

Opleiding Bioinformatica

3D mesh optimization strategy for images acquired from Bioimaging

Amarise-Jourmaine Silié

Supervisors: Fons Verbeek & Lu Cao

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS) <u>www.liacs.leidenuniv.nl</u>

05/07/2022

Abstract

The analysis of biological structures on a macroscopic and microscopic scale can lead to new insights in the biology research field. By gathering bioimages from microscope systems, extracting the point cloud and converting it to a surface representation, computerized threedimensional (3D) models can be generated for measurement and visualisation. In this paper, a pipeline is proposed to improve the quality of the 3D mesh derived from 3D bioimages. This will in turn increase the accuracy of the 3D representation. For this research, a zebrafish embryo model and a mouse mammary gland model are used. The contour stack derived from the raw image data is first aligned and then converted to an oriented point cloud. Three surface reconstruction algorithms (Screened Poisson, Alpha Shapes and Ball Pivoting) are then used to generate the 3D models. The results are compared to determine which algorithm can be best used for the pipeline. Looking at the zebrafish embryo it can be concluded that the model generated by the Screened Poisson method adheres to the evaluation criteria the most. Aside from creating a smooth watertight surface, the model contains the most preserved details. Due to inconsistent orientation of the point cloud normals, the mammary gland model could not be constructed optimally for two of the three algorithms. This is a point of improvement that can be worked on in future development of the pipeline.

Acknowledgements

I would like to thank my supervisors Fons Verbeek and Lu Cao from LIACS for proposing this research subject. I especially would like to thank Lu for her guidance throughout the trajectory of this project. A special thanks to my parents, little sister and my friends for their support.

Contents

1	Intr	Introduction 1			
	1.1	Problem statement	1		
	1.2	Related work	1		
	1.3	Thesis overview	2		
2	Methodology 3				
	2.1	Data Acquisition	3		
	2.2	Image Alignment	4		
	2.3	Point Cloud Extraction	5		
	2.4	3D Surface Reconstruction	6		
		2.4.1 Screened Poisson reconstruction	6		
		2.4.1.1 Parameters	6		
		2.4.2 Alpha Shapes reconstruction	7		
		2.4.2.1 Parameters	7		
		$2.4.2.2 \text{Post-processing} \dots \dots \dots \dots \dots \dots \dots \dots \dots $	7		
		2.4.3 Ball-Pivoting reconstruction	7		
		$2.4.3.1 \text{Parameters} \dots \dots \dots \dots \dots \dots \dots \dots \dots $	8		
		$2.4.3.2 \text{Post-processing} \dots \dots \dots \dots \dots \dots \dots \dots \dots $	8		
	2.5	Evaluation criteria	8		
3	Results and Evaluation 9				
	3.1	Screened Poisson reconstruction	9		
	3.2	Alpha Shapes reconstruction	10		
	3.3	Ball Pivoting reconstruction	11		
4	Discussion 12				
	4.1	Comparison	12		
	4.2	Limitations	14		
5	Cor	nclusion	16		
R	efere	ences	18		

1 Introduction

1.1 Problem statement

In biology, researchers aim to gain insights in the interrelations between structure, function and evolution of organisms. To investigate the spatial assembly of biological structures on a macroscopic and microscopic scale, three-dimensional (3D) models are derived [LIS⁺12]. An example of this process includes collecting the data using microscope systems. The acquired stack of images can be processed by performing segmentation or manual delineation [CV14]. The resulting contours can be reconstructed to form a 3D model. A way to do this is by using a point cloud based surface reconstruction method, as done by [CV14]. Here, a point cloud, which is a set of points in a 3D space, is extracted from the contour stack. Subsequently, the point cloud can be converted to a surface representation, which is composed of a 3D mesh. The issue is that the quality of the mesh can be affected by noisy data. This also affects the accuracy of the 3D model in respect to the object of interest. A possible consequence is that important details of biological structures are compromised or not visible in the 3D model. Therefore, the goal of this research is to improve the quality of the 3D mesh from image stacks, using an optimization strategy. The research question is as follows: what kind of optimization strategy can be used for optimizing the mesh of 3D bioimages?

