3D Face Reconstruction from Stereo Images

3D Models in Computer Vision (Prof. Alain Tremeau)
Jean Monnet University

Taekyung Kim (GitHub)
Luel Abrha Gebre (GitHub)
Kidu Abrha Welegerima (GitHub)

June 5, 2025

Abstract :

This computer vision project aims at 3D face reconstruction from 2D images, with three different approaches by three students: Taekyung Kim, Luel Abrha Gebre and Kidu Abrha Welegerima.
Taekyung Kim works on a classical method based on epipolar geometry and an AI-based method using the Face Landmark Detection model. The data images are generated by Blender, and CLAHE, SIFT, RANSAC, triangulation, depth filtering, smoothing, and UV texture mapping techniques are used. Furthermore, a psychophysical experiment with 38 participants showed that smaller camera transformations improve classical results, but the AI-based method was indicated as the best. Additionally, an attempt was made to integrate classical and AI methods, but it failed due to some difficulties.
The second technique (Luel) presents a multi-view stereo (MVS) reconstruction pipeline that is based on combination of feature detection using SIFT (scale Invariant feature transform) and facial landmark for key point detection with classical geometric techniques to generate high-fidelity textured 3D face models. The proposed method first extracts 478 2D facial landmarks per image and also detects features using SIFT, combines them, then estimates relative camera poses via essential matrix decomposition (un known extrinsic camera parameters), and finally reconstructs a dense 3D mesh using Poisson surface reconstruction. Quantitative experimental results shows that my approach achieves sub-pixel re-projection accuracy while maintaining computational efficiency.
Kidu Abrha focused on 3D face reconstruction from stereo images under three calibration scenarios: fully calibrated, partially calibrated, and uncalibrated. Using synthetic multi-view data with controlled baselines and rotations, he implemented SIFT-based pipelines with RANSAC and triangulation. His results show that moderate baselines with small rotations (±5°, ±0.1 m) achieve optimal reconstruction, balancing depth accuracy, reprojection error, and point cloud completeness. While calibrated methods remain robust under wider baselines, uncalibrated setups degrade sharply with increasing viewpoint changes. His work highlights key trade-offs between geometric fidelity, depth precision, and calibration dependency in stereo-based 3D face modeling.

Index Terms: Computer Vision, 3D reconstruction, Epipolar Geometry.

1 Introduction

In computer vision, the 3D reconstruction task is known for the process of estimating the 3D structure of a scene or object from one or more 2D images and involves dealing with depth and spatial information from images. Also, in general, some data is lost during image capture by angle, thus, we apply the geometry in images, specifically the epipolar geometry that we have learned from the course, of the object in a 3D coordinate space.
This project focuses on the reconstruction of human faces. The reason for the choice of faces, rather than arbitrary objects, is motivated by their wide applicability for various things, including industry, education, and research. For example, the cosmetic industry, L'Oréal, can benefit from 3D facial models to test products under different lighting conditions for simulation. Similarly, historical reconstruction and educational tools can leverage 3D models of historical figures, generated from archival images, to enhance engagement and understanding Yoon, 2021.
We utilised Blender, which is a 3D modelling application, to generate facial images from multiple viewpoints: left, front and right.
We utilised Blender, which is a 3D modelling application, to generate facial images from multiple viewpoints: left, front and right.
Three different technical approaches are described in this report, each proposed by Taekyung Kim, Luel Abrha Gebre and Kidu Abrha Welegerima. For the evaluation and conclusion, assessing all results of the three methods at the end, not only the individual psychophysical experiment organised by Taekyung Kim, is aimed at with their conclusion by combining them if possible.

2 Background/Related Work

This section describes the theoretical background and related research for the project.

2.1 Approach 1 (Taekyung)

2.1.1 Epipolar Geometry Epipolar geometry relies on stereo vision with the projective geometry between two views of a 3D scene captured by separated cameras. With the positions and orientations of the cameras, the correspondence problem is reduced to epipolar lines rather than the entire image plane with stereo matching Hartley, 2003. The epipoles are the points of intersection of the line connecting the optical centres with each image plane, and this is mathematically encapsulated by matrices Longuet-Higgins, 1981.

2.1.2 Camera Calibration Camera calibration is one of the key elements for 3D reconstruction tasks. This consists of intrinsic parameters, which are focal length, principal point, skew, and rotation and translation information for a world coordinate system, as

the extrinsic matrix given by:

$$[\,\mathbf{R} \mid \mathbf{t}\,] \in \mathbb{R}^{3\times 4}. \tag{1}$$

The extrinsic parameters define the transformation between the coordinates of the world and images Zhang, 2002. In addition, intrinsic parameters describe the internal geometry of the camera and the matrix below:

$$\mathbf{K} = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \tag{2}$$

where f_x and f_y are the focal lengths in pixels along the x and y axes, respectively, c_x and c_y denote the coordinates of the principal point, and s represents the skew coefficient (usually zero for most modern cameras).

2.1.3 Feature Detection and Matching For feature detection and matching, several algorithms are used in general. Initially, CLAHE, Contrast Limited Adaptive Histogram Equalisation, Reza, 2004, is applied to the original image to improve contrast. This image processing limits the image noise while adapting histogram equalisation in small tiles and returns a more uniform contrast distribution of the image.
The SIFT, Scale-Invariant Feature Transform, algorithm Lowe, 2004 is one of the most commonly used methods that detects keypoints using a DoG, difference of Gaussians approach, and generates descriptors for matching. The detected features are reliably matched between images taken under different conditions.
To match keypoints between images, the Random Sample Consensus algorithm Fischler et al., 1981 is applied. It iteratively estimates the parameters of a geometric model, the fundamental matrix, by randomly sampling minimal subsets of matched points and identifying inliers. After that, it eliminates outlier correspondences of noise or mismatches.

