

Multi View Registration of C.elegans Embryo

Fatemeh Bagheri

A thesis presented for the degree of
Doctor of Philosophy

Frankfurt Institute For Advanced Studies
Department of Computer Science Goethe University
September 2019

Dissertation for Computer Science Department
Of Johann Wolfgang Goethe University

Advisor: Prof. Dr. Matthias Kaschube

Abstract

Multi-view microscopy techniques are used to increase the resolution along the optical axis for 3D imaging. Without this, the resolution is insufficient to resolve subcellular events. In addition, parts of the images of opaque specimens are often highly degraded or masked. Both problems motivate scientists to record the same specimen from multiple directions. The images, then have to be digitally fused into a single high-quality image. Selective-plane illumination microscopy has proven to be a powerful imaging technique due to its unsurpassed acquisition speed and gentle optical sectioning. However, even in the case of multi view imaging techniques that illuminate and image the sample from multiple directions, light scattering inside tissues often severely impairs image contrast.

Here we show that for c-elegans embryos multi view registration can be achieved based on segmented nuclei. However, segmentation of nuclei in high density distribution like c-elegans embryo is challenging. We propose a method which uses 3D Mexican hat filter for preprocessing and 3D Gaussian curvature for the post-processing step to separate nuclei. We used this method successfully on 3 data sets of c-elegans embryos in 3 different views. The result of segmentation outperforms previous methods. Moreover, we provide a simple GUI for manual correction and adjusting the parameters for different data.

We then proposed a method that combines point and voxel registration for an accurate multi view registration of c-elegans embryo, which does not need any special experimental preparation. We demonstrate the performance of our approach on data acquired from fixed embryos of c-elegans worms. This multi step approach is successfully evaluated by comparison to different methods and also by using synthetic data. The proposed method could overcome the typically low resolution along the optical axis and enable stitching together the different parts of the embryo available through the different views. A tool for running the code and analyzing the results is developed.

Acknowledgments

Countless people supported my effort on this essay. Professors Matthias Kaschube provided invaluable feedback on my analysis and framing, at times responding to emails late at night and early in the morning. Several other people gave helpful advice as I wrote, including Bettina Hein. I want to thank her for helpful discussions, proofreading and Support; Also B.Eppler, D.Bibichkov for discussions .

The analysis presented in this work was performed on computational resources supported by the Frankfurt Institute for Advanced Studies (FIAS). I like to thank IT department for their technical help and support. The data of c-elegans was provided by Buchman Institute for Molecular life science (BMLS) Goethe university, Developmental Cell Biology Group laboratory. I thank their group members for their kind help and support.

Lastly, my family deserves endless gratitude: my father always taught me how to be patient and Diligent, my mother showed me how to be thoughtful and motivated and my brother for teaching me how to fight without fighting. To my family I give everything, including this.

Contents

Contents

1	Introduction	18
2	Segmentation	22
2.1	Data	22
2.2	Preprocessing	23
2.3	Mexican Hat Filter or Laplacian of Gaussian	24
2.4	Binarization	26
2.5	3D Segmentation	27
2.6	2D nuclei separation based on graph cut and convexity–concavity analysis.	33
2.7	2D Concavity-Convexity Analysis	34
2.8	Final result of real 3D images segmentation	35
2.9	Evaluation	35
2.10	Evaluation of Results	36
2.11	Optimized parameter values	39
2.12	Comparison with Lines-of-Sight decomposition method	39
2.13	Summery	40
3	Feature Selection	42
3.1	Local geometry descriptor	42
3.2	Gaussian mixture parameters	42
3.3	Feature Vectors	44
3.4	Producing Feature Vectors	45
3.5	Optimizing feature vector	46
3.6	Summery	47
4	Point Set Registration	49
4.1	Rigid registration	49
4.2	Non-rigid point set registration	55
4.3	Summery	57
5	Multi Step Registration	59
5.1	5.1 Multi-step multi-view registration of c-elegans embryo by LMICP+ICP huber	59
5.2	Evaluation of registration for synthetic data 1	63
5.3	Evaluation of registration for synthetic data 2	66
5.4	Applying transformation to centres before feature selection: Synthetic data 3	69
5.5	Method Evaluation	72
5.6	Evaluation of registration of a different set	75
5.7	Summery	76
6	Fusion	78
6.1	6.1 Fusion and Groupwise registration	78
6.2	6.2 Summery	79

7	GUIs	82
7.1	How to use REG3D GUI.	85
8	Conclusion	88
A	Appendix	91

List of Figures

2.1	The Orthogonal views of the first view at depth $z=80$	23
2.2	The Orthogonal views of the second view at depth $z=80$, a) before alignment, b) after alignment	24
2.3	The Orthogonal views of the first view after preprocessing.	25
2.4	An example of 2d kernel of Laplacian Gaussian in 3D view.	26
2.5	The Orthogonal views of a) the preprocessed image of first view after applying 3D Mexican hat filter, b) kernel of Mexican hat filter with size $1.82*1.82*2.66$ m ($13*13*19$ pixels).	26
2.6	An example of histogram of a 2d image. There are 2 peaks and the red line determines the threshold.	28
2.7	The Orthogonal views of the binary image obtained by 3D stack histogram at depth 80.	28
2.8	Overlap nuclei after the 3D distance watershed transformation.	29
2.9	Visualization of the Gaussian curvature on the test object of Gold feather.	31
2.10	a) Larger principle of Gaussian curvature , b) Smaller principle of Gaussian curvature of 3 overlap nuclei. The Larger principle it has a sharper changes around the border of two Gaussian distributions and could be more useful when the nuclei are densely distributed.	32
2.11	Overlap nuclei after Gaussian curvature separation (a) and after that 3D watershed distance transformation (b).	32
2.12	The orthogonal views of binary image after 3D segmentation.	33
2.13	A connected component R and its convex hull.	34
2.14	D concavity-convexity analysis separation. a)Part of 2D slice of binary image before 2D convex-concavity separation at depth $z=60$. b) The same part after separation. c)contour of result of separation and the original image at the the same depth. d) chosen nuclei from (a). P_1 deepest point of boundary with respect to convex-hull of nuclei (red contour). P_2 closest point to P 1 that doesn't belong to boundary curve of P_1 . 2 nuclei get separated by green line (S3Appendix). The sign of 2D curvature, positive or negative, corresponds to convexities and concavities, respectively, but in both cases, there are also important singularities along the contour when curvature reaches a local maximum or a local minimum. To avoid these singularities the collinearity of the points of the objects should be considered before.	35
2.15	Schematic of analysis pipeline to segment nuclei. Analysis is done on 3D stack image of c-elegans embryo (nuclei channel) by the algorithm. This data goes through the pipeline and the result is 3D segmented stack of embryo. This segmented data can be used to register the embryos in different views.	37
2.16	Enlarged area of 2D slice during separation. a)2D slice of original data at depth $z=60$, b)Outline of binary image at depth $z=60$ up)Initial state of separation procedure, down) Final result of separation procedure. c) overlay of (b) and (c).	38
2.17	3D Evaluation of detected nuclei by the proposed method. Overlay of orthogonal views of original data and ellipsoid fitting at depth= 60 . Yellow, red and green ellipsoids show True Positive, False Negative, False Positive result respectively. The nuclei which are not binarized are the ones that their intensity pixels, due to limitation of z-axis resolution, are too low.	39
2.18	a) A set C is said to be strongly convex with respect to R if any two points of x_1 and x_2 in C, $D R(x_1, x_2)$ is constrained in C. b) A set C is said to be weakly convex with respect to C, if for any two points x_1 and x_2 in C, $D R(x_1, x_2)$ intersects the closure of C at, at least, one point distinct.	40
3.1	Rotation invariant local geometric descriptor. Explanation is included in the text.	43
3.2	All types of conic sections, arranged with increasing eccentricity. Note that curvature decreases with eccentricity, and that none of these curves intersect.	46
3.3	Two ellipsoid with the same first eigenvector and eigenvalue.	46
3.4	12 most similar nuclei in 2 views from the first pair nuclei to 12 th pair nuclei altogether. (a)-(c) the pair nuclei which are selected , are similar in shape (a)- (c) . The feature selection is successful in representing the shape and density of nuclei in c-elegans embryo too (d)-e).	47
4.1	The point cloud data of a bunny left) before , right) after registration by ICP.	51

4.2	(a) 2D Basins of convergence. (b) 3D Example. Top: Initial alignment for 3D registration. Bottom: LM-Huber optimum.	54
4.3	The proposed generative model for joint registration of multiple point clouds (a) and the associated graphical model (b). Unlike pairwise registration strategies, the proposed model simultaneously registers an arbitrary number of point clouds with partial or total overlap and optimally estimates both the GMM and registration parameters. Hence, the solution is not biased towards a particular cloud.	55
4.4	(a) Two given point sets. (b) A coherent velocity field. (c, d) Velocity fields that are less coherent for the given correspondences.	58
5.1	The Overlap of the reference view (red) and the transformed view (blue) result of: a)initial state b)step 1: Registration of centers after feature selection, c)step 2: Rejecting the false correspondences nuclei by Ransac, d)step3: Alignment of embryos after Ransac. (a) and (d) are overlap of embryos. (b) center of nuclei and (c) 12 closest nuclei. 3D views of embryos are volume rendering of segmented nuclei [57].	63
5.2	The Overlap of the reference view (red) and the transformed view (blue) of nuclei . a)after feature selection b)step 1, c)step 2, d)step3. The progressive overlap of nuclei in different step shows the accuracy of registration.	64
5.3	8 features of overlap nuclei after registration for multi-step registration and centre registration. Spatial positions, volume, dot product of first eigenvectors, first eigenvalue and eccentricities are calculated. The log of differences between these values based on the error function are shown above (except than volume and first eigenvalue). Smaller the values are more similar the nuclei are.	65
5.4	8 features of overlap nuclei after registration for multi-step registration and centre registration. Spatial positions, volume, dot product of first eigenvectors, first eigenvalue and eccentricities are calculated. The log of differences between these values based on the error function are shown above (except than volume and first eigenvalue). Smaller the values are more similar the nuclei are.	67
5.5	12 features of overlap nuclei after EM registration: spatial positions, volume, orientations ,principle axes length and eccentricities are calculated. The log of differences between these value are shown above. Smaller the values are more similar the nuclei are.	68
5.6	12 features of overlap nuclei after EM registration: spatial positions, volume, orientations ,principle axes length and eccentricities are calculated. The log of differences between these value are shown above. Smaller the values are more similar the nuclei are.	69
5.7	The Overlap of the reference view and the transformed reference view nuclei centres a)before registration, b)after registration.	70
5.8	8 features of overlap nuclei after and before registration. Spatial positions, volume, dot product of first eigenvectors, first eigenvalue and eccentricities are calculated. The log of differences between these values based on the error function are shown above (except than volume and first eigenvalue). Smaller the values are more similar the nuclei are.	71
5.9	The Overlap of the reference view and the transformed reference view. A)before registration, B)after multi-step registration , C)after centre registration. All nuclei are matched except than 20 ± 1 nuclei which were added/removed. (D)-(E) 2D contour of the reference view nuclei (purple) and transformed reference view nuclei (green). The overlap of nuclei is shown by white. The arrows show the same nuclei with and without jitters.	72
5.10	The Overlap of the reference view and the transformed reference view nuclei centres a)before registration, b) after registration.	73
5.11	3D view of overlap of the reference view and the registered transformed reference view nuclei..	73
5.12	8 features of overlap nuclei after registration for multi-step registration and centre registration. Spatial positions, volume, dot product of first eigenvectors, first eigenvalue and eccentricities are calculated. The log of differences between these values based on the error function are shown above (except than volume and first eigenvalue). Smaller the values are more similar the nuclei are.	74
5.13	The Overlap of the reference view and the transformed reference view nuclei centres a)before registration, b) after registration.	76
5.14	3D view of overlap of the reference view and the registered transformed reference view nuclei.	76
5.15	8 features of overlap nuclei after registration for multi-step registration and centre registration (with out feature selection). Spatial positions, volume, dot product of first eigenvectors, first eigenvalue and eccentricities are calculated. The log of differences between these values based on the error function are shown above (except than volume and first eigenvalue). Smaller the values are more similar the nuclei are.	77
5.16	8 features of overlap nuclei after registration for multi-step registration and centre registration (with out feature selection). Spatial positions, volume, dot product of first eigenvectors, first eigenvalue and eccentricities are calculated. The log of differences between these values based on the error function are shown above (except than volume and first eigenvalue). Smaller the values are more similar the nuclei are.	79
6.1	The Overlap of the reference view and the fused view of reference view and the second view. First row 2D outline of reference view and the fused view. Pink colour shows fused view and green colour reference view. Common	