In this paper, a pipeline is proposed that aims to optimize the 3D mesh of models constructed from bioimages. It is based on the point cloud based pipeline proposed by [CV14].

1.2 Related work

In the science research field, 3D reconstruction for analysis of biological structures is widely used. For example, in neuroscience, a model for the 3D reconstruction of neuron structures was proposed by [PRAS10]. More recently, [XLD⁺18] developed an automated pipeline which enables the 3D reconstruction of synapses. 3D reconstruction is also used for plant phenotyping. In [Rav16], a point cloud based method is developed for the reconstruction of apple trees. A recent study [DCMS⁺20] shows that 3D phenotypes can potentially give new insights in the morphological characteristics of plants that are influenced by interactions between the genotype and the environment. Another contribution to the research field is the pipeline developed by [CV14]. The pipeline performs contour interpolation on the collected contour stack and then applies the Poisson reconstruction method on the derived oriented point cloud. Ultimately, it is used for phenotype measurement. Based on previous work, it can be concluded that the 3D reconstruction of biological structures is important for the life science research field.

Reconstruction methods can be categorized as an implicit or an explicit approach. Explicit techniques, such as the ball-pivoting algorithm by [BMR⁺99], are usually straightforward. However, they do not always generate a watertight surface [Cha21]. In contrast, implicit techniques do produce a watertight mesh and are also robust against noise and non-uniformly distributed point clouds. However, the produced models might be less accurate, depending on the voxel resolution [Cha21]. An example of such a method is Poisson reconstruction by [KBH06]. In this paper, both methods are used and compared to observe their differences and efficiency in practice.

1.3 Thesis overview

This paper proposes a 3D mesh optimization pipeline. An overview of the pipeline can be found in figure 1. The paper is structured as follows. Firstly, the methodology, including the data acquisition and steps of the developed pipeline, will be explained in Section 2. This section also contains the selected criteria with which the optimization strategy will be evaluated. Then the resulting 3D models will be analyzed and evaluated in section 3. The 3D models will also be compared for each 3D reconstruction method in order to formulate the limitations of the optimization strategy and possible improvements. This is discussed in section 4. A conclusion and the answer of the research question is presented in section 5.



Figure 1: An overview of the pipeline.

2 Methodology

2.1 Data Acquisition

The acquisition of the data used in this research was performed by specialists. Two models are used in this thesis: a zebrafish (Danio rerio) embryo model and a mouse mammary gland model, the latter of which is more complex. Imaging of the zebrafish embryo is performed using a confocal microscope (see figure 2), while the raw image data of the mammary gland (see figure 3) is collected using a bright field microscope. Afterwards, the images undergo delineation to extract the structures of interest. For each slice, delineation is performed either manually or via automated segmentation. In the case that it is performed manually, annotation software can be used. In this case annotation software TDR [VdH⁺93] is used in combination with a WACOM digitizet tablet (WACOM, Cintiw LCD-tablet) [CV14]. The derived contours will be used for the next step of the pipeline.



Figure 2: Image data of the zebrafish model. The size of the embryo images are 1300x1000. (a) the original image taken with the confocal microscope. (b) the binary mask of the original image. (c) the derived contour, which will be used for the rest of the pipeline.



Figure 3: Image data of the mouse mammary gland model. The size of the mammary gland images is 768x574. (a) the original image taken with the bright field microscope. (b) the binary mask of the original image. (c) the derived contour, which will be used for the rest of the pipeline.

2.2 Image Alignment

The first step of this pipeline (after acquisition) is aligning the contour images. In some cases, the contours are not aligned when stacked on top of each other. This can be observed in the case of the zebrafish model. In figure 4, it can be seen that the center of the contours are in different positions in the black image. This could have been caused during imaging when the samples are placed under the microscope. In that case, the samples are not all placed in the exact same position.



Figure 4: Two contour images from the zebrafish model. The centers of the two contours are different, therefore the contours need to be aligned.