2.1.4 Triangulation Triangulation is a technique in computer vision for recovering the 3D position $\mathbf{X} = (X, Y, Z, 1)^\top$, from its projections in different camera images in homogeneous coordinates. Each camera has a projection matrix that maps the 3D point to the image coordinates, $\mathbf{x} = (u, v, 1)^\top$.

$$\mathbf{x} \sim \mathbf{P}\mathbf{X}. \tag{3}$$

The projection matrix, size 3×4, relies on the intrinsic and extrinsic parameters of each camera, which are encapsulated in the matrix:

$$\mathbf{P} = \mathbf{K}\left[\begin{array}{c|c} \mathbf{R} & \mathbf{t} \end{array}\right] \tag{4}$$

where $\mathbf{K}$ is the intrinsic calibration matrix, and $(\mathbf{R}, \mathbf{t})$ encode rotation and translation for a world coordinate system.
For two scenes $\mathbf{x}_1, \mathbf{x}_2$ with corresponding projection matrices $\mathbf{P}_1, \mathbf{P}_2$, the triangulation problem solves for $\mathbf{X}$ such that:

$$\begin{cases} \mathbf{x}_1 \times (\mathbf{P}_1\mathbf{X}) = \mathbf{0}, \\ \mathbf{x}_2 \times (\mathbf{P}_2\mathbf{X}) = \mathbf{0}. \end{cases} \tag{5}$$

2.1.5 Texture Mapping Texture mapping has many techniques; among them, UV mapping is widely used in computer graphics. It is efficient to produce appearance and refers to the parameterisation of a 3D mesh surface by assigning 2D coordinates (u, v), called UV coordinates, to each vertex of the mesh. These coordinates define a correspondence between points on the texture image and points on the 3D surface, controlling how the texture wraps around complex geometries.

2.1.6 Smoothing Function The smoothing function is a method that reduces noise and produces more coherent lines or surfaces. This operates by averaging or fitting local neighbourhoods of data to enforce continuity and eliminate outliers. The cKDTree algorithm is a binary space-partitioning tree that recursively subdivides the data space along axis-aligned hyperplanes, resulting in a hierarchical structure that enables logarithmic-time nearest-neighbour searches Bentley, 1975. Within a specified radius around a query point, this identifies a local neighbourhood and adjusts it by averaging.

3 Technical Approaches

This section conducts three different approaches, each proposed by Taekyung Kim, Luel Abrha Gebre and Kidu Abrha Welegerima in this order.

3.1 Approach 1 (Taekyung)

The main approach in this section is to reconstruct the 3D face with feature detection and matching algorithms, specifically SIFT combined with RANSAC by the OpenCV Python library OpenCV Team, n.d. The facial images are generated in Blender from five different transformations (translation and rotation). Starting from the front image, the left and right views are obtained by applying transformations consisting of translations along the x-axis ranging from ±0.05 m to ±0.25 m, and rotations about the z-axis ranging from ±5 degrees to ±25 degrees.
Additionally, to support a psychophysical evaluation of reconstruction quality, an AI method based on the Google MediaPipe Face Landmark Detection model, which is a series of models: Face detection model, Face mesh model and Blendshape prediction model, and detects 478 3D coordinates of a face, is introduced Google, n.d. This model enables an alternative 3D face estimation approach using dense 2D facial landmarks projected into 3D space. The classical 3D reconstruction is described in Sections 3.1.1 to 3.1.7, while Section 3.1.8 focuses on the AI model.

3.1.1 Data Preparation Since accurate camera parameters are important for the classical method of reconstructing 3D with epipolar geometry, Blender was selected as the data generation tool, simulating camera models. I have found the Marilyn Monroe 3D sculpture on the internet and rendered the facial images. The cameras in Blender were configured with the following parameters: focal length of 50 mm, sensor width of 36 mm, image resolution of 1920 x 1080 pixels, and the distance of approximately 1 m between the front camera and the object.
To ensure that feature detection focuses on facial regions only and avoids unnecessary background, the rendered images were cropped tightly around the face (see Figure 1). This preprocessing step improves the robustness and relevance of subsequent feature matching and 3D reconstruction.

3.1.2 Image Processing Figure 2 illustrates how CLAHE enhances contrast by making image gradients stronger without

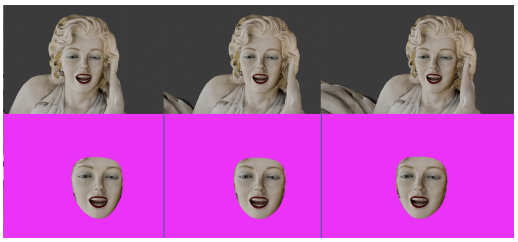

Figure 1. The top row is the original rendered images, and the bottom row is the cropped versions. From left to right, the scenes are ordered as left, front, and right. The example left and right scenes are translated ±0.10 m on the x-axis and rotated ±10 degrees on the z-axis from the front scene.

noise. This enhancement improves the performance of feature detection algorithms such as SIFT. For example, in the front scene grayscale image, 365 keypoints were detected, while the CLAHE image had 429 keypoints.

Figure 2. The top row is the cropped images with grayscale, and the bottom row is the contrast-increased images using CLAHE. From left to right, the scenes are ordered as left, front, and right. The left and right scenes are translated ±0.10 m on the x-axis and rotated ±10 degrees on the z-axis from the front scene.

3.1.3 Feature Detection The SIFT algorithm has several parameters ("nfeatures", "nOctaveLayers", "contrastThreshold", "edgeThreshold" and "sigma") to be set. Among these parameters, "contrastThreshold", which controls the minimum contrast required for a keypoint to be retained, is the most important for maximising feature detection. The lower value allows more keypoints to be detected in the low-texture or shadow spots. Through this, I was able to maximise the number of features detected from 429 to 1494 (see Figure 3). The parameters are determined by manually configuring them multiple times.

3.1.4 Feature Matching and Epipolar Lines To match the correspondences between facial scenes, keypoints detected by SIFT are matched across image pairs: left-front and front-right, using their descriptors via Euclidean distance. The RANSAC algorithm is applied with custom parameter settings: "ransacReprojThreshold"=2.0, "confidence"=0.99. Still, the results were not a big variant from the default parameters (For example, the left-front pair has 746 correspondences from the default parameters and 742 correspondences using customised parameters).
With the computed fundamental matrix that bridges between the two scenes by providing points where to lie in one image to the other, the epipolar lines are drawn (see Figure 4).

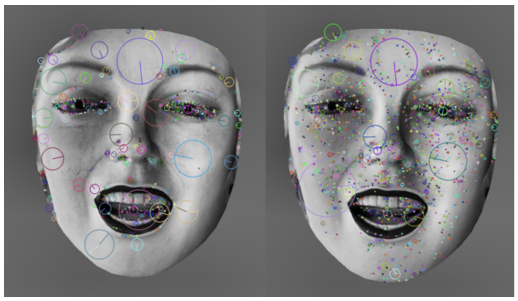

Figure 3. Feature detection results using SIFT: the left image is with default parameters, and the right image is with custom-tuned parameters (nfeatures=0, nOctaveLayers=3, contrastThreshold=0.01, edgeThreshold=5, sigma=1.56).