nuclei are shown by white. b) Green colour shows the second view. c) blue colour shows the third view. Second row: 3D view of overlap stack images. d) result of fusion step. e) result of groupwise registration step.	82
7.1 CS3D: Graphical user interface (GUI).The GUI includes different tabs for different task. The image should be first imported. After opening the raw image preprocessing parameters should be adjusted. The next step is separate the nuclei 3D and 2D. To correct the segmented image the radio button manual should be chosen and the segmented nuclei could be either separated or connected..	85
7.2 CS3D: Graphical user interface (GUI).The GUI includes different tabs for differer task. After importing the image and preprocessing step segmentation part will be started by clicking segment push button. Th procedure will not start in case the segmented image already exists or parameters of this section is not valid.	86
7.3 CS3D: Graphical user interface (GUI).The GUI includes different tabs for different task. After segmentation the user can correct the segmentation manually. (CS3D does that automatically too.) Correction type should be set to manual and then either Add Edge or Remove Edge push-buttons should be chosen. The user can modify the nuclei by drawing an edge or erasing an edge.	87
7.4 REG3D: Graphical user interface (GUI). The GUI includes different tabs for different task.	89
A.1 Cumulative distribution of histogram of A,B)Major and minor diameters of nuclei in XY and XZ planes and C)ratio of major to minor diameters of nuclei in XY plane. D)Distance between nuclei of Dataset 1. E,F) The same procedure of (A) and (B) but for XZ plane. The value of XZ plane are just calculated for 20 slices while the XY plane for 201 slices. The measurements are calculated from manual ellipse fitting of original data set 1 after Binarization.	95
A.2 Cumulative distribution of histogram of A,B)Major and minor diameters of nuclei in XY and XZ planes and C)ratio of major to minor diameters of nuclei in XY plane. D)Distance between nuclei of Dataset 1. E,F) The same procedure of (A) and (B) but for XZ plane. The value of XZ plane are just calculated for 20 slices while the XY plane for 201 slices. The measurements are calculated from manual ellipse fitting of original data set 2 after Binarization.	96
A.3 3D Gaussian intensity distribution of nuclei. A)Surf plot of larger principle of Gaussian curvature, B)Smaller principle of Gaussian curvature of a Gaussian mixture distribution. C)2D view of the larger principle of Gaussian curvature. D)2D view of the smaller principle of Gaussian curvature. The distribution used is $z = 3 \cdot (1 - x)^2 \cdot \exp((x - 2) \cdot (y + 1)^2) - 10 \cdot (x/5 - x^3 - y^5) \cdot \exp(x^2 \cdot y^2 - 1/3 \cdot \exp((x + 1)^2 \cdot y^2))$. With x and y being a 49*49 2D matrix. By using the positive value of larger principle the 3 Gaussians could be separated.	97
A.4 Registration of centres. The Overlap of the reference view (red) and the transformed reference view (blue) nuclei centres a)before registration, b)after registration. c) Enlarged area of ROI chosen in (b).	98

List of Tables

2.1	Evaluation of different segmentation methods: Mean values between 3 data sets are calculated..	37
5.1	Mean of euclidean distance between all nuclei centres of embryos at each step for Multi-step registration.	64
5.2	Quantification of result of registration.	65
5.3	Mean of Log difference of features for synthetic data2 based on centres and multi step registration plotted in Figure 5.3.	66
5.4	Mean of euclidean distance between all nuclei centres of embryos at each step for 2 methods..	66
5.5	Mean of Log difference of features for synthetic data2 based on centres and multi step registration plotted in Figure 5.8.	71
5.6	Mean of euclidean distance between all nuclei centres of embryos at each step for 2 methods.	72
5.7	Mean of Log difference of features for synthetic data2 based on centres and multi step registration plotted in Figure 5.12.	75
5.8	Mean of euclidean distance between all nuclei centres of embryos at each step for 2 methods.	75
5.9	Mean of Log difference of features for synthetic data2 based on centres and multi step registration plotted in Figure 5.15.	78
5.10	Mean of euclidean distance between all nuclei centres of embryos at each step for 2 methods.	78
A.1	S1: Names and parameter values of filters used in the proposed method.	94

Chapter 1

Introduction

In confocal microscopy, resolution along z-axis(axial resolution) is less than resolution in x-y plane due to light diffraction[1]. It makes the analysis of objects smaller than the axial resolution limit harder. Multi-view microscopy techniques, such as tilted confocal acquisitions, SPIM, and Ultramicroscopy (2 views) [2], have been developed to increase this resolution. Software-based methods of image registration and reconstruction based on the image intensities typically require a significant overlap of image content which is often difficult to achieve particularly in live imaging of dynamically changing samples. Moreover, these methods are computationally demanding. The idea to incorporate fiducial markers to facilitate sample independent reconstruction is widely used in medical imaging and electron tomography. Due to the low amount of fiducial markers available for registration, research has focused on error analysis rather than the efficiency of matching of thousands of markers with only partial overlap [3]. The use of local descriptors instead of complete scenes for matching is proposed in many fields comprising image registration, robotics and autonomous systems, and computer vision.

Segmentation based methods were suggested for the reconstruction of multi-view microscopy data. The centers of segmented objects are used as references. However, the problem with the centers is that they are not adequate to have an accurate registration [4]. Centers of objects can neither represent the spatial information about the objects nor sufficient information of the position of them relative to their neighbors.

Another approach is to combine the idea of using trustee markers (or beads), local descriptors, and geometric hashing and apply global optimization. This can register any or a random number of partially overlapping point clouds. The efficiency of this method [5,6] depends on the number of incorporated beads, bead distribution, amount of overlap. However, the problem with this method is that it cannot be used for small samples like *C.elegans* embryo during morphogenesis. The beads are big and using a sufficient number of them is not compatible with the sample preparation. It is not either compatible with limitation of the field of view of the confocal microscopy.

To be able to use the nuclei as markers and use their features, segmentation of nuclei is needed. Segmentation of *C.elegans* nuclei is challenging since they are densely distributed, elongated, and have irregular shapes. In previous works, in this area, Santella et al. [7] segmented 2D slices and then modeled the 3D segmentation of nuclei based on a trained probabilistic model of nuclei shape. Segmenting of 2D slices, labeling the nuclei, using the Gaussian mixture, and KNN to obtain 3D nuclei is another attempt to segment 3D nuclei of the embryos [8]. 2D separation of nuclei to correct the detection of objects has been done before [9,10,11,12]. The focus there is to calculate the concavity of objects. This parameter could be used to remove the saddle area and split the nuclei. There have been such attempts for 3D segmentation of nuclei [13,14,15,16]. Line of sight (LOS) is an approach that has been used to separate overlap nuclei, recently (2015) [17]. This method is based on the computation of inner visibility between points on the surface of the shape. Two points on the surface of a shape are mutually visible and are said to be in a line-of-sight (LOS) if the straight line segment between them does not leave the inner volume of the shape. Two surface points or any clique of points that are in LOS are also said to be in a convex position. This method does not need to calculate the exact value of curvature and has been able to separate the nuclei of the early mouse embryo and two different cellular spheroids successfully. Toyoshima et al. [13] have developed a method that has been shown to be more accurate than the common segmentation tools such as Ilastik [18], FARSight [19] for an adult *C.elegans*. This method has been used successfully for the segmentation part of tracking neurons in a moving and deforming brain in *C.elegans* [20].

For *C.elegans* embryo nuclei are so dense that none of the above methods are able to segment and separate them accurately. Thus we propose a method to segment the nuclei in 3D for *C.elegans* we proposed a method. The pipeline of the proposed method is as follows: preprocessing, 3D Mexican hat filter, segmentation, separation. In the preprocessing step, the contrast of the 3D image is enhanced and low-level noise is removed. Here a Mexican hat filter was used

that resembles the shape and size of nuclei. This feature of the Mexican hat filter helps to enhance the shape of nuclei while losing the effect of pixel intensity values between nuclei [21]. In the segmentation step, the background will be separated from nuclei by thresholding the image stack using the otsu method [22] implemented in Fiji. 3D distance watershed transformation is implemented in a way to avoid over or undersegmentation. Too small objects are regarded as background noise and are removed. Separation happens by using 3D Gaussian curvature separation and 2D concavity-convexity analysis. The curvature of the iso-intensity surface at each voxel is calculated [22]. The maximum of principle curvature is chosen and based on the algorithm proposed in chapter 2 section 2.5.2, saddle-shaped surfaces voxels will be deleted. To make the separation faster and more accurate, 2D Concavity-Convexity analysis is implemented in Matlab [12]. 2D outline of nuclei with size bigger than 40 pixels is chosen and the convex hull of each is calculated. Based on the outline and convex hull, the nuclei are separated. The proposed method is explained in chapter 2 in more details.

In computer vision applications, global features describe the image as a whole. Global features include contour representations, shape descriptors, and texture features. Shape Matrices, Invariant Moments (Hu, Zerinke), Histogram Oriented Gradients (HOG), and Co-HOG are some examples of global descriptors [23]. Local features describe the image patches (key points in the image) of an object. SIFT, SURF, LBP, BRISK, MSER, and FREAK are some examples of local descriptors. Generally, for low-level applications such as object detection and classification, global features are used and for higher-level applications such as object recognition, local features are used [24].