Alignment is done using the OpenCV library [Bra00]. First, the center of each contour is identified. This is done by converting the image to grayscale and then applying a Gaussian blur to reduce the Gaussian noise in the image [Ope22]. Consequently, the image is thresholded after which the contour can be found in the image. For each contour, we can compute the x and y coordinates of the contour center. The next step is shifting the contour, so that the center of the contour is aligned with that of the image. This is done by applying an affine transformation [Ope22] and using a transformation matrix $M = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \end{bmatrix}$. Here, t_x, t_y are the amount of pixels in the x, y direction that the image must be shifted in [Ope22]. The verification of the alignment method can be seen in figure 5. In the left-side image, an overlay of the contours before alignment is shown. The right-hand image is the result of the alignment method, where the center of the contours are aligned.

The image alignment step might be optional if the contours are already properly aligned. This was the case for the mammary gland model, thus for that model this step was omitted.



Figure 5: An overlay of all the zebrafish model contours before (a) and after (b) alignment.

2.3 Point Cloud Extraction

The second step is deriving the point cloud from the stack of contours. A point cloud is a collection of data points in a 3D space. Extracting the point cloud is done by acquiring the x, y-coordinates of the contour, for each aligned 2D contour image. Consequently, a z-coordinate is manually assigned to each contour. For the zebrafish model, the distance between two neighborhood images is 10 pixels in the z-direction. The optical cutting distance in this direction is 5 µm. For the x, y-direction, the pixel to µm ratio is 0.821 µm [CV14]. For the mammary gland model, a distance of 10 was used as well. Using the Open3D library [ZPK18], the coordinates are mapped to a 3D point cloud. This can be seen in figure 6.



Figure 6: Point clouds of the zebrafish (a, b) and mammary gland (c) models.

Some of the 3D reconstruction algorithms used in the next step require an oriented point cloud as input. In order to accomplish this, the normals of the point cloud have to be estimated. The Open3D library is used to estimate the normals. The normals are derived by fitting a plane per 3D point

in a local manner [ZPK18]. Sometimes, the normals are not oriented consistently, thus an extra pre-processing step is necessary. Using k-nearest neighbours, a Riemannian graph is constructed. The orientation of the normals are then propagated using the minimum spanning tree [ZPK18]. To verify that the normals of the point cloud are properly oriented, the normals are observed in MeshLab [CCC⁺08]. For the zebrafish model, the normals seem to be oriented correctly as they are pointing outwards of the model. For the mammary gland model, this is not the case for all of the points. Inconsistent orientation of the normals may influence the 3D reconstruction later on.

2.4 3D Surface Reconstruction

The next step of this pipeline is reconstructing a 3D model from our oriented point cloud. Three reconstruction algorithms are used and compared in order to observe their advantages and disadvantages in generating a 3D model from a biological image stack. The best reconstruction algorithm can then be selected to be part of the pipeline.

2.4.1 Screened Poisson reconstruction

The first method is Screened Poisson surface reconstruction [KH13]. This reconstruction method is an extension of the original Poisson surface reconstruction method [KBH06]. The Poisson method is an implicit surface reconstruction approach that generates a 3D mesh with a watertight surface from an oriented point cloud. It does this by transforming the point cloud to a continuous vector field and then solving the corresponding Poisson problem. This problem involves computing a scalar function with gradients that match the vector field the best. After this the appropriate isosurface can be extracted [KH13, KBH06]. While the Poisson method is robust against noisy data and misregistration artifacts, it tends to oversmooth the data [KH13]. The screened poisson method incorporates scale-independent screening and solves for a screened Poisson equation. It has been proven to out-perform the Poisson method and to be computationally faster [KH13].

2.4.1.1 Parameters

PyMeshLab [MC21], a Python library that interfaces to MeshLab [CCC⁺08], is used to apply the screened poisson surface reconstruction method to the oriented point cloud. The algorithm has various parameters that need to be tuned in order to acquire an optimal model. This was done manually. One of the important parameters is the depth parameter which determines the depth of the octree used for reconstruction. The higher the value of the depth parameter, the more details will be preserved. However computation time will also increase [KH13]. For this research, the default value of 8 was used.