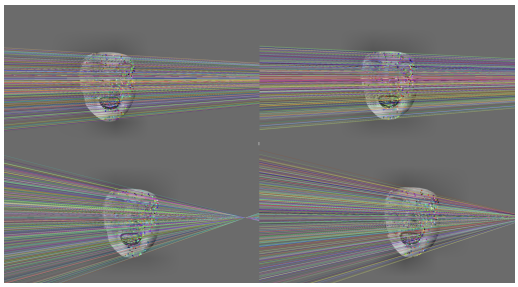

Figure 4. The results of feature matching: the top row is from the left-front pair, and the bottom row is from the front-right pair. The example left and right scenes are translated ±0.10 m on the x-axis and rotated ±10 degrees on the z-axis from the front scene.

3.1.5 Triangulation and Depth Filtering The 3D reconstruction is achieved through the triangulation method using camera parameters. Since the intrinsics and extrinsics of the camera are known from Blender, the essential matrix is not needed in this part. Each camera's projection matrix Equation 4 makes back-projecting 2D image correspondences into 3D. By intersecting these, the 3D points are estimated in homogeneous coordinates and converted to Euclidean space. The reconstruction result is shown in Figure 14.
Following triangulation, the output, the image on the left in the bottom row of Figure 14, shows that most of the depth is calculated between 0.84 m and 0.92 m, and some coordinates have errors due to geometric inconsistencies or errors of matching. This is why the depth filtering is required to reduce errors, and the filtered output demonstrates a cleaner and more accurate point cloud (see the right image on the bottom row in Figure 14).

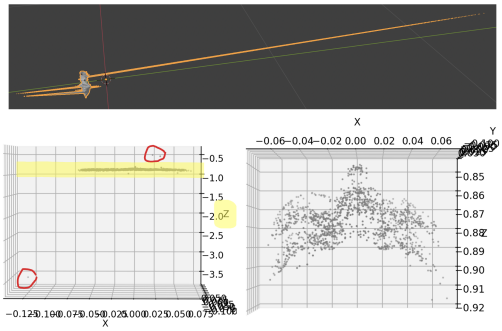

Figure 5. The top row represents an example of wrong depth reconstruction in Blender. In the bottom row, the left image is a 3D point cloud with wrong coordinates (pointed in red), and the right image shows a modified point cloud after depth filtering. The result from images with translations of ±0.10 m on the x-axis and rotations of ±10 degrees on the z-axis.

3.1.6 Mesh Generation and Texture Mapping Only with the reconstructed 3d point cloud, it is hard to distinguish the face because there are no textures or appearance. Therefore, the valid 3D points are projected back onto the front-view image using the camera projection matrix. The resulting 2D coordinates are normalised for image dimensions to form UV texture coordinates, which are the vertices of the mesh (see Figure 6).
To create a surface from the point cloud, Delaunay triangulation is used, and this makes a set of triangular faces that represent the mesh topology. These triangular faces form the final mesh with 3D vertices and UV coordinates.

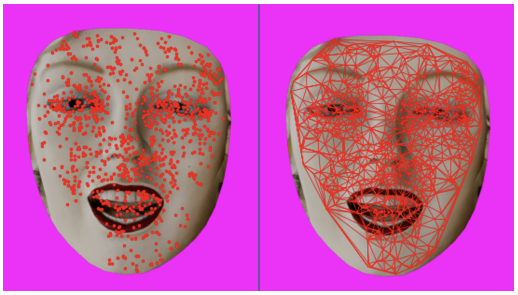

Figure 6. The left image shows the 2D coordinates on the face, while the right image is the UV texture map. The result from images with translations of ±0.10 m on the x-axis and rotations of ±10 degrees on the z-axis.

3.1.7 Surface Smoothing Figure 7 indicates the importance of smoothing the surface. Right after triangulating the point cloud, the local surface persists irregularity due to noise, correspondence errors, etc. To mitigate this, a smoothing function is applied to the surface, which is the depth, z-axis. This smoothing process reduces surface artefacts significantly and results in a better visualisation of the face.
The cKDTree, which is a radius-based algorithm, adjusts the depth by averaging all points. The radius parameter was determined by multiple testing as 0.02 (2 cm).
Through the process, the face can be reconstructed in 3D from three images. In psychophysical experiments, this process is repeated for other transformation conditions, introduced in the beginning, to understand how image deformation affects reconstruction.

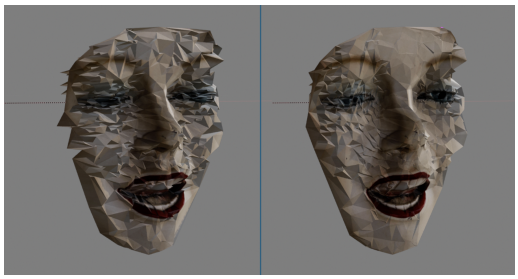

Figure 7. The left image shows the reconstructed face before smoothing, and the right image is after applying the smoothing function. The result from images with translations of ±0.10 m on the x-axis and rotations of ±10 degrees on the z-axis.

3.1.8 Reconstruction using Face Landmark Detection Model For the Face Landmark Detection model Google, n.d., which outputs estimated 3D coordinates of a face, a single face scene is enough to reconstruct a 3D face. However, since this model provides only geometric data, texture mapping is required to visualise it. Therefore, the same UV texture mapping approach described earlier in Section 3.1.6 is applied.
The image on the right in Figure 8 shows the result of the reconstruction. The front scene has been used to have facial data evenly from any angle.

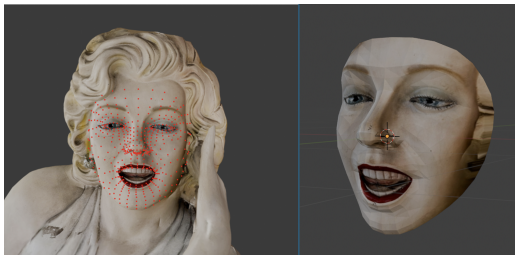

Figure 8. The left image shows the face landmarks detected by the Face Landmark Detection model, and the right image is the 3D reconstruction result with UV texture mapping.

3.1.9 Combination of classical and Face Landmark Detection Model methods Following the professor's suggestion from the presentation to improve reconstruction quality, I integrated the classical reconstruction method with the AI model. Figure 9 demonstrates that this combined approach makes a higher density of keypoints than Figure 6. This provides a more precise facial shape. Although I encountered an obstacle to overcome that the 3D point clouds from the classical method and the AI model could not be normalised together in a consistent coordinate space. The depth of 3D coordinates from the AI model and the classical method was variant. I have tested another face, but I could not find a pattern to determine the depth from the AI model. However, adjusting the depth manually could be an option, but not efficient. Therefore, unfortunately, I was not able to produce a viable output.