Another term that was introduced in feature selection is geometric hashing that was first used in object recognition [21] but later was applied to different problems such as structural alignment of proteins [22]. To have the combination of the feature of global features, local features, and geometric hashing was proposed. For feature extraction, segmented nuclei are used (chapter 3). At the next step, a feature vector is obtained and the coefficient of features are optimized. By using the distance between these vectors and KNN algorithm the nuclei with the close feature values are chosen.

To solve the registration problem, practical point set registration algorithms (chapter 4) should be able to model the transformation required to align the point sets accurately and robustly while handling high dimensionality of the point sets very well. The transformation obtained is either rigid or non-rigid. The simplest non-rigid transformation is affine, which allows anisotropic scaling and skews. This transformation could be seen in many real-world problems such as deformable motion tracking, shape recognition and medical image registration [25,26].

The Iterative Closest Point (ICP) is the most well-known algorithm for registering two 2D or 3D point-sets under Euclidean (rigid) transformation (introduced in the early 1990s). This algorithm starts with an initial transformation (rotation and translation), then alternates between building closest-point correspondences under the current transformation and estimating the transformation with these correspondences, until convergence. Since this algorithm and its variants have conceptual simplicity, high usability and good performance in practice, they are very popular and have been successfully applied in numerous real-world tasks [26].

Nevertheless, the initial value of the iteration should be determined before the first step of the ICP algorithm is performed otherwise the algorithm may lead to a local optimum, and the iteration cannot converge to the correct registration result [27,28]. To overcome this problem feature selection was applied to have an accurate solution for ICP. To make the ICP method more robust, either Trimming [29] or Ransac [30] are often used to remove the outliers and finds an affine transformation based on the closest correspondences points.

The Levenberg-Marquardt algorithm is one of the variants of the ICP method, which replaces the Euclidean distance with the Chamfer distance and uses a Levenberg-Marquardt algorithm to compute the transformation T_k , which accelerates the convergence of data registration while keeping high accuracy. In the case of high overlapping ratios, the LM-ICP method is superior to the standard ICP method [31]. While there are many methods developed for point registration most of them select one of the sets as the model and perform pairwise alignments between the other sets and this set. This brings in noise and outliers to model estimation. The methods based on Expectation Maximization treats all the point sets equally as they are realizations of a Gaussian mixture (GMM) and the registration is cast into a clustering problem [32].

To register multi-view images of C.elegans the registration method should be robust and efficient for voxel registration. Among the suggested methods combination of LMICP and ICP Huber method (chapter 4) could lead us to this goal. The Multi-step registration we propose is a combination of point and voxel set registration. After feature selection, the control points obtained from centers of nuclei are registered. To have an efficient and robust registration Ransac algorithm is used to obtain 12 points that could give us the transformation matrix. To be able to keep the spatial information, nuclei are used instead of centers at this step. The 12 nuclei work as anker for embryo so at the next step the embryo will be aligned with respect to that.

The proposed method is evaluated with different methods and different datasets. In addition to LMICP and ICP which were already introduced, Expectation-Maximization (EM), Coherent Point Drift (CPD) and RPM methods were used to compare the result of the proposed method.

Finally, a machine learning-based approach is developed to fuse the registered image (chapter 6). At decision-level fusion, merging non-overlap nuclei by means of KNN learning algorithm; and at feature-level fusion, concatenating nuclei of both sets by means of geometry features will be used.

To use this registration for a group of images, all of them will be registered with respect to the reference view. In our case, we have 6 views of the embryo that to be registered.

GUIs (Graphical user interface) developed for these two methods are explained in chapter 7.

Chapter 2

Segmentation

Image segmentation is the process of dividing a digital image into multiple segments (sets of pixels, contours, or area). The goal of segmentation is to modify or change the representation of an image into something that is more meaningful and easier to interpret [34]. The common techniques used for image segmentation are: thresholding method, edge detection based techniques, region-based techniques, clustering-based techniques, watershed-based techniques, partial differential equation-based, and artificial neural network-based techniques, etc [35]. How to segment correctly is always challenging. Here we used a combination method for preprocessing (Mexican hat filter, Otsu binarization) and post-processing (3D Gaussian curvature separation, 2D separation based on graph cut and convexity-concavity analysis). The result of segmentation is compared with other methods and it showed a drastic improvement in the performance of our method.

2.1 Data

The raw datasets consisted of 3 z-stacks of high-resolution 2D images taken from a C.elegans embryo in different views. The images have the 512*512 pixels in width and 201 slices in z with the voxel size of $0.14*0.14*0.21 \mu m^3$ respectively. Figure 2.1 shows the Orthogonal views of the first view at depth z=80.

At depth 80-100 the two original images have high correlation calculated for pixel values and at the depth 80, the highest correlation happens. The first view could be chosen as the reference image or fixed image. The other views should be aligned manually based on the angle of imaging in the experiment to be able to start registering the image. For the second view the stack first turns down upside (x-mirror) and then rotates clockwise 90° (Figure 2.2).

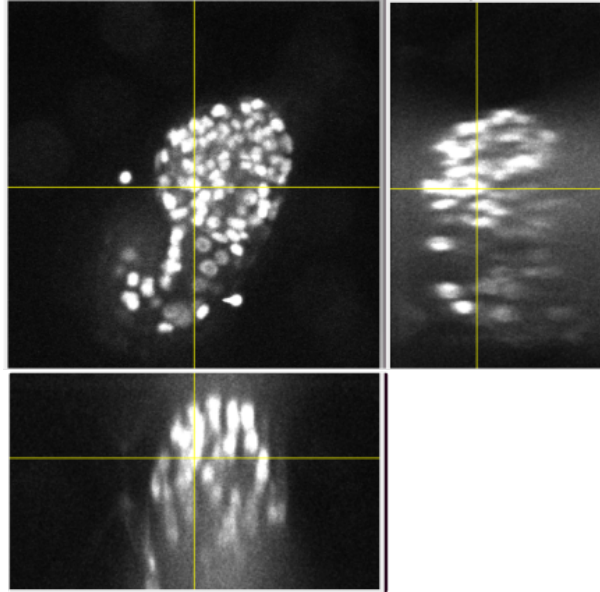


Figure 2.1. The Orthogonal views of the first view at depth $z=80$ with YZ views and XZ views. The Orthogonal views of data displays the XZ and YZ planes at a given point in the 3D image. The intersection of the yellow lines on the stack indicating the point in the stack that is being analyzed.

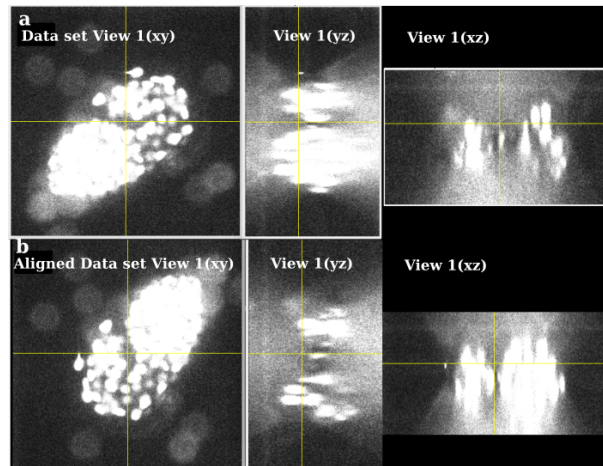


Figure 2.2. The Orthogonal views of the second stack at depth $z=80$, a) before alignment, b) after alignment with YZ views and XZ views.

2.2 Preprocessing

To enhance contrast and remove low-level noise of 3D image 3 filters were used. The result is shown in Figure 2.3.

3D Gaussian Blur

A Gaussian blur filter (also known as Gaussian smoothing) blurred an image by a Gaussian function. It is typically used to reduce noise and smooth the image. The size of the filter is $0.24*0.24*0.24 \mu m$.

3D Maximum

This filter does grayscale dilation by replacing each pixel in the image with the largest pixel value in that pixel's neighborhood. The size of the filter is $0.24*0.24*0.24 \mu m$.

Sharpen

Increases contrast and accentuate the detail in the image or selection. This filter is used to decrease the effect of the 3D Gaussian blur filter used before. This filter uses the following weighting factors to replace each pixel with a weighted average of the 3x3 neighborhood.

$$M = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 12 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

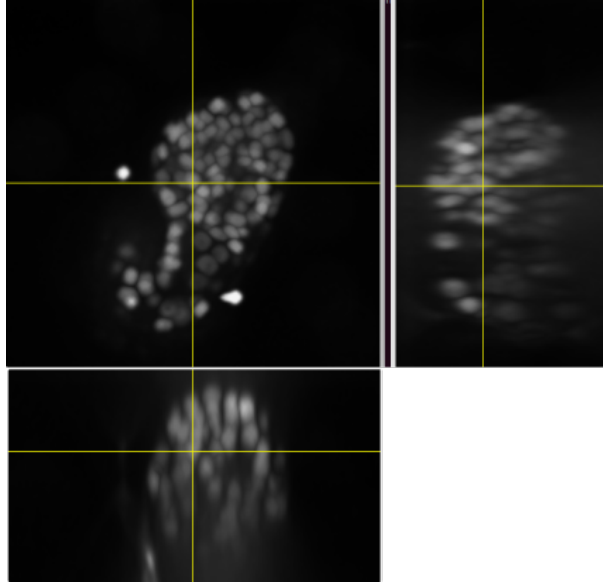


Figure 2.3. The Orthogonal views of the first view after preprocessing.

2.3 Mexican Hat Filter or Laplacian of Gaussian

In imaging science, the difference of Gaussian is a feature enhancement algorithm that involves the subtraction of one blurred version of an original image from another, less blurred version of the original. In the simple case of grayscale images, the blurred images are obtained by convolving the original grayscale images with Gaussian kernels having differing standard deviations.

The algorithm can also be used to obtain an approximation of the Laplacian of Gaussian (LoG) when the size ratio of the kernel (2) to the kernel (1) is roughly equal to 1.6 [37,38]. The Laplacian of Gaussian is useful for detecting edges that appear at various image scales or degrees of image focus. The exact values of sizes of the two kernels that are used to approximate the Laplacian of Gaussian will determine the scale of the difference image, which may appear blurry as a result.

Because of the separability and cascability of Gaussians of DoG filter, efficient implementation of the LoG filter could be obtained by DoG.

The laplacian of Gaussian for 3D Image $I(x,y,z)$ is approximated:

$$\nabla^2 I \simeq DoG * I = (G_{\sigma_1} - G_{\sigma_2}) * I = \left(\frac{1}{\sigma_1 \sqrt{2\pi}} e^{-\frac{(x^2+y^2+z^2)}{2\sigma_1^2}} - \frac{1}{\sigma_2 \sqrt{2\pi}} e^{-\frac{(x^2+y^2+z^2)}{2\sigma_2^2}} \right) * I \quad (2.3.0.1)$$

Which represents an image convoluted to the difference of two Gaussian, that could also be approximated as Mexican Hat Filer (Figure 2.4).