Another parameter is the minimum number of samples. During reconstruction, the octree is adjusted according to the sampling density. Thus, this parameter denotes the minimum amount of points that should be in an octree node during this process. The recommended value for noise-free samples is between the range of 1.0 and 5.0 [CCC⁺08]. After tuning, this value is set to 5.

2.4.2 Alpha Shapes reconstruction

The second algorithm that is used is Alpha Shapes surface reconstruction. Developed by [EM94], alpha shapes are based on the concept of a convex hull. They are a generalized variant of the convex hull of a set of points. Let S be the point set, or point cloud, and α be a real number in the range between 0 and infinity. An α value of infinity generates the convex hull, and decreasing α results in the disassembly of the shape [EM94]. An α value that is too low will leave holes in the model, while a too large value causes details to disappear.

Alpha shapes have the advantage that it is an effective method for generating the shape of the point cloud [BMR+99]. The method can be applied to a dense unorganized set of points [DLY22]. The shape of S generated by the alpha shapes algorithm is a polytope, which is constructed from the Delaunay triangulation of the point cloud [EM94]. A polytope is defined as a shape with flat faces. To gain an intuition of how the alpha shapes algorithm works, we imagine a mass of ice cream in the space of \mathbb{R}^3 , which contains the points of S. We remove the ice cream that can be carved out without bumping into S, using a metaphorical sphere-formed spoon with radius α [DLY22]. This includes the parts we can reach inside of S. The resulting model is the alpha shape of S.

2.4.2.1 Parameters

Alpha Shapes reconstruction is done using PyMeshLab. The algorithm accepts a point cloud and the α parameter as input. The α parameter controls the amount of detail the model has. As was explained earlier, if α is too high details will be lost and if it is too low, there will be holes. Thus, the α parameter has been manually tuned to acquire a 3D model with higher quality. Ultimately, α = 70 (4.771%) was selected for the embryo model and α = 10 (1.430%) for the mammary gland model.

2.4.2.2 Post-processing

Post-processing was performed using PyMeshLab. As alpha shapes constructs a shape with flat faces, we can apply a smoothing algorithm to smooth the surface. Vertex-based smoothing methods are straightforward to implement, although they can inadvertently cause volume reduction. These methods smooth the mesh based on spatial neighbourhood information [BCL21]. Two vertex-based approaches are Humphrey's Classes (HC) Laplacian smoothing [VMM99] and Taubin smoothing [Tau95]. These methods are similar as they both correct shrinkage using forward and backwards smoothing. However, in Taubin smoothing, both of these steps can be fine-tuned [BCL21]. Therefore, the Taubin smoothing function is applied to the entire model.

2.4.3 Ball-Pivoting reconstruction

The last method that is used is Ball Pivoting surface reconstruction. First introduced by [BMR⁺99], the ball pivoting algorithm (BPA) is an advancing-front algorithm that is related to alpha shapes (see section 2.4.2). In contrast to screened poisson, which is an implicit surface reconstruction method, BPA is an explicit surface reconstruction method [Cha21]. Thus, BPA was selected so we can compare its performance to the related alpha shapes method and the implicit screened poisson method. This way, we can observe where their advantages and disadvantages lay in practice. The

algorithm is simple and works the best when applied to uniformly distributed point clouds [Cha21]. BPA uses a ball that pivots on the point cloud. Let S be the point cloud that will be used for reconstruction. We assume that the density of S is sufficient enough that a ball of radius ρ cannot fall through the S, without touching sample points of S. The ρ -ball connects sample points with an edge when the ball touches two sample points. After this, the ball is pivoted until it comes in contact with a third point. This results in the creation of a triangle. The ρ -ball continues to pivot to find seed triangles, which are three points in S that the ball can touch without containing any other data points [BMR⁺99]. This process is repeated and ultimately, an interpolating mesh is formed.

