixel and the neighboring pixels to ensure edge preservation.

By observing image Fig. 5, we can see that the optimized disparity map has the effect of filling in some missing holes. This refined disparity map is now ready for further processing and analysis, providing a more reliable representation of the scene's depth information.

h. 3D Reconstruction: Upon obtaining a reliable disparity map, the subsequent step is the reconstruction of the 3D scene. The essence of 3D reconstruction lies in mapping the disparity values to corresponding 3D coordinates in a Cartesian coordinate system. This transition from 2D to 3D is facilitated by the camera projection matrices obtained during the stereo calibration process.

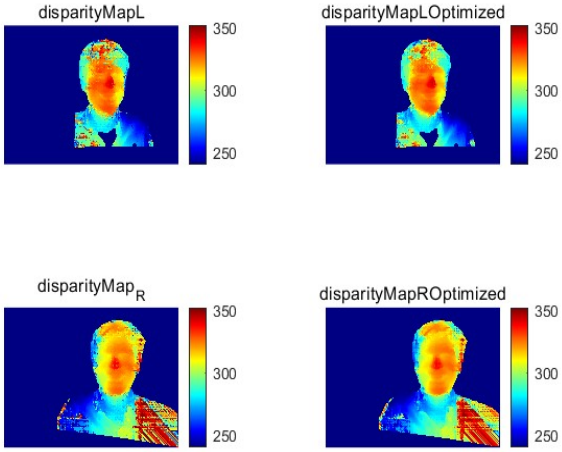

Fig. 5: Disparity map contrast

The 3D coordinates (X, Y, Z) of a point in the scene can be derived from its disparity d and its coordinates (x, y) in the image, using the following relations:

$$X = \frac{(x - c_x) \cdot Z}{f}, \tag{9}$$

$$Y = \frac{(y - c_y) \cdot Z}{f}, \tag{10}$$

$$Z = \frac{f \cdot b}{d}, \tag{11}$$

where f is the focal length, b is the baseline (the distance between the cameras), and c_x, c_y are the coordinates of the principal point, typically the image center.

An array of 3D points is generated by applying these equations to every pixel in the disparity map. Each 3D point is also associated with a color value from the rectified image, forming a colored point cloud representation of the 3D scene, as shown in Fig. 6.

This point cloud encapsulates the geometry and appearance of the scene, and is amenable for various post-processing steps such as mesh generation, surface reconstruction, and visualization. The colored point cloud is visualized to provide an intuitive understanding of the scene's 3D structure, thereby marking the culmination of the stereo vision processing pipeline.

i. Point Cloud Registration and Merging: The process of aligning and merging distinct point clouds is paramount for creating a comprehensive 3D representation of the scene. The Iterative Closest Point (ICP) algorithm is employed for this registration task, which iteratively minimizes the distance between corresponding points from the two point clouds.

The ICP algorithm can be described in a generalized form as follows:

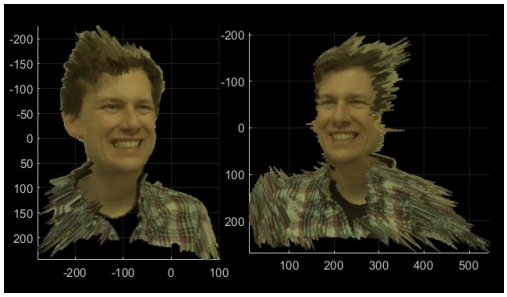

Fig. 6: Point cloud

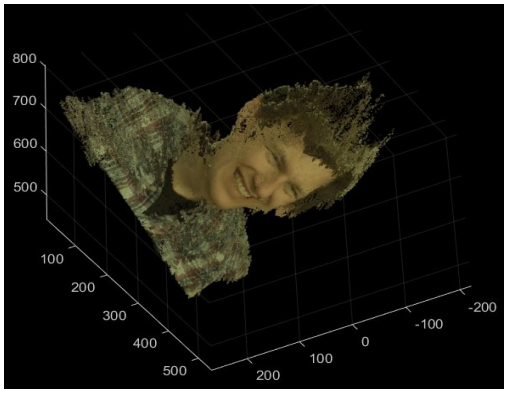

Fig. 7: Point Cloud Merging(ICP)

1) Initial guess of the transformation is made.
2) For each point in one point cloud, the nearest point in the other point cloud is found.
3) A rigid transformation (rotation and translation) that minimizes the mean squared error between corresponding points is computed.
4) The transformation is applied to one of the point clouds.
5) Steps 2 to 4 are repeated until the change in mean squared error falls below a certain threshold, or a maximum number of iterations is reached.

Mathematically, the objective of the ICP algorithm is to find the transformation T that minimizes the following cost function:

$$E(T) = \sum_{i=1}^{N} ||p_i - T(q_i)||^2 \tag{12}$$

where p_i and q_i are corresponding points from the two point clouds, N is the total number of point pairs, and T is the transformation matrix [4].