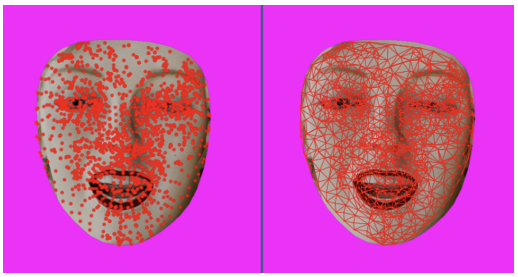

Figure 9. The output of combination point clouds: the left image shows the 2D coordinates on the face, while the right image is the UV texture map.

3.2 Approach 2(Luel)

The pipeline consists of four main stages: (1) facial landmark extraction, (2) relative pose estimation, (3) 3D triangulation, and (4) mesh reconstruction with texture mapping.

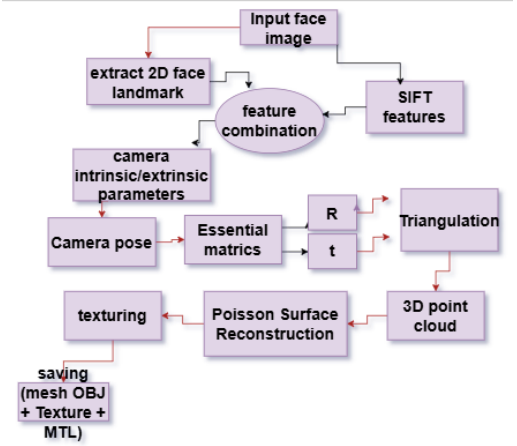

Figure 10. Architecture of the model

3.2.1 Facial Landmark extraction The media Pipe Face Mesh Google, n.d. has been used to detect a 2D 478 facial landmarks per image. Given an input image I, the model outputs a set of 2D landmarks pi=(ui,vi) where I belongs to [1,…,468].

3.2.2 Scale invariant feature transform: My approach combines landmark-based facial structure estimation with feature matching to improve robustness in textureless regions. First, I detect SIFT (Scale-Invariant Feature Transform) features Lowe, 2004 and filter them near facial landmarks. Next, a FLANN (Fast Library for Approximate Nearest Neighbors)-based feature matching is applied, incorporating Lowe's ratio test Lowe, 1999 to discard ambiguous matches.

3.2.3 Camera Pose Estimation Given three views of a face, we compute the relative pose (rotation R and translation t) using the 5-point algorithm Nistér, 2004. The essential matrix E is estimated via RANSAC to handle outliers:

3.2.4 3D Triangulation Using the estimated poses, I triangulated 3D points XiXj from the combined matched landmarks pi(1)pj(1)-front-left , pi(2)pj(2)-front-right and SIFT points/descriptors.

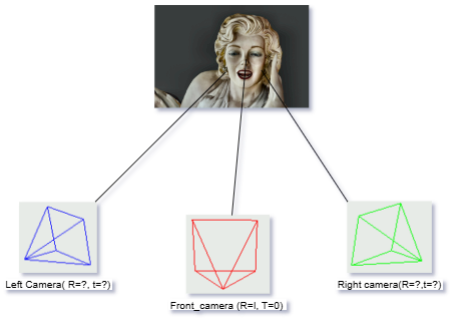

Figure 11. Schematic illustration

3.2.5 Mesh Reconstruction and Texture Mapping: The Poisson reconstruction method Hou et al., 2022 converts the point cloud into a watertight mesh. Texture enhancement is then applied through the following steps:

- **CLAHE (Contrast-Limited Adaptive Histogram Equalization):** Enhances local contrast.
- **HSV color correction:** Normalizes skin tones for consistency.

Finally, UV mapping ensures accurate texture alignment on the 3D mesh.

3.3 Approach 3 (Kidu)

In this study, I investigated the problem of 3D face reconstruction from stereo images under three distinct scenarios using the camera projection matrix presented in Equation 6 University, 2025. The **first scenario** involves reconstruction from the **calibrated setup**. The **second scenario** discards the extrinsic parameters and **estimates the relative camera pose** directly from the stereo image pairs. In the **third scenario**, the 3D face is reconstructed from an **uncalibrated camera setup** by estimating both intrinsic and extrinsic parameters K(R/T) from the images.

$$w \underbrace{\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}}_{x} = \underbrace{\begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}}_{K} \underbrace{\begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix}}_{[R\,|\,T]} \underbrace{\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}}_{X} \tag{6}$$

To conduct the experiments, a calibrated 3D face was created in Blender, featuring three cameras positioned to capture the left, front, and right views of the face from different angles and translations. The front camera remains fixed at the center and serves as a reference, while the left and right cameras are varied in pose across four configurations to assess the impact of rotation and translation on reconstruction performance. All translations are along the x-axis and all rotations are around the z-axis. All distances are expressed in meters (m), with negative values for the left camera and

positive values for the right camera. **The datasets was captured under four calibration setups, which are as follows:**

1. Translation of ±0.1 m without rotation.
2. Rotation of ±5° combined with ±0.1 m translation.
3. Rotation of ±10° combined with ±0.2 m translation.
4. Rotation of ±20° combined with ±0.4 m translation.

After setting up the scene, a Python script was written to render images and extract the camera parameters and per-pixel depth from each camera. The script generates three images (left, front and right), along with their corresponding camera parameters saved in JSON format and ground truth depth maps for each image saved as.exr files.

To evaluate the accuracy of the reconstructed 3D geometry in different scenarios and camera setups, ground-truth depth maps were rendered at the pixel level from each camera perspective. In addition to depth-based evaluation, the accuracy of the reconstructed 3D points was assessed using the epipolar and reprojection error. Reprojection error is computed as the average Euclidean distance (in pixels) between the original 2D keypoints and the reprojected 2D points obtained by projecting the triangulated 3D coordinates back onto the image plane. The reprojection error directly measures how accurately the reconstructed 3D geometry aligns with the original image observations by evaluating how close the projected 3D points are to the actual image points. In contrast, the epipolar error is used to assess the geometric consistency between image pairs by evaluating how well the corresponding points satisfy the epipolar constraint.

4 Experiments/Analysis

This section discusses the experiment and analysis for reconstruction results.