2.4.3.1 Parameters

Ball pivoting reconstruction is done using the Open3D library in Python. The algorithm is based on [BMR⁺99] and [Dig14], the latter of which is a parallel variant of [BMR⁺99]. The algorithm accepts an oriented point cloud and the radii parameter. This parameter determines the radii of the ball. Choosing a good radius can be difficult. A too-low radii value might cause the generation of holes, while a too-large value might result in the loss of details [Dig14]. A solution to this problem was suggested by [BMR⁺99]: We can perform multiple BPA runs, each time with a ball radii larger than the last. The radius parameters have been manually tuned to acquire a 3D model with higher quality. The selected radii are $\rho = [20, 50, 80, 110]$.

2.4.3.2 Post-processing

Due to the nature of the ball pivoting algorithm, it is possible for the resulting 3D model to contain holes. These holes can be caused by a too-low sample density or because the ball cannot reach some of the sample points [BMR⁺99]. Therefore, extra post-processing steps might be necessary. In this research, post-processing is done in PyMeshLab. First the borders of the faces and the edges are selected on the boundary. The close holes function is then used to close any holes that might have been produced. The threshold for the size of the holes to be closed is set to 70. This number was chosen based on a trial-and-error procedure. Lastly, to smooth out the surface and remove any residual noise, the Taubin smooth [Tau95] algorithm is applied to the entire model.

2.5 Evaluation criteria

To verify the performance of each algorithm and compare the quality of the resulting 3D models, three evaluation criteria were determined. Firstly, we perform visual inspection to observe whether certain details are preserved in the reconstructed models. Then, we look at the amount of holes the 3D models have. In the case of the embryo and mammary gland models, holes are inaccuracies we want to prevent. And lastly, the smoothness of the surface of the 3D models will be observed.

3 Results and Evaluation

3.1 Screened Poisson reconstruction

The 3D models constructed by the screened poisson reconstruction algorithm can be found in figure 7. In both cases, the 3D models have a smooth surface, which does not contain any holes. In the case of the zebrafish model, some level of detail has been preserved (see figure 7a, 7b). However, as highlighted in figure 7c, there is an incorrect feature. On the bottom of the model, a small "bump" is visible. This feature cannot be derived clearly from the point cloud, as the points there do not indicate that there should be an inclination.

In the case of the mammary gland model (see figure 7d and 7e), the point cloud was reconstructed incorrectly. Not all of the branches of the gland are visible and it is hard to recognize the structure. Therefore, the level of detail preservation is low.



Figure 7: MeshLab snapshots of the zebrafish embryo (a, b, c) and mammary gland (d, e) models generated by the screened poisson reconstruction method. Models are shown from different angles. While the embryo model contains a high level of detail preservation (a, b), the algorithm also generated a seemingly incorrect feature on the bottom of the model (c). The mammary gland model was not reconstructed correctly.

3.2 Alpha Shapes reconstruction

The 3D models constructed by the alpha shapes surface reconstruction algorithm can be seen in figure 8. For the zebrafish model, there is a severe loss of detail. The round protruding feature (figure 8a) is less clear in terms of shape. A similar conclusion could be drawn about the feature in 8b. Aside from the lost of detail, the model contains a hole on the bottom, which can be seen in figure 8c. Despite the Taubin smoothing step, the surface of the model consists of flat faces and sharp edges.

As for the mammary gland model, the shape of the mammary gland was preserved, as were the branches. The model does contain multiple holes. While the smoothing step did round out some parts of the model, the model is still composed of flat faces and sharp edges.



Figure 8: MeshLab snapshots of the zebrafish embryo (a, b, c) and mammary gland (d, e) models generated by the alpha shapes reconstruction method. Models are shown from different angles. The magenta circles in (a) and (b) indicate details (comparable to those in figure 7a and b) that are hard to see due to the color of the surface. The magenta circle in (c) indicates the hole in the bottom of the model.

3.3 Ball Pivoting reconstruction

The 3D models constructed by the ball pivoting surface reconstruction algorithm are shown in figure 9. Both models contain small holes, even after the post-processing steps as described in section 2.4.3.2. Both models have a smooth surface and the embryo model has a high level of detail preservation. The mammary gland model suffers from the inconsistent normal orientation, as big parts of the surface are incomplete. This resulted in a loss of details and causes the model to be harder to identify. Furthermore, branches of the model that are supposed to be unjoined are connected (figure 9d).