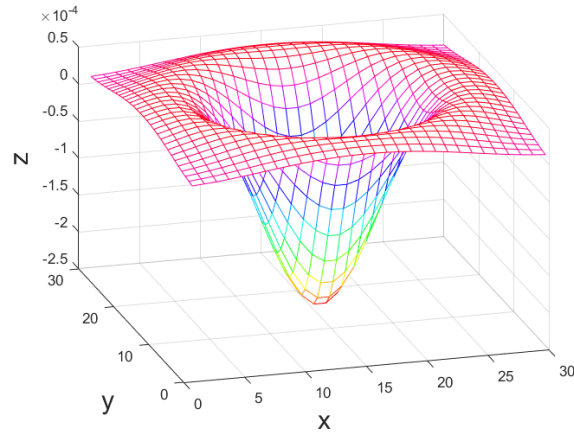


Figure 2.4. An example of 2D kernel of Laplacian Gaussian in 3D view. Figure reproduced from Ref [75].

Fast implementation

For 3D data, it is very advantageous to have a separable implementation to make the computation time acceptable. This method expresses the LoG detector as a sum of 3 separable filters. This new separable implementation of the 3D LoG filter speeds up computation time dramatically. The following derivation is reproduced from Ref [38].

$$LoG = g(x, y, z) = C \left(\frac{x^2}{\sigma_1^4} - \frac{1}{\sigma_x^2} + \frac{y^2}{\sigma_y^2} - \frac{1}{\sigma_y^2} + \frac{z^2}{\sigma_z^4} - \frac{1}{\sigma_z^2} \right) e^{-\frac{x^2}{\sigma_1^4} - \frac{y^2}{\sigma_y^2} - \frac{z^2}{\sigma_z^4}} \quad (2.3.0.2)$$

The result of the convolution of Mexican hat kernel with a 3D image is shown in Figure 2.5.

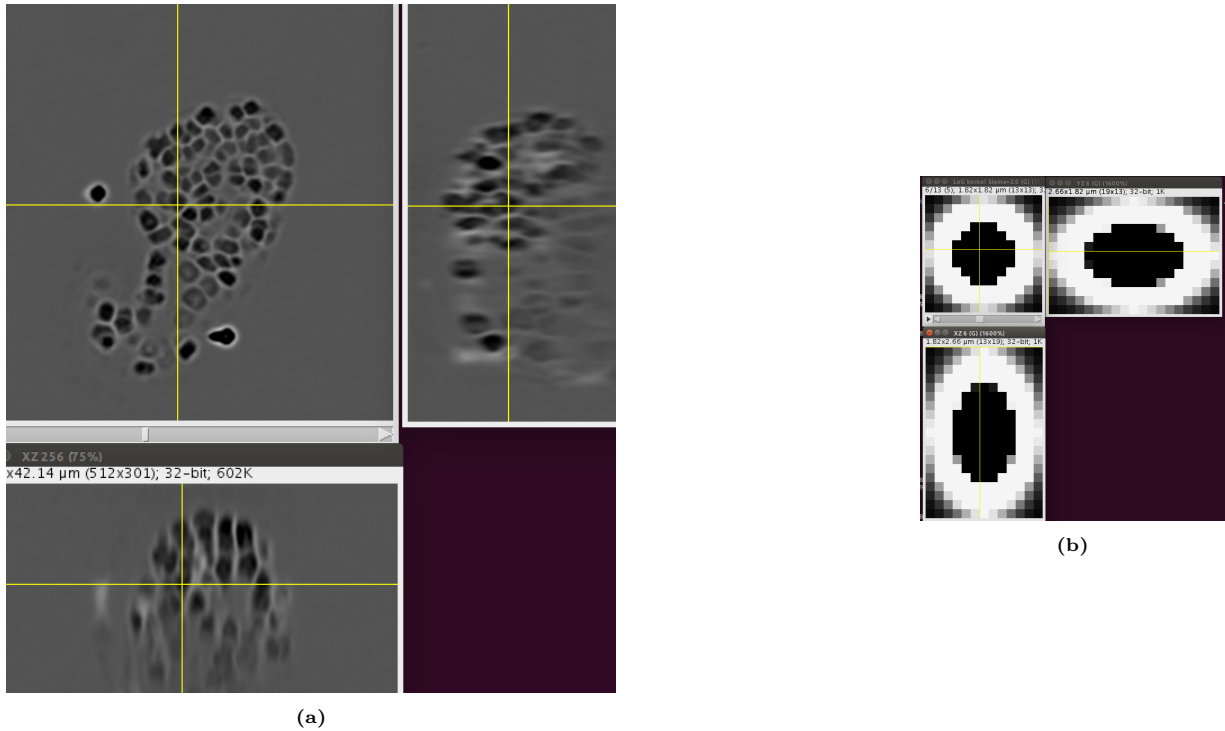


Figure 2.5. The Orthogonal views of a) the preprocessed image of first view after applying a 3D Mexican hat filter, b) kernel of Mexican hat filter with size $1.82 \times 1.82 \times 2.66 \mu\text{m}$ ($13 \times 13 \times 19$ pixels).

2.4 Binarization

Binary images are images whose pixels have only two possible intensity values. They are normally displayed as black and white. Numerically, the two values are often 0 for black, and either 1 or 255 for white.

Research had explored a variety of ways to choose a threshold objectively by examining the histogram of image pixel values. The basic idea is to look for two peaks, representing foreground and background pixel values, and pick a point in between the two peaks as the threshold value (Figure 2.6).

To see how threshold value is chosen, it is explained that if the pixels can take on the set of values $i=1,2,\dots,L$. Then histogram count for pixel value i is n_i and the associated probability is $p_i = n_i/N$, where N is the number of image pixels.

The thresholding task is formulated as the problem of dividing image pixels into two classes. C_0 is the set of pixels with values $[1,\dots,k]$, and C_1 is the set of pixels with values in the range $[k+1,,L]$. The overall class probabilities, ω_0 and ω_1 , are: (The following derivation is reproduced from Ref [39].)

$$\omega_0 = \sum_{i=1}^k p_i = \omega(k) \quad (2.4.0.1)$$

$$\omega_1 = \sum_{i=k+1}^L p_i = 1 - \omega_0(k) \quad (2.4.0.2)$$

The class means, μ_0 and μ_1 , are the mean values of the pixels in C_0 and C_1 . They are given by:

$$\mu_0 = \sum_{i=1}^k \frac{ip_i}{\omega_0} = \frac{\mu(k)}{\omega(k)} \quad (2.4.0.3)$$

$$\mu_1 = \sum_{i=k+1}^L \frac{ip_i}{\omega_1} = \frac{\mu_T - \mu(k)}{1 - \omega(k)} \quad (2.4.0.4)$$

where

$$\mu(k) = \sum_{i=1}^k ip_i$$

and μ_T , the mean pixel value for the total image, is:

$$\mu_T = \sum_{i=1}^L ip_i$$

is the total mean level of the original picture. We can easily verify the following relation for any choice of k :

$$\omega_0\mu_0 + \omega_1\mu_1 = \mu_T \quad \omega_0 + \omega_1 = 1 \quad (2.4.0.5)$$

The class variances, σ_0^2 and σ_1^2 , are:

$$\sigma_0^2 = \sum_{i=1}^k \frac{(i - \mu_0)^2 p_i}{\omega_0} \quad (2.4.0.6)$$

$$\sigma_1^2 = \sum_{i=k+1}^L \frac{(i - \mu_1)^2 p_i}{\omega_1}. \quad (2.4.0.7)$$

There are three measures of "good" class separability: λ , F , η . These are given by:

$$\lambda = \frac{\sigma_B^2}{\sigma_\omega^2} \quad (2.4.0.8)$$

$$F = \frac{\sigma_T^2}{\sigma_\omega^2} \quad (2.4.0.9)$$

$$\eta = \frac{\sigma_B^2}{\sigma_T^2} \quad (2.4.0.10)$$

where

$$\sigma_\omega^2 = \omega_0\sigma_0^2 + \omega_1\sigma_1^2$$

$$\sigma_B^2 = \omega_0(\mu_0 - \mu_T)^2 + \omega_1(\mu_1 - \mu_T)^2 = \omega_0\omega_1(\mu_1 - \mu_0)^2$$

$$\sigma_T^2 = \sum_{i=1}^L (i - \mu_T)^2 p_i.$$

are within-class variance, between-class variance, and total variance.

Consider an example in which you have a part of an x-ray image which composed of soft tissue and bone and you want to separate them. We know bones generally have a higher intensity than soft tissue in x-ray image but some part of bones are very thin and some part of soft tissue is very thick so some pixels which belong to the bone have lower intensity than pixels belong to soft tissue. In this problem, any threshold that we choose leads to some miss labeled pixels.

let's consider more details. Bones have thick and thin areas that lead to a variance around the average intensity of bones, this variance is called within-class variance and the average value represents the bone intensity value. The same is true for soft tissue. The difference between the average of soft tissue and the average of bone is called between-class variance.

The optimal threshold k^* that maximizes η , or equivalently maximizes σ_B^2 is selected in the following sequential search by using the simple cumulative quantities $\omega(k)$ and $\eta(k)$, or explicitly using $[\omega_0\omega_1\eta_0\eta_1]$:

$$\eta(k) = \frac{\sigma_B^2(k)}{\sigma_T^2} \quad (2.4.0.11)$$

$$\sigma_B^2 = \frac{[\mu_T\omega(k) - \mu(k)]^2}{\omega(k)[1 - \omega(k)]}. \quad (2.4.0.12)$$

The equations above are the heart of the algorithm. σ_B^2 is computed for all possible threshold values (k) and the threshold is the value that maximizes it.

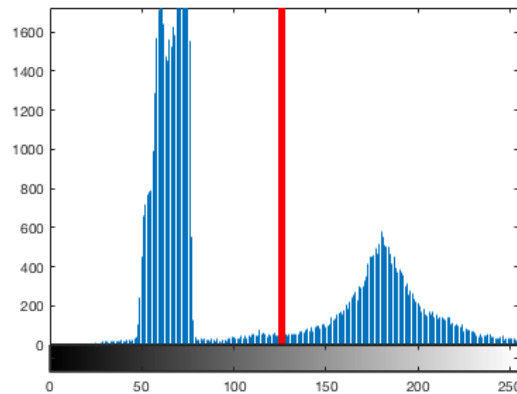


Figure 2.6. An example of histogram of a 2D image. There are 2 peaks and the red line determines the threshold. Figure reproduced from Ref [74].

In different implementations of this algorithm, we are looking to use either η or λ and k . In the case of C.elegans data, Otsu's threshold clustering algorithm searches for the threshold that minimizes the intra-class variance, defined as a weighted sum of variances of the two classes. There have been some other approaches like using the maximum between-class variance for automatic gridding of cDNA microarray images.

For C.elegans embryo data, at the first attempt, a 3D stack histogram is used to find the threshold and make a binary image. In this case less but more accurate number of nuclei will be available to use for registration.

For a 3D image, either a 3D stack histogram is chosen or a 2D stack histogram. To have the most available nuclei we choose the 2D stack histogram of C.elegans embryo data.