4.1 Approach 1 (Taekyung)

Following the classical approach in Section 3.1, Figure 12 shows five reconstructions using different transformation datasets. Overall, the triangulation at 3.1.5 was performed properly for all results because the depth range between 0.84 m and 0.92 m was accurate compared to the real camera and object settings (the distance from the front camera to the object is about 1 m). I was able to perceive that the first result is well-reconstructed the most, and the quality (face shape, surface homogeneity, etc) goes down as the transformation increases.

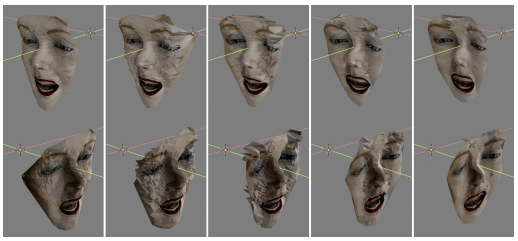

Figure 12. The reconstruction results in five different transformations are displayed from left to right, with translations along the x-axis increasing from ±0.05 m to ±0.25 m in 0.05 m increments, and rotations about the z-axis increasing from ±5 degrees to ±25 degrees in 5-degree increments. The leftmost result corresponds to ±0.05 m translation and ±5 degrees rotation, followed by ±0.10 m and ±10 degrees in the second image from the left.

To evaluate the results effectively, a psychophysical experiment was conducted. Observers were first asked to select the most realistic reconstruction from a set of results in Figure 12 produced using the classical method. In the second part, they compared their chosen classical reconstruction against the reconstruction by an AI-based method.
For this, I have created a website myself with a mobile-friendly UI and hosted it to increase people's engagement. Through this, I was able to have 38 participants, and Figure 13 indicates their selections.

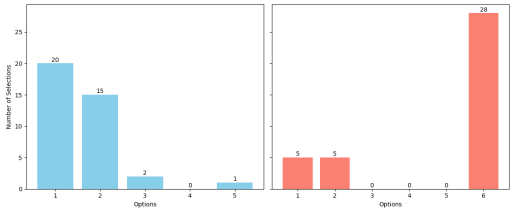

Figure 13. For the first plot from the left, the options 1 to 5 represent different conditions of the camera transformation increase in the first question, from 5 to 25. The result of the second question in the second plot is the result of the comparison of the chosen classical reconstruction from the first question against the AI-based reconstruction denoted 6.

In the first question of the experiment, most observers have chosen options 1 or 2, which are low camera transformation settings. Specifically, Option 1 involved a translation of ±0.05 m and a rotation of ±5 degrees, and received the highest number of selections. This indicates that smaller camera transformations make higher perceived reconstruction quality when using Approach 1.
Regarding the second question, Option 6, the AI-based reconstruction, received the highest number of selections, with 28. As shown in Figure 8, the reconstruction exhibits a smooth surface with small bumpiness, unlike the classical method reconstructions. Notably, facial features such as the nose and mouth are well defined in terms of elevation, and overall, the face shape is well preserved. I may guess these are likely to contribute to the preference for observers.

4.2 Approach 2 (Luel)

Three face images from different directions of a single face image (front, left, right) are rendered from blender with intrinsic and extrinsic camera parameters obtained from the blender. For the

Figure 14. sample datasets used

evaluation metrics, a reprojection error is used to show how the quality of the 3d reconstructed face is good enough quantitatively. **How does the reprojection error work**? After triangulation, the 3D points are projected back into each camera (left, right) using their projection matrices (P = K @ Rt). The projected 2D points are compared with the original detected points.

In my camera calibration experiments, I compared two different scenarios: the reconstruction with known intrinsic and extrinsic camera parameters and known intrinsic but unknown

extrinsic parameters. The essential matrices are used to estimate the rotation and translation of the two cameras by setting the front camera as a reference.

Metrics	Configuration	
	Known intrinsic and extrinsic	Known intrinsic only/estimating R and T
Mean error	4.5 px	15 px
RMS	6.27 px	21 px
Median	3.09 px	11 px
Inlier ratio	70.92%	45.8%

Table 1: Reprojection error results for both setup

The results clearly show that knowing both intrinsic and extrinsic parameters yields significantly better performance across all metrics, with the mean reprojection error improving from 15 px to just 4.5 px when full calibration data is available. The inlier ratio shows particularly dramatic improvement, jumping from 45.8% to 70.92%.

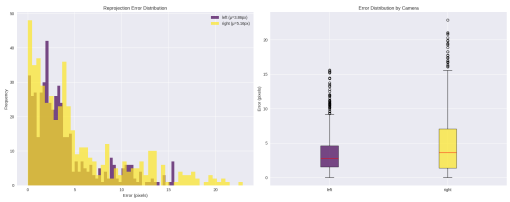

Figure 15. box plot-error distribution by camera (right side) reprojection error distribution(right side) known intrinsic+extrinsic parameters

The reprojection error distribution figure 15 shows the individual errors for the left- and right-face images relative to the front-face image, with mean values of 3.86 px and 5.16 px, respectively.

figure 16 shows frequency distribution of reprojection errors,

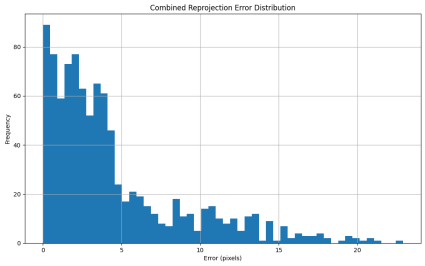

Figure 16. combined reprojection error distribution known intrinsic+extrinsic parameters

the taller peaks (high frequency) at lower value shows good reconstruction quality. A box plot figure 15 Visualization error statistics across cameras also shows 25th-75th percentiles.

From figure 17, we observe that the red color indicates a high error rate, which occurs on both sides of the view and is more concentrated on the cheeks due to texture-matching issues. These

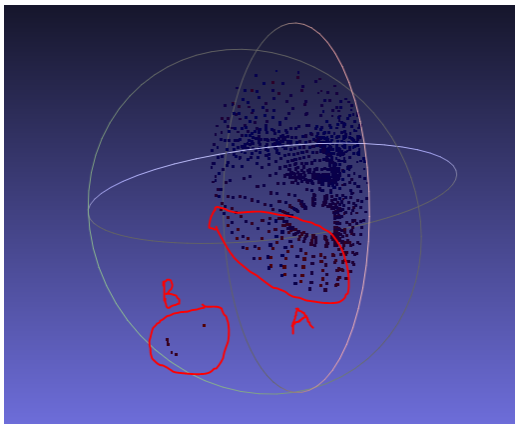

Figure 17. Error visualization, known intrinsic+extrinsic parameters

issues contribute significantly to the overall mean reprojection error. Additionally, there are very small outlier features (B) that also exhibit a significant error accumulation in the reprojection error.