Figure 9: MeshLab snapshots of the zebrafish embryo (a, b, c) and mammary gland (d, e) models generated by the ball pivoting reconstruction method. Models are shown from different angles. Both models contain small holes. While the embryo model exhibits high detail preservation, the mammary gland model is largely incomplete.

4 Discussion

4.1 Comparison

In figure 10, a comparison of the zebrafish embryo models constructed by the three reconstruction methods is shown. Each of the three reconstruction algorithms has its advantages and disadvantages. The screened poisson method is able to produce a watertight surface, meaning a model with a lack of undesirable holes. Aside from also producing a smooth surface, the algorithm is able to preserve a high level of details (see figure 10a). However, the "bump" in the model (figure 10d) appears to be an incorrect feature as it seemingly cannot be derived from the point cloud, nor does it appear in the model generated by the ball pivoting algorithm (figure 10f). The bump smooths out with an increasing number of samples per octree node. At a number of samples equal to 20, the bump is barely visible. However, a high number of samples results in a loss of surface detail.

The ball pivoting algorithm has a model similar to that of the screened poisson algorithm. It preserves the details and has a smooth surface due to the Taubin smoothing that was done. However, it does contain holes that could not be closed even after an extra post-processing step. A characteristic of the ball pivoting algorithm is that it is very sensitive to the selected radius ρ of the ball [Dig14]. Correctly choosing the radius can be difficult. The holes were likely formed because the sampling density was too low or the curvature of the manifold was too large [BMR+99] for the ball.

A naive heuristic of choosing the ball radii was suggested by [Dig14]. The method is based on the size of the bounding box s and the number of points N in the point cloud. It is assumed that the averaged amount of points per range neighborhoods is approximately 20. Therefore, the radius ρ is defined as $\rho = \sqrt{\frac{20}{N}s}$. Although it is a rough estimation, such a method can be tried in the future to possibly generate a generalized set of radii that can be used for multiple types of models.

In comparison to the screened poisson and ball pivoting models, the alpha shapes algorithm performed worse (see figure 10b, e). While screened poisson had a "bump", the alpha shapes model had a hole. Certain features of the model were less detailed and seemed to have a more generalized shape. Lowering the α might bring some of those details back, but could also have the generation of more holes as a consequence.

However, the alpha shapes algorithm did perform the best at reconstruction of the mammary gland model. Figure 11 shows a comparison of the three reconstruction methods and the surface model produced by [CV14]. Likely due to incorrect orientation of some of the point normals, the screened poisson and ball pivoting algorithm are unable to correctly reconstruct the model. While screened poisson does show some of the details of the branches in the big surface it generated, the general shape of the mammary gland is not visible nor easily identifiable. The ball pivoting algorithm has holes and is missing sections of structure. Comparing it to the surface model from [CV14], we can also see that the bottom two branches are connected in the ball pivoting model. A reason for this inaccuracy occurring could be that the radius of the ball was too large for that section of the mammary gland. The ball might have come in contact with sample points of both of the branches, which caused the generation of triangles there.

Research by [Cha21] has shown that, given a uniformly distributed point set, the ball pivoting method is robust. Thus, a possible way to increase the quality of the ball pivoting algorithm is by applying poisson-disk sampling. This was also suggested by [CCC⁺08]. The poisson-disk sampling method generates a uniformly random distribution. Here, the neighborhood distance between the points is 2r. If we imagine a disk with radius r and place the center of the disk on each point, none of the disks will overlap each other [CCS12]. Using this pre-processing step, we could generate a more uniformly distributed point cloud.



Figure 10: MeshLab snapshots of the zebrafish embryo models constructed by the screened poisson (a, d), alpha shapes (b, e) and ball pivoting (c, f) methods. The model by screened poisson adheres the best to the evaluation criteria, though it does contain an apparent inaccurate feature (figure (d), shown in magenta circle).