2.5 3D Segmentation

In this step to segment binary image 3D watershed algorithm was used, the 3D watershed algorithm is not able to segment overlap nuclei (Figure 2.8). To be able to separate nuclei Gaussian curvature separation is used to be explained

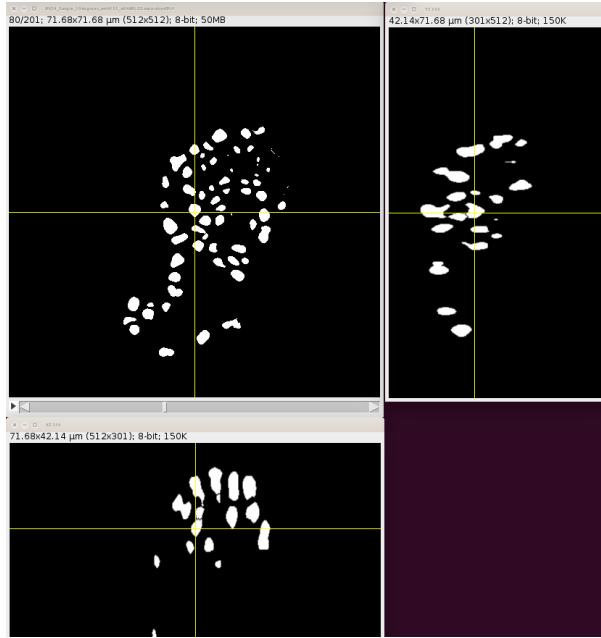


Figure 2.7. The Orthogonal views of the binary image obtained by 3D stack histogram at depth 80.

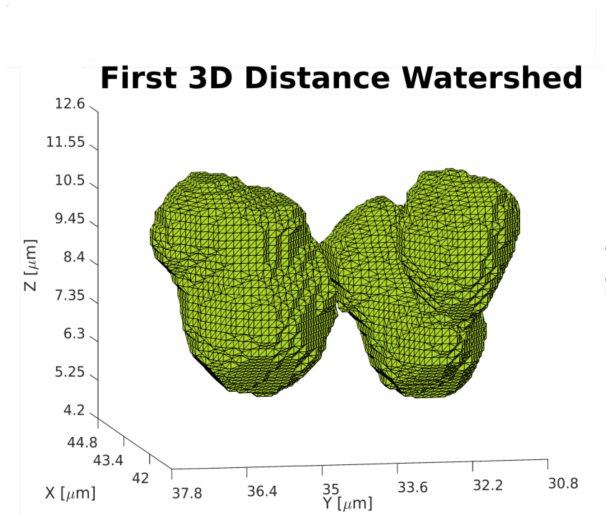


Figure 2.8. Overlap nuclei after the 3D distance watershed transformation.

in the following.

Gaussian curvature

A regular surface S in three-dimensional Euclidean space defined by a mapping $X : R^2 \rightarrow R^3$ is considered. Thus, X maps any point $(u, v) \in S$ to a point $X(u, v) \in R^3$. By $p = (u, v) \in S$ some point on that surface which corresponds to a point $(x, y, z) = X(u, v) \in R^3$ will be denoted. Furthermore, X_u and X_v for the partial derivatives of the surface in directions u and v will be written, respectively, and notations such as X_{uv} for second partial derivatives will be used, analogously. (The following derivation is reproduced from Ref [22].)

First fundamental form

The first fundamental form enables local measurements on a surface without referring to the surrounding space R^3 . For example, the length of a curve on a surface, the angle between two curves on a surface, and the surface area of a region on a surface can be determined.

Definition 1

Let S be a regular surface. Thus, a tangent plane $T_p(S)$ exists in every point $p \in S$. The first fundamental form $I_p : T_p(S) \rightarrow R$ is given by

$$I_p(w) = \langle w, w \rangle_p \text{ with } w \in T_p(S). \quad (2.5.0.1)$$

The mapping I_p assigns the restriction of the scalar product $\langle \cdot, \cdot \rangle_p : R^3 * R^3 \in R$ on the tangent plane in S to each point $p \in S$.

The components $g : R * R \in R$ of the first fundamental form g are defined as follows:

$$\begin{aligned} g_{11}(u_0, v_0) &:= \langle X_u, X_u \rangle_p, & g_{12}(u_0, v_0) &:= \langle X_u, X_v \rangle_p, \\ g_{21}(u_0, v_0) &:= \langle X_v, X_u \rangle_p, & g_{22}(u_0, v_0) &:= \langle X_v, X_v \rangle_p. \end{aligned}$$

$$g = \begin{pmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{pmatrix}, \quad (2.5.0.2)$$

Second fundamental form

To make statements about curvature, which belongs to the properties of the outer geometry, the second fundamental form is considered.

Definition 2

Let S be a regular surface. Thus, a tangent plane $T_p(S)$ exists in every point $p \in S$. Let N_p denote the normal vector of $T_p(S)$ in point p . The second fundamental form is then given by the bilinear form I_p , that is defined on $T_p(S)$ through

$$I_p(w) = \langle dN_p(w), w \rangle \text{ with } w \in T_p(S). \quad (2.5.0.3)$$

The components $h : R * R \rightarrow R$ of the second fundamental form h are defined as follows:

$$\begin{aligned} h_{11}(u_0, v_0) &:= \langle X_{uu}, N \rangle_p, & h_{12}(u_0, v_0) &:= \langle X_{uv}, N \rangle_p, \\ h_{21}(u_0, v_0) &:= \langle X_{vu}, N \rangle_p, & h_{22}(u_0, v_0) &:= \langle X_{vv}, N \rangle_p. \end{aligned}$$

The second fundamental form can be represented as a matrix h as well. This matrix is also symmetric since $h_{12} = h_{21}$,

$$h = \begin{pmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{pmatrix}, \quad (2.5.0.4)$$

The first and second fundamental forms allow us to compute the Gaussian curvature K by calculating the determinant h and g . To be able to do that, first, we need to define A as a matrix known as the Shape Operator or Weingarten operator. The components of A can be obtained from the first and second fundamental forms.

$$A = -hg^{-1} \quad (2.5.0.5)$$

The eigenvalues of A are the principal curvatures, whose product is the Gaussian curvature and the average gives the Mean curvature.

Curvature of Implicit Surfaces Our goal is to segment a given 3D intensity image. This image can be interpreted as the level function $t = f(x, y, z)$, where t is the intensity value at the point (x, y, z) . Therefore the image can be thought of as an embedded surface in R^3 , with the parametrization $X = (u, v, f(u, v))$ by letting $u = x, v = y$ [40]. The first order partial derivatives are the vectors $X_u = (1, 0, f_u), X_v = (0, 1, f_v)$, and the second order partials are also defined accordingly.

Notations above follows that the shape matrix is:

$$A = \frac{1}{l} \begin{pmatrix} f_{uu} & f_{uv} \\ f_{vu} & f_{vv} \end{pmatrix} \begin{pmatrix} 1 + f_{uu}^2 & f_u f_v \\ f_u f_v & 1 + f_{vv}^2 \end{pmatrix}^{-1}, \quad (2.5.0.6)$$

where

$$l = \sqrt{(1 + f_u^2 + f_v^2)}$$

The Gaussian curvature K , is the product of the principal curvatures (the eigenvalues of A). That is, $K = \det(A)$,

$$K = \det(A) = \frac{f_{uu}f_{vv} - f_{uv}f_{vu}}{(1 + f_u^2 + f_v^2)^{1.5}} \quad (2.5.0.7)$$

Similarly, the Mean curvature is

$$K_m = \frac{1}{l}(f_{uu}(1 + f_v^2) + f_{vv}(1 + f_u^2) - 2f_u f_v f_{uv}) \quad (2.5.0.8)$$

The sign of K reflects the shape of the surface: Planar and cylindrical objects have a curvature K close to zero, whereas spherical objects possess positive and objects featuring saddle-shaped surfaces possess negative curvature, locally (see Figure 2.10).

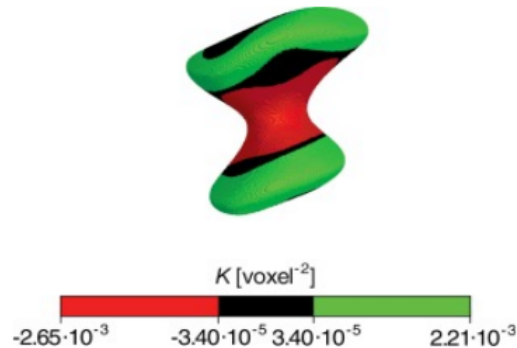


Figure 2.9. Visualization of the Gaussian curvature on the test object of Gold feather. Modified from [22].

To be more precise about this, the signs of the principal curvatures, k_1, k_2, k_3 , are used to characterize the points on the surface. At the points where $k_1, k_2, k_3 > 0$, the surface is locally convex and it bends outward in the direction of the chosen normal vector. This follows from the fact that the principal curvatures are the stretching factors on the surface in the direction of the respective principal direction. The intuition can easily be visualized for 2D image (visualized as 3D surface) or binary image, in which case there are only two principal curvatures k_1, k_2 .

Nuclei Separation algorithm used by Gaussian curvature criterion

Step 1. A 3D-labeling algorithm using a 26-neighborhood is applied (Image Labeling is the process of recognizing different entities in an image obtained from the connected component).

Step 2. The Gaussian curvature for all surface voxels using the binary image is computed.

Step 3. The next step is to remove voxels where objects overlap. Therefore, the minimal negative curvature value K_{min} is used and a threshold K_{thr} such that ψ percent of the voxels with negative curvature are lying in the interval from K_{min} to K_{thr} is selected. In other words, K_{thr} is the ψ -quantile of the interval from K_{min} to 0. The voxels in this interval are expected to be located at the places of contact between objects and are set to zero in the image.

Step 4. To smoothen the shape of the resulting novel surfaces, a median filter with a $3 * 3 * 3$ mask is applied to the image. This removes single voxels standing out or missing from the surfaces [13,40].

Modification of separation algorithm used by Gaussian curvature criterion

As mentioned in [13], A point on an iso-intensity surface has two principal curvatures, which can be calculated from Gaussian curvature. The smaller of the two principal curvatures is positive at any point in a single Gaussian distribution but is negative around the border of two Gaussian distributions (S2Fig). Therefore, once voxels that have negative curvature are removed from the blob, two or more nuclei should be detected easily in 3D images.

In the case of C.elegans data, the larger of the two principal curvatures was chosen because it has a sharper change around the border of two Gaussian distributions and could be more useful when the nuclei are densely distributed (Figure 2.10, S2Fig).