The reconstructed mesh preserves fine details, such as lip con-

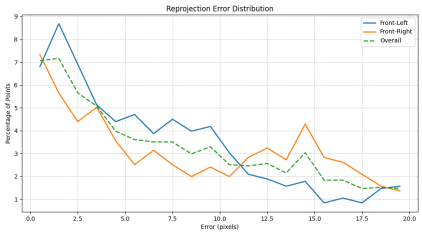

Figure 18. reprojection error distribution in uncalibrated camera

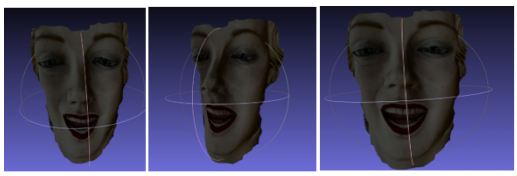

Figure 19. reconstructed result, calibrated camera/intrinsic+extrinsic parameters. Although SIFT detects features across the entire image, I intentionally eliminate unstable matches (which often correspond to background features), retaining only those features that are geometrically consistent with facial landmarks

tours. Outlier rejection via RANSAC ensures robustness against occlusions. Visually, we can observe that the reconstructed 3D face aligns well with the face image when using Google's facial landmarks Figure 8.

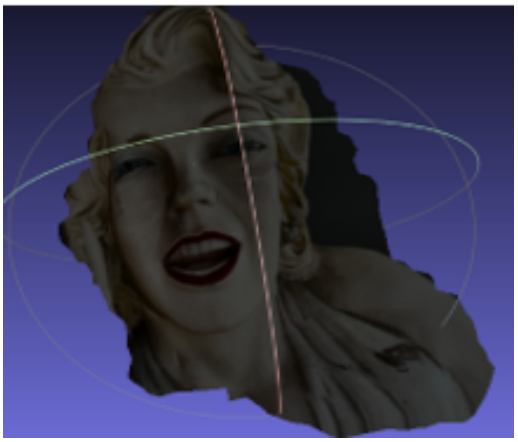

Figure 20. reconstruction result, unknown extrinsic parameters

4.3 Approach 3 (Kidu)

In this section, 3D face reconstruction using calibrated, semi-calibrated, and uncalibrated setups will be evaluated on four different datasets captured from varying camera poses.

4.3.1 Calibrated Setup: I use SIFT to detect key points and compute descriptors for each image. After extracting the key points, the matching features were identified between the pair of images, left front, right front, and left right, using the k-nearest matching algorithm. To ensure geometric consistency and eliminate outliers, I applied a RANSAC-based Fundamental Matrix estimation to refine the matches. Using the front camera as a reference, projection matrices were constructed from the intrinsic and relative poses parameters. I then triangulated the matched points between each image pair to reconstruct 3D points and combined them into a sparse 3D point cloud in the front camera's coordinate frame.

The reconstructed sparse 3D points are evaluated by comparing them to ground-truth 3D points obtained by back-projecting depth maps from all camera views into world coordinates. Each reconstructed point is matched to its nearest ground-truth neighbor. Evaluation metrics such as mean error, RMSE, inlier ratios, and coverage are calculated to assess precision and completeness. Moreover, the reprojection error is computed as the average pixel distance between the reprojected 3D points and their corresponding 2D keypoints, indicating how well the reconstruction aligns with the original images. The lower error reflects better triangulation and camera calibration.

Setup	Matched Keypoints	Mean error	RMSE	Coverage area	Inlier <1 cm	Inlier <5 cm	Riproj. Err. (px)
1 (0°)	19065	0.33 mm	0.55 mm	100%	99.94%	100%	0.05
2 (5°)	16141	7.5 mm	9.1 mm	100%	69.57%	100%	0.06
3 (10°)	14130	14.65 mm	18.44 mm	100%	40.3%	99.5%	0.06
4 (20°)	5655	49.67 mm	54.75 mm	99.27%	3.85%	49.8%	0.07

Table 2: Quantitative evaluation metrics for each setup.

In Table 2, the coverage indicates how much the reconstructed point found a valid match in ground truth within 1 cm. The mean and RMSE measures how much the predicted point cloud is away from ground truth point cloud. The inlier ratio tells you how many of the reconstructed points fall within a given distance threshold from ground truth.

Based on the quantitative result in Table 2, as both rotation and translation between stereo cameras increase, the quality of 3D reconstruction tends to decline. Moderate rotation combined with small translation offers an optimal balance, preserving features overlaps and minimizing geometric distortion. However, as the baseline widens and the cameras rotate further, feature matching becomes less reliable due to increased occlusions and greater viewpoint differences. This results in fewer valid correspondences and less stable triangulation, ultimately reducing the accuracy and consistency of the reconstructed 3D points. Although wider baselines can theoretically improve depth sensitivity, they require more robust calibration, feature matching, and refinement techniques to maintain reconstruction quality. The results in the table clearly demonstrate that as rotation and translation increase, the estimated depth deviates more significantly from the ground truth depth captured by the front camera.

Example of Qualitative Result from Calibrated Setup

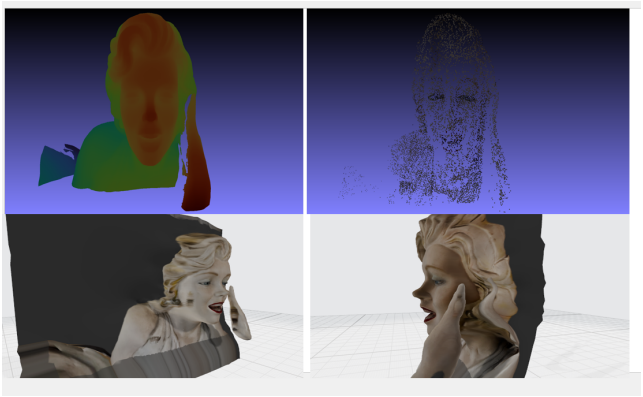

Figure 21. The top-left image shows the ground truth depth, and the top-right image shows the reconstructed 3D sparse point cloud, while the second row shows the result after applying texture to the reconstructed 3D sparse point cloud.