Figure 11: MeshLab snapshots of the mouse mammary gland models constructed by the screened poisson (a), alpha shapes (b) and ball pivoting (c) methods. (d) shows the surface model generated by [CV14]. The model produced by the alpha shapes method resembles (d) the closest, though it does miss details in the branches.

Lastly, we compare the computation time of the three algorithms for each model. These results can be found in table 1. For both models, the screened poisson reconstruction method is the fastest. For the embryo model, the alpha shapes algorithm is with 150.627 seconds significantly slower than the other two algorithms. For the mammary gland model, the alpha shapes algorithm is faster than the ball pivoting algorithm. The difference in computation time for alpha shapes between the models is likely caused by the amount of points in the point cloud. The embryo model has significantly more points than the mammary gland model. For the ball pivoting algorithm, the difference is likely caused by both the amount of points in the point cloud and the amount of triangles the algorithm was able to form. As shown in figures 10 and 11, the algorithm was able to generate a more complete model for the embryo than for the mammary gland.

Algorithm	Zebrafish embryo runtime (s)	Mammary gland runtime (s)
Screened Poisson	1.825	0.706
Alpha Shapes	150.627	1.010
Ball Pivoting	45.134	14.726

Table 1: Runtime of each reconstruction algorithm in seconds.

4.2 Limitations

The functionality of the image alignment step in the pipeline was verified in figure 5, which shows the alignment of the contours for the zebrafish embryo. This method relies on the notion that the contours need to be aligned based on their center point. However, this might not be the case for all models. In those cases, alignment by searching for common key features in the contour stack might be a more suitable solution.

As could be seen in the 3D mammary gland model, inconsistent orientation of the normals in the point cloud can lead to major inaccuracies in the reconstructed 3D model. The reason for this inconsistency could be that the models are undersampled in the z-direction. This might have caused incorrect calculation of the normals. As two of the three tried algorithms rely heavily on correctly oriented point clouds, this could be a limitation in the case that consistently orienting the normals is not straightforward and needs extra pre-processing steps. This could especially be the case for more complicated models.

5 Conclusion

Images acquired from bioimaging can be converted to a 3D model, which in turn can be used for analysis, measurement and visualization. However, the accuracy of the 3D model can be compromised by noisy data. To this end, an optimization strategy was proposed in this research which aims to optimize the 3D mesh. Based on the pipeline constructed by [CV14], the optimization strategy consisted of multiple steps including image alignment, point cloud extraction and 3D surface reconstruction (see figure 1). The functionality and performance of three surface reconstruction algorithms was analyzed and compared to each other. While the ball pivoting algorithm and the screened poisson method were able to produce a smooth zebrafish embryo model with preserved details, they suffered from the inconsistently oriented point normals in the mouse mammary gland model. In contrast, the alpha shapes models lacked detail but did produce a identifiable mammary gland model. Nevertheless, the screened poisson method did not contain any holes and complied the best to the evaluation criteria. Additionally, it had the fastest computation time. The pipeline could be further built upon to improve on its shortcomings and continue optimization of the resulting 3D mesh. Extra pre- or post-processing steps could be added in order to increase the accuracy of the 3D model. In particular, poisson-disk sampling could be applied to create a point cloud with less noise to increase the quality of models generated by the ball pivoting algorithm. Another important improvement would be to develop a method that ensures consistent orientation of the point cloud normals. In addition, other reconstruction methods from other libraries such as the Computational Geometry Algorithms Library (CGAL) could be tried and analyzed to see if they could be possible candidates for this pipeline.