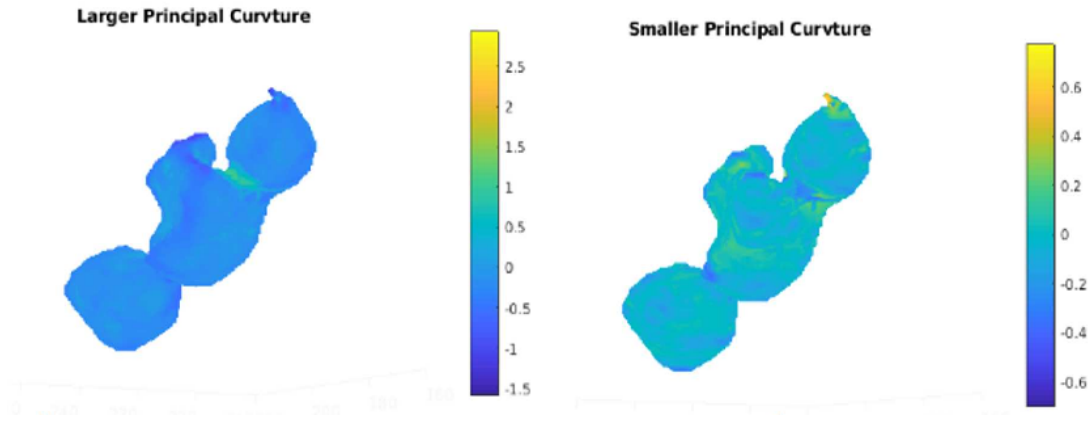


Figure 2.10. Visualization of the Principle curvatures on the 3 overlap nuclei. Larger Principle curvature has sharper changes around border of two Gaussian distributions.

In addition, all values of Gaussian curvature and its principals have multiplied in minus due to the rotation of data in a way that normal vector of it becomes $[-1 \ 0 \ 0]$. This helps to describe data by XY instead of XYZ.

If the nuclei considered to be ellipsoid with the maximum diameters of calculated in S1Fig, The maximum value of S (surface of nuclei) could be approximated by 3338 voxel^2 for the second view and 3590 voxel^2 for the first view (how to calculate S is shown in S1Table).

Using one of the principal curvature and repetition of more than once of the separation method could bring us to a new algorithm that is shown in the following:

Step 1. 3D-labelling algorithm using a 26-neighborhood is applied.

Step 2. For labelled objects larger than S , the larger principle curvature K_l is calculated and the voxels between 0 and K_{lmax} are removed. A 3D watershed is applied. If no separation happens, again larger principle curvature is calculated and the voxels with values between $K_{max}/2$ and K_{max} are removed. A 3D watershed is applied. (The watershed algorithm tends to over segment mostly or undersegment if it is not guided by Gaussian curvature separation.) (Figure 2.10 a,b)

Step 3. To smoothen the shape of the resulting novel surfaces, an order filter with a $3 * 3 * 3$ mask on the image is applied.

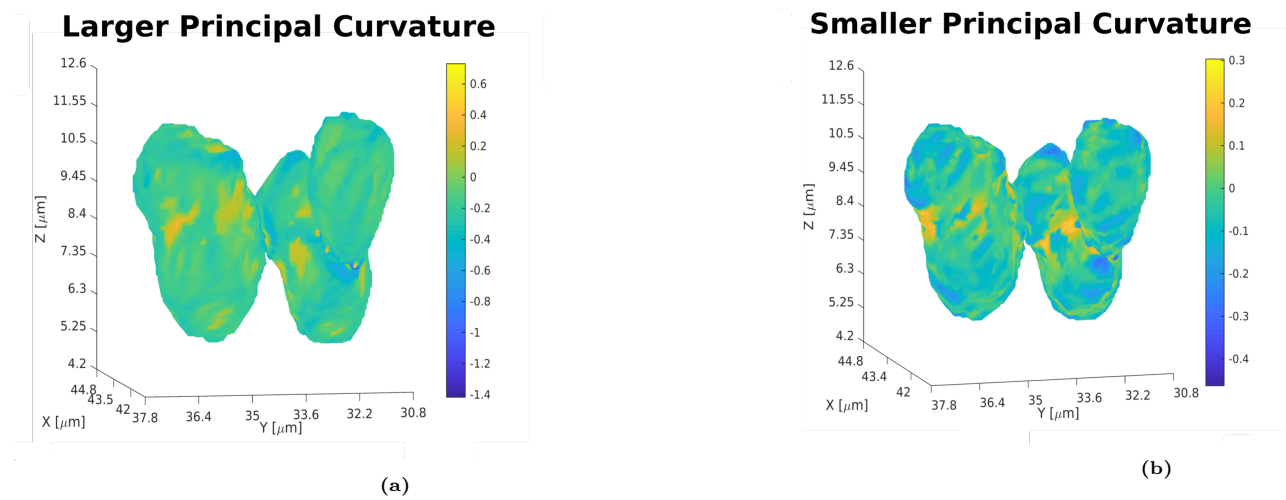


Figure 2.11. a) Larger principle of Gaussian curvature , b) Smaller principle of Gaussian curvature of 3 overlap nuclei. The Larger principle has a sharper change around the border of two Gaussian distributions and could be more useful when the nuclei are densely distributed.

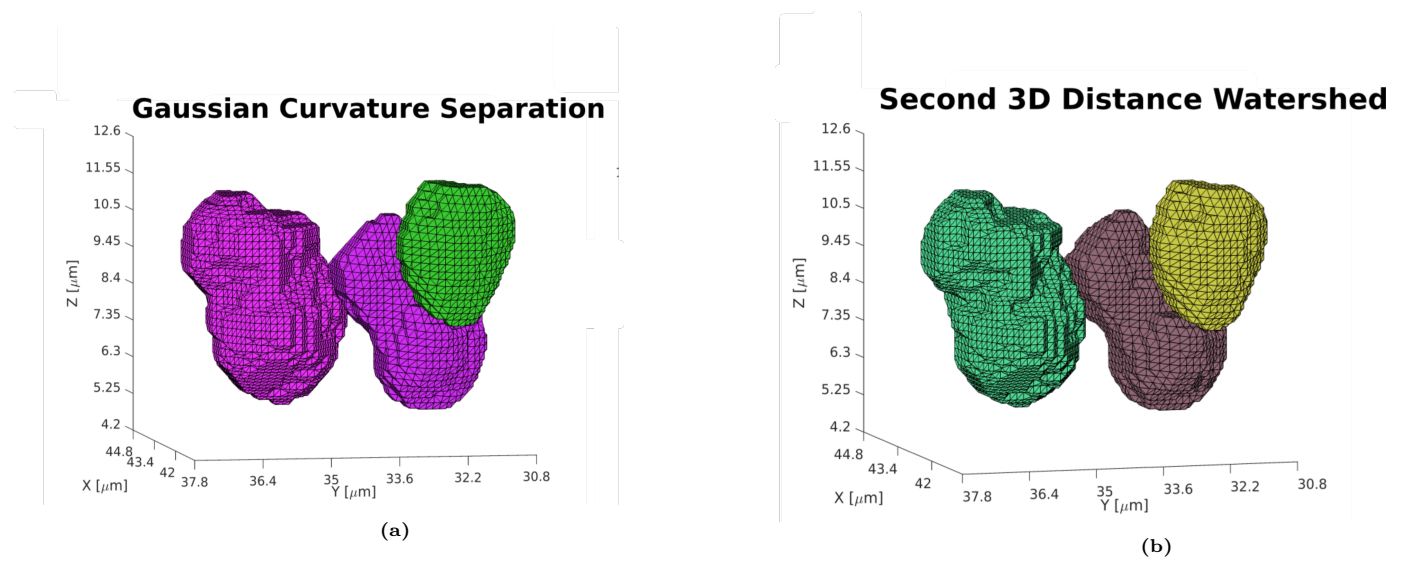


Figure 2.12. Overlap nuclei after Gaussian curvature separation (a) and after that 3D watershed distance transformation (b). Same color shows overlap nuclei. There are 2 separated nuclei in (a) and 3 separated nuclei in (b).

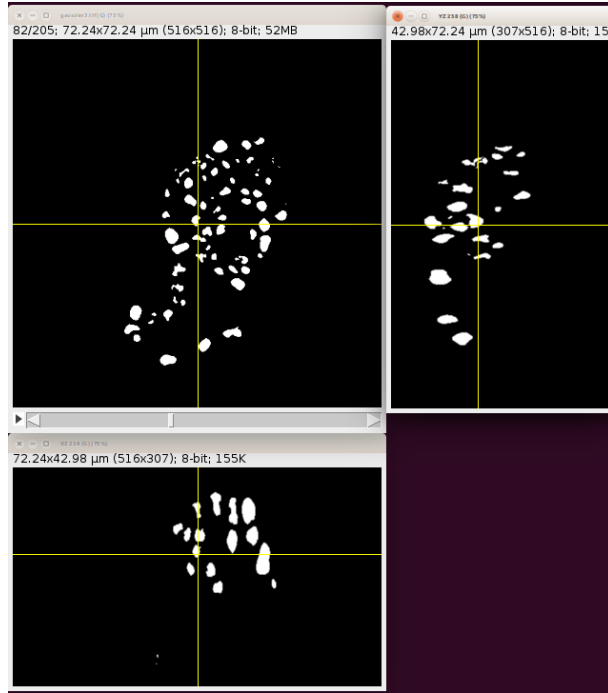


Figure 2.13. The orthogonal views of binary image after 3D segmentation.

2.6 2D nuclei separation based on graph cut and convexity–concavity analysis.

If R is a simply connected region bounded by a rectifiable simple closed curve S (Figure 2.14) [11,12], then the convex hull of R , indicated by R is simply connected and its boundary made up of a finite sequence of curves S_1, S_2, S_3, \dots . Each $S_j, j = 1, 2, 3, \dots$, is either an arc or a chord of S .

Each chord indicated by $K_i, i = 1, 2, 3, \dots$, is a line of support of S . B_i is the curve with the same end-points as K_i but is a part of the boundary curve S . The point on curve B_i with the longest distance d_i to equivalent chord K_i is indicated by P_i . The point P can be called the steepest concave point (SCP) if (The following derivation is reproduced from Ref [11].)

$$P_1 = \underset{P_i}{\operatorname{argmax}} d_i. \quad (2.6.0.1)$$

In Figure 2.14, the SCP is P_1 and the subscript index of the SCP is $t = 2, i.e. t = 1$. Point P between all points on boundary curves $\{B_i, i \neq t\}$ is called the nearest boundary point (NBP) if

$$P_2 = \underset{P \in \{B_i, i \neq t\}}{\operatorname{argmin}} D(P_t, P), \quad (2.6.0.2)$$

where $D(P_t, P)$ is the distance between the SCP P_t and the boundary point P belong to the boundary curves $\{B_i, i \neq t\}$.

It should be noted that point P in Eq. (2.6.0.2) should not lie on the boundary curve B_t where the SCP positions. It means P_2 shouldn't be on boundary curve B_1 . (Since the closest point to P_1 is on boundary curve B_1 .) The line connecting points SCP and NBP is an optimal separating path, as shown in Figure 2.14 in green color. If the next longest distance d_t that is equivalent to the next steepest point P_t is less than a distance threshold T_d , then the task of separating 2D labeled component R is completed. Otherwise, the 2D labelled component R is separated into two parts, R_1 , and R_2 along the optimal separating path. For each separated part, the repetition of the above separating process should be done until distance d_t is less than the threshold T_d .