4.3.2 Semi-Calibrated Setup: Known K, Estimated Pose: In this scenario, 3D points are reconstructed from stereo images by matching SIFT features, estimating the essential matrix with RANSAC using known intrinsics, recovering relative camera pose, and triangulating inlier matches to form the 3D point cloud. The qualitative evaluation of the 3D face reconstruction using this process on the four datasets is provided in Table 3 below.

Setup	Total Keypoints	Essential Inliers	Riproj. Err.	Triangulation Angle(depth accuracy)		
		(%)	(px)	Min (°)	Mean (°)	Max (°)
1 (0°)	13388	86.6	0.07	2.5	6.1	15.5
2 (5°)	11865	84.5	0.11	3.7	5.5	16.7
3 (10°)	12432	81.4	0.20	10.2	12.8	16.3
4 (20°)	7306	71.7	0.10	11.5	23.3	26.6

Table 3: Quantitative evaluation metrics for each setup.

From Table 3, as the camera setup transitions from minimal rotation and translation (Setup 1) to wider baselines and increased angular separation (Setup 4), the 3D reconstruction quality shifts from high reprojection precision to improved depth accuracy. Setup 1 provides the most keypoints and lowest reprojection error but suffers from poor triangulation geometry, making depth estimates less reliable. Introducing moderate rotation and trans-

lation in Setups 2 and 3 leads to better triangulation angles and more accurate 3D structure, though at the cost of slightly higher projection errors. Setup 4, with the largest baseline and rotation, yields the most accurate depth due to strong triangulation but has fewer keypoints and reduced robustness. Overall, increasing baseline and parallax improves depth accuracy while slightly compromising feature match density and projection consistency.

Example of Qualitative Result from Pose Estimation

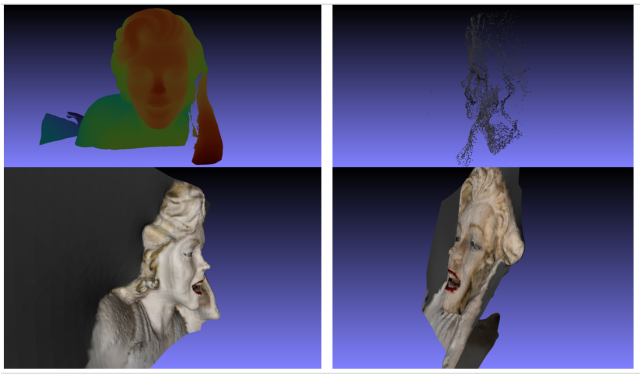

Figure 22. The top-left image shows the ground truth depth, and the top-right shows the reconstructed 3D point cloud, whereas the second row displays the results after applying texture.

4.3.3 Uncalibrated Reconstruction: Joint Estimation of K, R, and T: The system first detects, and matches SIFT features between stereo images to establish point correspondences. It then self-calibrates the camera intrinsics (K) by estimating the fundamental matrix (F) using RANSAC, assuming the principal point lies at the image center, and optimizing the focal length (f) so that the singular values of the essential matrix $E = K^\top FK$ approximate the ideal $[\sigma, \sigma, 0]$ structure Hartley, 2003. With the estimated intrinsics, the system computes the essential matrix to recover the relative camera pose and performs stereo rectification to align the epipolar lines, thereby simplifying the correspondence problem, as shown in Figure 23. The pipeline then triangulates the matched points into 3D coordinates, filters out unrealistic depths and applies statistical outlier removal to generate an accurate and sparse 3D point cloud without requiring prior camera calibration. The qualitative evaluation of the 3D face reconstruction using this process on the four datasets is provided in Table 4 below.

Setup	Keypoints	Reproj. Error (px)	Epipolar Error (px)	Vertical Disparity (px)	Point Count	Depth Range (m)	
						Min	Max
1 (0°)	4500	0.08	0.0025	0.16	3780	1.79	2.43
2 (5°)	3996	0.09	0.0020	0.16	3358	8.44	11.04
3 (10°)	3801	0.082	0.0046	0.134	3159	4.46	5.44
4 (20°)	2030	0.11	0.0110	9.24	1667	2.33	2.71

Table 4: Quantitative evaluation of the uncalibrated setup.

To evaluate 3D reconstruction quality, epipolar error was used to assess geometric consistency and reprojection error to measure reconstruction accuracy, along with additional point cloud statistics. The results demonstrate that uncalibrated stereo reconstruction performs best under small camera rotations ($\leq 5°$) and short baselines (0.1 m), yielding dense point clouds, low errors, and reliable depth estimates. Pure translation, despite geometric alignment, results in less stable depth due to poor parallax. Increasing the baseline to 0.2 m can improve depth precision but introduces occlusion and reduces match quality. At extreme configurations such as setup 4 rectification becomes unstable, with vertical disparities exceeding 9 px, leading to sparse, inaccurate reconstructions. Depth range was found to be more influenced by camera angle than baseline, with near-parallel views offering the most consistent coverage. These findings underscore the limitations of uncalibrated methods under large viewpoint differences.

Example of Qualitative Result from Uncalibrated setup

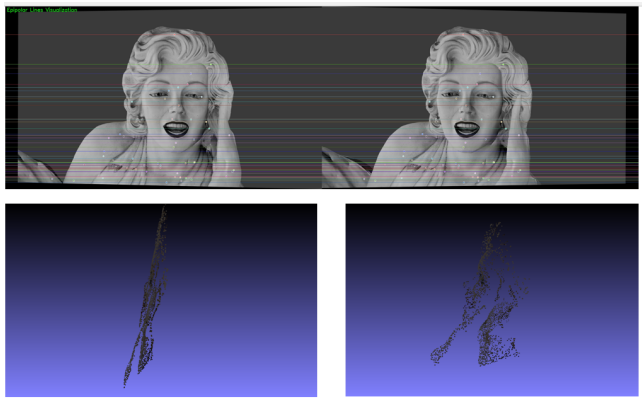

Figure 23. The top row shows the rectified images, with the epipolar lines aligned horizontally, whereas the bottom left shows the 3D point cloud estimation of setup 1, and the last one shows the estimation of setup 2 without camera parameter information.

Figure 23 shows that although both setups share the same baseline (0.1m), the absence of rotation in Setup 1 results in poor depth estimation despite good keypoint matches. In contrast, the 5° rotation in Setup 2 improves depth estimation by enhancing triangulation geometry.

5 Conclusion

5.1 Approach 1 (Taekyung)

This approach focuses on the 3D reconstruction of human faces from three 2D images (left, front and right scenes) using the classical computer vision method with epipolar geometry. In addition, by adopting Blender, the precise images and camera parameters were able to be prepared easily in the research.