References

- [BCL21] Antonio Bacciaglia, Alessandro Ceruti, and Alfredo Liverani. Surface smoothing for topological optimized 3d models. *Structural and Multidisciplinary Optimization*, 64, 12 2021.
- [BMR⁺99] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva, and G. Taubin. The ballpivoting algorithm for surface reconstruction. *IEEE Transactions on Visualization* and Computer Graphics, 5(4):349–359, 1999.
- [Bra00] G. Bradski. The OpenCV Library. Dr. Dobb's Journal of Software Tools, 2000.
- [CCC⁺08] Paolo Cignoni, Marco Callieri, Massimiliano Corsini, Matteo Dellepiane, Fabio Ganovelli, and Guido Ranzuglia. MeshLab: an Open-Source Mesh Processing Tool. In Vittorio Scarano, Rosario De Chiara, and Ugo Erra, editors, *Eurographics Italian Chapter Conference*. The Eurographics Association, 2008.
- [CCS12] Massimiliano Corsini, Paolo Cignoni, and Roberto Scopigno. Efficient and flexible sampling with blue noise properties of triangular meshes. *IEEE Transactions on Visualization and Computer Graphics*, 18(6):914–924, 2012.
- [Cha21] Tathagata Chakraborty. Interior ball-pivoting on point clouds for o setting triangular meshes. 2021.
- [CV14] Lu Cao and Fons J. Verbeek. Nature inspired phenotype analysis with 3d model representation optimization. In Ajith Abraham, Pavel Krömer, and Václav Snášel, editors, *Innovations in Bio-inspired Computing and Applications*, pages 165–174, Cham, 2014. Springer International Publishing.
- [DCMS⁺20] Sruti Das Choudhury, Srikanth Maturu, Ashok Samal, Vincent Stoerger, and Tala Awada. Leveraging image analysis to compute 3d plant phenotypes based on voxel-grid plant reconstruction. *Frontiers in Plant Science*, 11, 2020.
- [Dig14] Julie Digne. An Analysis and Implementation of a Parallel Ball Pivoting Algorithm. Image Processing On Line, 4:149–168, 2014. https://doi.org/10.5201/ipol.2014.
 81.
- [DLY22] Tran Kai Frank Da, Sébastien Loriot, and Mariette Yvinec. 3D alpha shapes. In CGAL User and Reference Manual. CGAL Editorial Board, 5.4.1 edition, 2022.
- [EM94] Herbert Edelsbrunner and Ernst P. Mücke. Three-dimensional alpha shapes. ACM Trans. Graph., 13(1):43–72, jan 1994.
- [KBH06] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson Surface Reconstruction. In Alla Sheffer and Konrad Polthier, editors, Symposium on Geometry Processing. The Eurographics Association, 2006.
- [KH13] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. ACM Transactions on Graphics (TOG), 32(3):29, 2013.

- [LIS⁺12] Christian Laforsch, Hannes Imhof, Robert Sigl, Marcus Settles, Martin Heß, and Andreas Wanninger. Applications of Computational 3D-Modeling in Organismal Biology. 03 2012.
- [MC21] Alessandro Muntoni and Paolo Cignoni. PyMeshLab, January 2021.
- [Ope22] OpenCV. The OpenCV Reference Manual, 4.6.0-dev edition, June 2022.
- [PRAS10] Hanchuan Peng, Zongcai Ruan, Deniz Atasoy, and Scott Sternson. Automatic reconstruction of 3d neuron structures using a graph-augmented deformable model. *Bioinformatics (Oxford, England)*, 26:i38–46, 06 2010.
- [Rav16] Jayakumaran Ravi. Spatial reconstruction of biological trees from point cloud. Open Access Theses, 1190, 2016.
- [Tau95] Gabriel Taubin. A signal processing approach to fair surface design, 1995.
- [VdH⁺93] F. J. Verbeek, M. M. de Groot, D. P. Huijsmans, W. H. Lamers, and I. T. Young. 3d base: a geometrical data base system for the analysis and visualisation of 3dshapes obtained from parallel serial sections including three different geometrical representations. *Computerized medical imaging and graphics*, 17(3):151–163, 1993.
- [VMM99] J. Vollmer, Robert Mencl, and Heinrich Müller. Improved laplacian smoothing of noisy surface meshes. *Computer Graphics Forum*, 18, 1999.
- [XLD⁺18] Chi Xiao, Weifu Li, Hao Deng, Xi Chen, Yang Yang, Qiwei Xie, and Hua Han. Effective automated pipeline for 3d reconstruction of synapses based on deep learning. BMC Bioinformatics, 19, 2018.
- [ZPK18] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3D: A modern library for 3D data processing. arXiv:1801.09847, 2018.