After the registration, the aligned point clouds are merged into a unified point cloud using a straightforward merging procedure, which can be seen in Fig. 7. This merging is performed within a specified distance threshold, ensuring that overlapping points from the different point clouds are combined into a single point, thereby preserving the structural and color integrity of the scene.

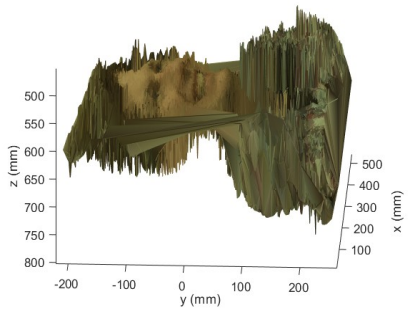

Fig. 8: 3D Mesh

j. Mesh Generation from 3D Points: The next stage involves the conversion of the 3D point cloud into a 3D mesh, which facilitates a structured representation of the spatial geometry. This transition employs the Delaunay Triangulation method, renowned for its ability to optimize the angular characteristics of the resultant mesh triangles, averting excessively sharp or flat corners [5].

The essence of Delaunay Triangulation lies in forming a mesh whereby the circumcircle of any triangle does not encompass any other points in the set, mathematically articulated as:

$$\forall\, \Delta abc \in \text{Mesh},\ \nexists\, p \in P : p \in \text{Circumcircle}(\Delta abc) \tag{13}$$

where P is the set of 3D points, and Circumcircle(Δabc) denotes the circumcircle of triangle Δabc [5].

The outcome of this triangulation is a connectivity relation, which when amalgamated with a predetermined resolution, operates as a sieve to refine the original data. The resolution parameter acts as a sieve, downsampling the data by selecting points that coincide with a grid of a specified resolution. This process alleviates computational load while retaining essential geometric information.

In the culmination of this procedure, the purified set of point positions alongside their corresponding colors is obtained. Moreover, the connectivity relations derived from the Delaunay Triangulation provide the necessary blueprint for assembling these points into a coherent 3D mesh. The result of the 3D mesh can be seen in Fig. 8.

In addition to Delaunay Triangulation, We also tried the Poisson reconstruction method. Poisson reconstruction infers a smooth surface from a point cloud by solving the Poisson equation. It works by treating each point in the point cloud as a sample with directions that indicate the normal vector of the surface. By solving a 3D Poisson equation, we can obtain a function whose gradient field is closest to the normal vector field of the point cloud. An isosurface can then be extracted from this function as the reconstructed surface [6]. The result can be seen in Fig. 9. The effect of Poisson reconstruction is usually smooth and can fill holes in the data

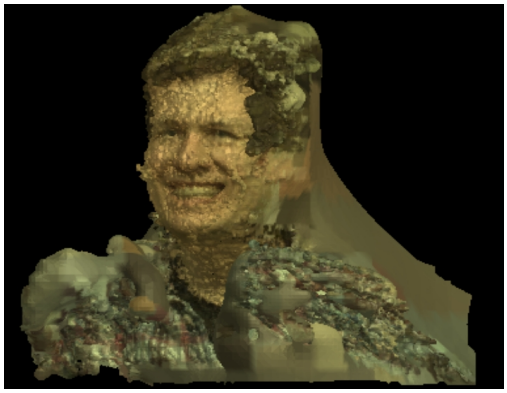

Fig. 9: 3D Mesh(Poisson reconstruction)

well. Nonetheless, the assessment of outcomes derived from this methodology poses considerable challenges. Consequently, despite initial experimentation with the Poisson approach, the decision was ultimately made to employ the Delaunay Triangulation method.

C. Performance evaluation

In the realm of mesh quality evaluation, various methodologies are prevalent, which could be seen on Table.II. These methods typically gauge different facets of the mesh to ensure its integrity, accuracy, and reliability for the desired application.

TABLE II: Sample table using tabularx

Column 1	Column 2
Minimum Angle	Evaluate the quality of the mesh by examining the smallest angles in every triangle. Triangles with acute angles below a certain threshold are considered of lower quality.
Area Ratio	Compare the maximum and minimum areas of triangles. Triangles with large discrepancies might indicate quality issues.
Edge Length Ratio	For each triangle, compare the ratio of its longest edge to its shortest. Large discrepancies might suggest potential problems.
Gaussian Curvature	Examine the Gaussian curvature for surface meshes. It indicates the amount of curvature at a specific point.
Neighboring Triangle Angle Ratio	Compare angles of a triangle to those of its neighboring triangles. Large differences can indicate potential quality issues.
Node Quality	Assess the quality of all triangles surrounding a node. If a node has many low-quality neighboring triangles, it might suggest a problem.