For the key steps, CLAHE, SIFT and RANSAC algorithms are introduced to maximise the detection of the number of keypoints and match them. These correspondences were applied for the triangulation to reconstruct the 3D point cloud. Notably, the triangulation was performed with high precision to estimate depth values closely matching the real depth. The result faced some error reconstructions initially, but this has been overcome using the depth filtering solution.

Mesh generation and texture mapping were performed to make the 3D visualisation from the point cloud, projecting the 3D coordinates back onto the front scene image. However, the result was still unsatisfactory due to noise and irregularities on the skin surface. To solve this, a smoothing function, the cKDTree algorithm, was applied and was able to return a much better result.

In parallel, the AI model, the Google MediaPipe Face Landmark

Detection model, is made for the comparison and integration between the classical and AI methods. For the integration, this increased the density of the keypoints with a certain face shape, but was not successful due to difficulties in normalising the 3D point cloud and depth.
The psychophysical experiment was conducted using five different sets of camera transformations and the AI-based approach with 38 participants in total via a mobile web. Overall, the smaller camera transformations, especially ±0.05 m translation and ±5 degrees rotation, in the classical method provided better quality, and the AI-based method received the majority of selections between the two methods. In conclusion, we can know that the classical method using epipolar geometry can perform and obtain that it is possible to reconstruct a face only with three images, but the AI-based method is better. Moreover, we perceived quality decreasing as camera transformations increased in the classical method.

5.2 Approach 2 (Luel)

In this work, I presented a robust and interpretable multi-view stereo 3D face reconstruction pipeline that combines classical geometric methods with hybrid keypoint detection using MediaPipe facial landmarks and SIFT features. By leveraging essential matrix decomposition for relative pose estimation and Poisson surface reconstruction for dense mesh generation, this approach achieves sub-pixel reprojection accuracy under known camera parameters and maintains structural consistency even when extrinsic are estimated. The use of adaptive texture enhancement techniques such as CLAHE and HSV-based color correction further improves visual fidelity.
The quantitative evaluation in Table 1 underscores how the accuracy of the pose governs the final accuracy. With both intrinsics and extrinsics of the camera known, the system achieves a mean re-projection error of 4.5 px (RMS 6.27 px; median 3.09 px) and retains 70.92% inliers, confirming the precision of subpixel triangulation once the radial distortion is taken into account. When extrinsics are estimated (intrinsics only), the error rises to 15 px across all statistics and the inlier ratio collapses to 45.8%, revealing the sensitivity of dense facial geometry to camera pose quality. Unlike deep learning-based methods, this geometry-driven pipeline offers transparency, low computational cost, and does not require extensive training data. Overall, this technique offers an effective and accessible solution for high-quality 3D face modeling using only a small set of RGB images and known intrinsics.
In future work, integrating learned feature descriptors (e.g. SuperPoint, LoFTR) with geometric constraints could improve keypoint matching in low-texture regions, enhancing robustness and precision.

5.3 Approach 3 (Kidu)

In this work, my comparative study of 3D face reconstruction across calibrated and uncalibrated stereo setups demonstrates clear performance trends. The fully calibrated system achieved the highest accuracy, reliably reconstructing facial structure with minimal errors. In contrast, uncalibrated approaches showed increasing sensitivity to camera geometry. When only intrinsics were known, accuracy declined slightly, whereas in the fully uncalibrated case, performance degraded significantly under large baselines and rotations, remaining reliable only under limited viewpoint differences.

The experiments highlight the trade-off between camera baseline, rotation, and reconstruction accuracy. Moderate baselines and rotations provided the best balance between depth precision and reliable feature correspondence. Small baselines offer high feature overlaps but poor depth estimation due to narrow triangulation angles. In contrast, wide baselines improve parallax and theoretical depth resolution but degrade matching quality due to increased occlusions and viewpoint disparity. As a result, extreme camera separations reduced reconstruction accuracy across both calibrated and uncalibrated setups. These findings confirm that an optimal intermediate baseline and rotation maximize 3D reconstruction fidelity.

Future work could explore improving occlusion handling via multi-view integration, enhancing feature matching under wide viewpoint differences and challenging conditions like lighting changes, and incorporating learning-based depth estimation for more accurate 3D face reconstruction from stereo images.

References

Bentley, Jon Louis (1975). "Multidimensional binary search trees used for associative searching". In: *Communications of the ACM* 18.9, pp. 509–517.

Fischler, Martin A and Robert C Bolles (1981). "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography". In: *Communications of the ACM* 24.6, pp. 381–395.

Google (n.d.). *MediaPipe Face Landmarker*. `https://ai.google.dev/edge/mediapipe/solutions/vision/face_landmarker`. Accessed: 2025-05-30.

Hartley, Richard (2003). *Multiple view geometry in computer vision*. Vol. 665. Cambridge university press.

Hou, Fei et al. (2022). "Iterative Poisson surface reconstruction (iPSR) for unoriented points". In: *arXiv preprint arXiv:2209.09510*.

Longuet-Higgins, H Christopher (1981). "A computer algorithm for reconstructing a scene from two projections". In: *Nature* 293.5828, pp. 133–135.

Lowe, David G (1999). "Object recognition from local scale-invariant features". In: *Proceedings of the seventh IEEE international conference on computer vision*. Vol. 2. Ieee, pp. 1150–1157.

— (2004). "Distinctive image features from scale-invariant keypoints". In: *International journal of computer vision* 60, pp. 91–110.

Nistér, David (2004). "An efficient solution to the five-point relative pose problem". In: *IEEE transactions on pattern analysis and machine intelligence* 26.6, pp. 756–770.

OpenCV Team (n.d.). *OpenCV Tutorials*. `https://docs.opencv.org/4.x/d9/df8/tutorial_root.html`. Accessed: 2025-05-30.

Reza, Ali M (2004). "Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement". In: *Journal of VLSI signal processing systems for signal, image and video technology* 38, pp. 35–44.

University, Brown (2025). *CSCI 1430: Computer Vision*. `https://browncsci1430.github.io/`. Accessed: 2025-06-05.

Yoon, Youngjoo (2021). *aitimes*. URL: `https://www.aitimes.com/news/articleView.html?idxno=139228`.

Zhang, Zhengyou (2002). "A flexible new technique for camera calibration". In: *IEEE Transactions on pattern analysis and machine intelligence* 22.11, pp. 1330–1334.