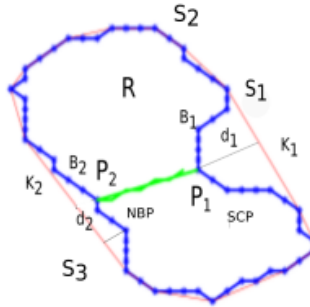


Figure 2.14. A connected component R and its convex hull. R is a connected region bounded by a rectifiable closed curve S . 2 chords K_1 , K_2 and 2 arcs S_1 , S_2 are shown. B_i is the curve with the same end-points as K_i but is a part of the boundary curve S . Points that belong to curve B_i and have the longest distance d_i to equivalent chord K_i are shown by P_i . P_1 and P_2 are defined in equations 2.6.0.1 and 2.6.0.2. P_1 and P_2 are also called the steepest concave point (SCP) and the nearest boundary point (NBP). The green line separates 2 nuclei.

Nuclei splitting algorithm based on convexity–concavity analysis is as following

1. Take all 2D labelled components from segmented foreground and make a list L_i out of them.
2. Take all 2D labelled components from list L_i with perimeter threshold bigger than Pr_d and make a list L out of them.
3. For each 2D labelled component R in list L , find the point with maximum concavity (SCP) P_t according to Eq. (2.6.0.1).
4. If equivalent d_t (computed differently for different data) is less than threshold T_d , delete this 2D labelled component R from list L and return to step 3. Otherwise, go to step 4.
5. Find the equivalent nearest boundary point P_b according to Eq. (2.5.09).
6. Separate 2D labelled component R into two parts R_1 and R_2 along the optimal separating line segment connecting Point P_t and the nearest boundary point (NBP). Add the split parts R_1 and R_2 back to the list L .
7. Iterate steps 2, 3, 4, 5, 6 until list L is empty.

2.7 2D Concavity-Convexity Analysis

In order to separate multiple overlapping nuclei, this procedure has to be iteratively applied to the data. However, more than three times iterations often remove parts of the nuclei and reduce their apparent size. To overcome this problem, an approach based on 2D convexity-concavity concave object separation [10,11,12] is used to separate nuclei in 2D images too. The idea here is to find the 2 steepest concave points on the contour of connected nuclei based on convexhull analysis (section 2.6).

The separation in this method happens by connecting these two points (Figure 2.15). The sign of 2D curvature, positive or negative, corresponds to convexities and concavities, respectively, but in both cases, there are also important singularities along the contour when curvature reaches a local maximum or a local minimum. To avoid these singularities the collinearity of the points of the objects should be considered before. A mixture of these two separation approaches is able to separate nuclei completely (Figure 2.16, Table A.1).

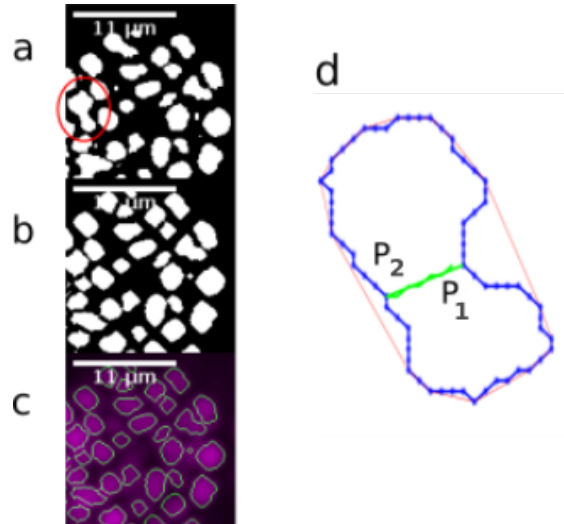


Figure 2.15. 2D concavity-convexity analysis separation. a) Part of 2D slice of binary image before 2D convex-concavity separation at depth $z=60$. b) The same part after separation. c) contour of result of separation and the original image at the the same depth. d) chosen nuclei from (a). P_1 deepest point of boundary with respect to convex-hull of nuclei (red contour). P_2 closest point to P_1 that doesn't belong to boundary curve of P_1 . 2 nuclei get separated by green line .

2.8 Final result of real 3D images segmentation

The Filtered image stack of C.elegans embryo first was binarized with a 2D stack histogram, then was separated by the 3D curvature separation method. After that 2D concavity method was used to separate remains.

With the 2D-histogram binary image number of nuclei identified and labeled is 191 (Figure 2.17a). After 2 steps of Gaussian curvature separation, it increases to 273 (Figure 2.17b). The number increased to 344 After using 2D convexity-concavity separation analysis(Figure 2.17c).

2.9 Evaluation

par:Evaluation

To validate the proposed methods, other states of the art methods were used on datasets for segmentation. These methods include: Ilastik [18], 3D watershed segmentation plugin (MorphoLibJ) in Fiji [41], LOS [42,43], Cellsegm [44], Toyoshima segmentation [13].

The optimization of the parameters of the other method was done based on accuracy and precision criteria. Since binarization of all methods results in a large number of overlap of nuclei, the binary image was input to all methods. The exception is Ilastik, this software uses a training approach to make a binary image out of the grayscale image. 3D watershed segmentation plugin (MorphoLibJ) in Fiji, uses distance transform watershed, which allows us to separate touching objects by combining the distance transform and watershed methods. LOS method is based on a computation of inner visibility between points on the surface of the shape and the Cellsegm toolbox uses an adaptive method to find seeds for watershed segmentation and then followed by cell splitting.

Cellsegm and LOS were run on a cluster environment, due to having an additional part for splitting the cells. Ilastik and 3D watershed segmentation plugin were run on PC.

The groundtruth obtained by manual 2D ellipsoid fitting of nuclei in embryos. To calculate TP (true positive), FP (false positive), and FN (false negative) values, the centers of the segmented area and spherical area of groundtruth were used. If the distance between these centers is less than the peak diameter of nuclei (S1Fig), then it counts as a TP otherwise FP. The nuclei which are not in the segmented image will be considered as FN.

Other parameters calculated for comparison are: Precision, Recall, Accuracy, True positive rate, False-negative rate, and False positive rate. (The following derivation is reproduced from Refs [11,13,26]).

$$Precision = \frac{TP}{TP + FP}$$

$$Accuracy = \frac{TP}{TP + FN + FP}$$

$$Recall = \frac{TP}{(TP + FN)}$$

$$True\ negative\ rate = \frac{TN}{GT}$$

$$False\ positive\ rate = \frac{FP}{GT}$$

$$False\ negative\ rate = \frac{FN}{GT}$$

$$True\ positive\ rate = 1 - False\ negative\ rate$$

Where GT is the number of nuclei in groundtruth.

2.10 Evaluation of Results

The performance of the proposed method with five previously published methods for nuclei segmentation is compared (Figure 2.18). 3D watershed segmentation plugin (MorphoLibJ) in Fiji [41], uses distance transform watershed, which allows us to separate touching objects by combining the distance transform and watershed methods. Ilastik [18] is based on machine learning techniques and uses image features such as Laplacian of Gaussian. LOS [42,43] method is based on a computation of inner visibility between points on the surface of the shape. This method was used on mouse embryo and cancer spheroid. CellSegm [44], is a Matlab based command-line software toolbox providing an automated whole-cell segmentation of images showing surface stained cells, acquired by fluorescence microscopy. Toyoshima segmentation [13] is based on calculating the smaller 3D Gaussian curvature for an adult C.elegans embryo. These six methods for 3 animals in 3 data sets were used and the performance criteria (Table 3.1, see par:Evaluation) were obtained. The parameters of each method were optimized for the datasets.

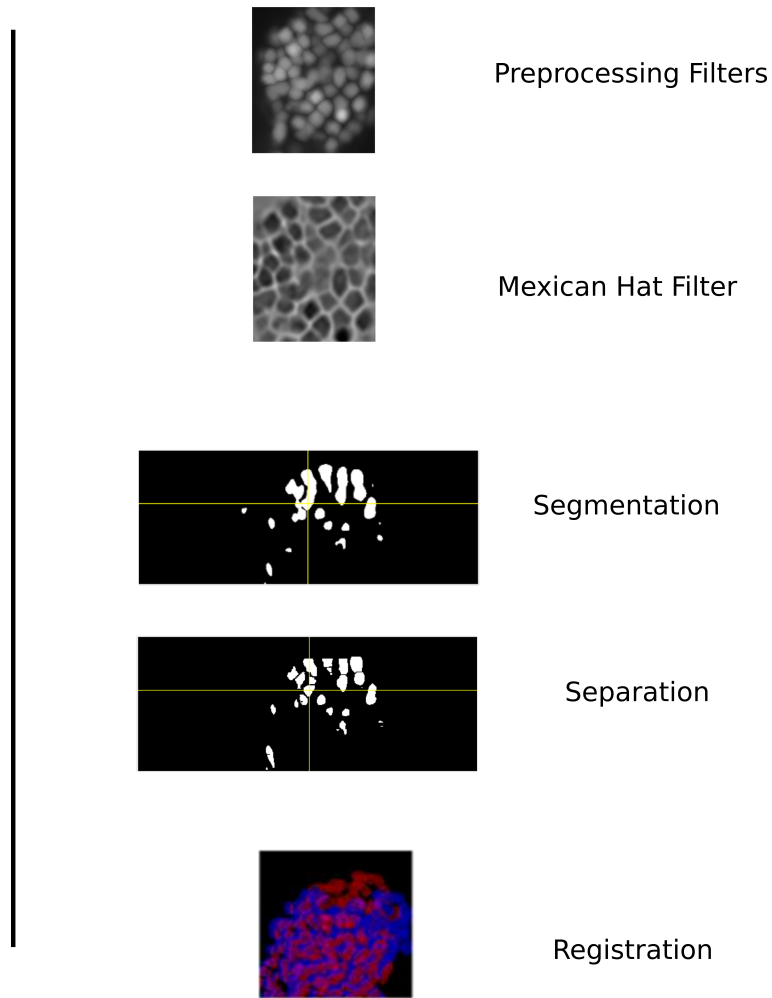


Figure 2.16. Schematic of analysis pipeline to segment nuclei. Analysis is done on a 3D stack image of C.elegans embryo (nuclei channel) by the algorithm. This data goes through the pipeline and the result is a 3D segmented stack of the embryo. This segmented data can be used to register the embryos in different views.

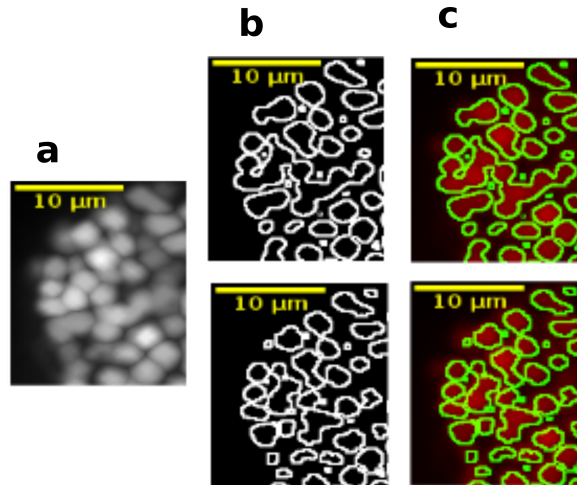


Figure 2.17. Enlarged area of 2D slice during separation. a) 2D slice of original data at depth $z=60$, b) Outline of binary image at depth $z=60$ up) Initial state of separation procedure, down) Final result of separation procedure. c) overlay of (b) and (c).