Among the diverse techniques available for mesh evaluation, our research underscores the *Angular Integrity* approach. The rationale behind this selection is the method's aptitude for accurately pinpointing regions of the mesh that fall short of desired quality metrics.

Central to this methodology is the scrutiny of the minute angles in each triangle constituting the mesh. Triangles exhibiting acute angles that are less than a pre-established benchmark are identified as having compromised quality. The mesh quality, based on this principle, can be mathematically represented as:

$$\textit{reliability} = 1 - \frac{N(\text{triangles with angle} < \text{set value})}{N(\text{total triangles})} \quad (14)$$

For the scope of our investigation, triangles with angles that are less than $\frac{\pi}{6}$ are labeled unsatisfactory.

III. Results

The obtained 3D-MESH diagram is shown in Fig. 8, and the calculated reliability is 53.79%. The overall effect of the face is presented well, but due to the introduction of 3D points other than the face, there are too many edges.

IV. Discussion

One of the primary limitations faced during the execution of our study was the choice of the Canny algorithm for background removal. While effective in certain scenarios, the relatively older Canny algorithm inadvertently introduced some extraneous points into the disparity map, including those from shoulders and clothes. These unintended points were subsequently passed on to the 3D point cloud, affecting the integrity of the mesh.We tried to experiment on another set of images where it was easier to remove the shirt background, and found that just as we had the same idea, the effect would be much better, see Fig. 10.

Another constraint was the thresholding rule applied to the 3-corner grid. By imposing a restriction wherein triangles with the largest side exceeding 2 units were removed, we indeed reduced the influence of edges and achieved a reliability increase to 59%. However, this also resulted in many missing triangles, particularly on the facial region of the mesh, thereby limiting the accuracy and aesthetic appeal of the model, which could be seen in Fig. 11.

For future endeavors, we recommend employing a more sophisticated background removal technique to enhance the quality and accuracy of the 3D point cloud. Additionally, incorporating a data-cleaning operation for the 3D points during the meshing process is advisable. With the integration of these advanced methodologies, we anticipate a significant improvement in the model's overall accuracy and aesthetic quality.

V. Conclusion

Our study advanced 3D facial reconstruction from 2D images, achieving a 53.79% reliability in mesh generation. The Angular Integrity evaluation underscored the need for precise preprocessing, as the Canny algorithm's limitations led to inaccuracies due to unwanted background features. Stringent thresholding further compromised facial details in the mesh.

Future improvements hinge on employing sophisticated background subtraction and incorporating a point cleaning step, which promise to refine the reconstruction process significantly. These advancements will not only enhance mesh quality but also have broader implications for applications in virtual reality and medical planning, warranting ongoing research and development.

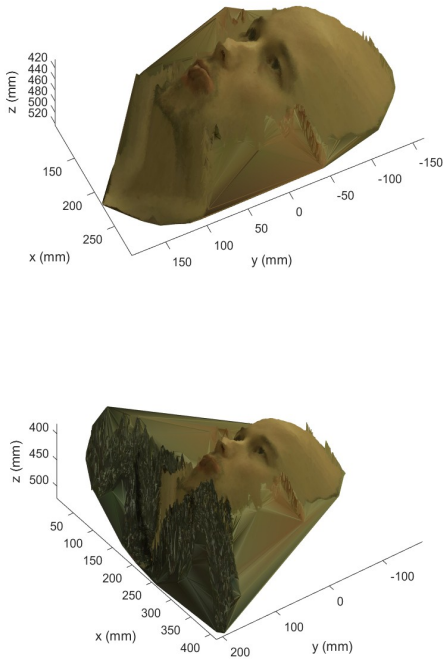

Fig. 10: Background influence

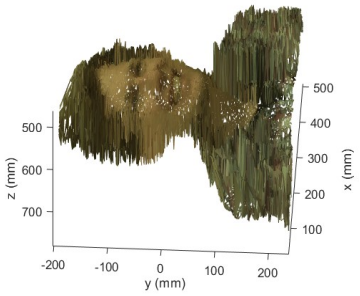

Fig. 11: 3D Mesh(threshold)

REFERENCES

[1] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[2] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[3] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328–341, 2008.

[4] P. J. Besl and N. D. McKay, "Method for registration of 3-D shapes," in *Sensor Fusion IV: Control Paradigms and Data Structures*, P. S. Schenker, Ed., vol. 1611, International Society for Optics and Photonics. SPIE, 1992, pp. 586 – 606. [Online]. Available: https://doi.org/10.1117/12.57955

[5] O. R. Musin, "Properties of the delaunay triangulation," in *Proceedings of the thirteenth annual symposium on Computational geometry*, 1997, pp. 424–426.

[6] M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson surface reconstruction," in *Proceedings of the fourth Eurographics symposium on Geometry processing*, vol. 7, 2006, p. 0.