The 3D images in our dataset contain 386 nuclei on average. The proposed method found 96% of the nuclei and the false-negative rate was 3%. The false-negative rate of the other methods were more than 11%. The false-positive rate of the proposed method was 6% and for the other methods were between 0.5% and 6%. The proposed method shows the highest performance according to the true positive rate and precision and recall criteria values. Table 2.1 shows the result of this computation for 3 data sets.

It should be added that all of the methods used for comparison were unable to binarize the data correctly and either number of connected nuclei are high or background counted as nuclei. For this reason, the binary image obtained by the proposed method used as input to the other methods.

These results certainly indicate that our proposed method detects super densely distributed cell nuclei in 3D space with the highest accuracy. A very low false-negative rate is another improvement made by this method. These improvements will cause a drastic change in nuclei segmentation compare to other methods.

Table 2.1. Evaluation of different segmentation methods: Mean values between 3 data sets are calculated.

	TP	FP	FN	TP rate	FP rate	FN rate	Precision	Accuracy	Recall
Proposed	365.5	24	12	0.96	0.06	0.03	0.96	0.90	0.50
LOS	289.5	4	73	0.84	0.005	0.21	0.98	0.78	0.45
Ilastik	162	4	200	0.47	0.02	0.58	0.96	0.40	0.32
Cellsegm	234	10	41	0.68	0.02	0.11	0.85	0.82	0.40
Fiji 3D watershed	246	43	97	0.71	0.06	0.28	0.91	0.67	0.43
Toyoshima segmentation	232	9	44	0.67	0.02	0.12	0.96	0.81	0.40

The Table contains the evaluation parameters of the proposed method and the other state of the art methods. The definitions of criteria are included in the text.

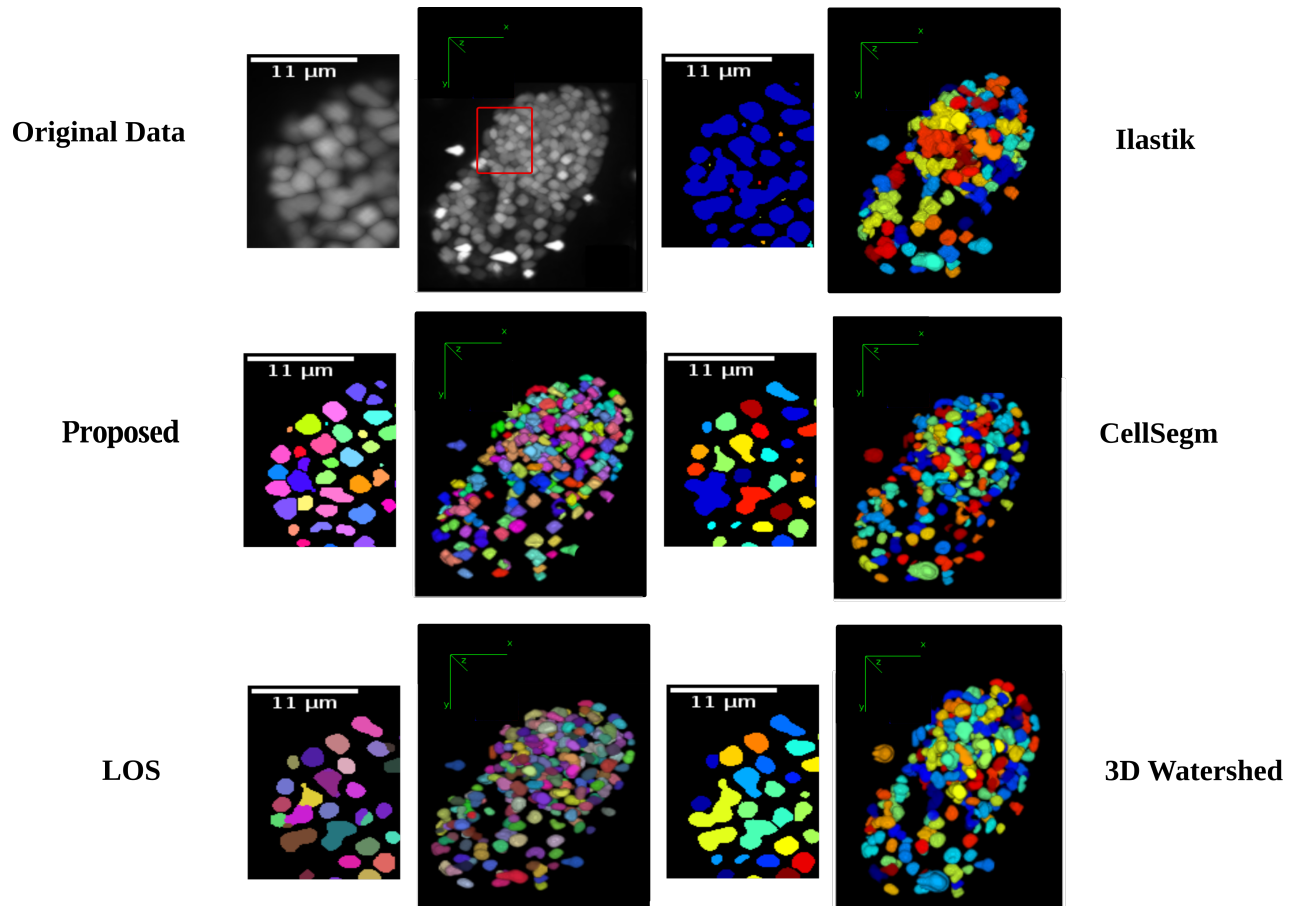


Figure 2.18. 3D Evaluation of detected nuclei by the proposed method. Overlay of orthogonal views of original data and ellipsoid fitting at depth=60. Yellow, red, and green ellipsoids show True Positive, False Negative, False Positive results respectively. The nuclei which are not binarized are the ones that their intensity pixels, due to the limitation of z-axis resolution, are too low.

2.11 Optimized parameter values

Optimized parameter values will make the model to do the task with relative accuracy. The cost function inputs a set of parameters and output is a cost that measures how well that set of parameters performs the task (on the training set). For the c-elegans data, we should have 2 parameter tuning one for the 2D separation and one for the 3D separation (Table 2.2).

1. Step 1. Decouple search parameters from code. Take the parameters that we should tune and put them in an array at the top of the code.
2. Step 2. Wrap training and evaluation into a function (usually distance function or number of separated nuclei).
3. Step 3. Run Hyperparameter Tuning script.

2.12 Comparison with Lines-of-Sight decomposition method

Line of sight (LOS) is an approach that has been used to separate overlap nuclei recently (2015) [17,43].

This method is based on a computation of inner visibility between points on the surface of the shape. Two points on the surface of a shape are mutually visible and are said to be in a line-of-sight (LOS) if the straight line segment between them does not leave the inner volume of the shape. Two surface points or any clique of points that are in LOS are also said to be in a convex position.

Table 2.2. S1: Optimizations of parameters used in the proposed method.

Process	parameter	Initial step	Final step	Optimized Parameters
Preprocessing	3D Gaussian Blur	1*1*1 pixels	4*4*4 pixels	2*2*2 pixels
Preprocessing	3D Maximum	1*1*1 pixels	4*4*4 pixels	2*2*2 pixels
Preprocessing	LOG	11*11*15 pixels	17*17*23 pixels	13*13*19 pixels
Segmentation	3D Gaussian Curvature Separation	$S=2003$ voxel ³	$S=8000$ voxel ³	$S=5101$ voxel ³
Segmentation	2D Concave-Convex Separation	$T_d=1.5$ pixels	$T_d=2.5$ pixels	$T_d=2$ pixels
Segmentation	2D Concave-Convex Separation	Perimeter Threshold=20 pixels.	Perimeter Threshold=45 pixels.	Perimeter Threshold=40 pixels.

This method doesn't need to calculate the exact value of curvature and has been able to separate the nuclei of the early mouse embryo and two different cellular spheroids successfully. This method is used for the overlap area with the weak convex property.

If B_R is the family of all objects with radius R in n -dimensional euclidean space E_n . Given any points x_1 and x_2 whose distance is less than or equal to $2R$, $DR(x_1, x_2)$ is denoted as of all objects in B_R which contains x_1 and x_2 . A set C is said to be strongly convex with respect to R if any two points of x_1 and x_2 in C , $DR(x_1, x_2)$ is constrained in C . A set C is said to be weakly convex with respect to C , if for any two points x_1 and x_2 in C , $DR(x_1, x_2)$ intersects the closure of C at, at least, one point distinct [42].

Because of this, this method has the same disadvantage of concave points for 2D contours, if the surface of the overlap area doesn't have a smooth shape, it could fall into a local minimum, thus it is not a robust separation in compare to Gaussian curvature for C.elegans embryo nuclei separation.

2.13 Summery

To be able to run the registration pipeline, nuclei should be segmented first. We have developed a method to segment the nuclei during morphogenesis in the C.elegans embryo. Nuclei during morphogenesis are densely distributed and they have an irregular shape (S1Fig). None of the states of the art methods like LOS [43], Toyoshima segmentation [13], and other methods were able to segment the nuclei accurately. The proposed method uses 3D and 2D Gaussian curvature to separate the nuclei (Figure 2.17).

Chapter 3

Feature Selection

Feature selection, also known as variable selection or quality selection is the process of selecting a subset of features for use in model construction. The main use of feature selection is to make the models easier to interpret for researchers [45]. The main difference between feature selection and extraction is that feature selection keeps a subset of the original features while feature extraction creates brand new ones. Feature selection could also reduce the dimensionality of data.

3.1 Local geometry descriptor

To find the features for registration of different views of biology samples, centres of nuclei are considered as beads and the geometric local descriptor method developed by Prebish et al [5] can be used to extract the features.

To register two views A and B the corresponding bead pairs (a, b) have to be identified invariantly to translation and rotation. The local descriptor of a bead can be defined by the locations of its 3 nearest neighbors in 3D image space ordered by their distance to the bead. To efficiently extract the nearest neighbors in the image space K-nearest neighborhood algorithm is used. Translation invariance is achieved by storing locations relative to the bead. That is, each bead descriptor is an ordered 3D point cloud of cardinality 3 with its origin $\vec{o} = (0, 0, 0)^T$ being the location of the bead.

Local descriptor matching is performed invariantly to rotation by mapping the ordered point cloud of all beads $\vec{a} \in A$ to that of all beads $\vec{b} \in B$ individually by means of least square point mapping error using the closed-form unit quaternion-based solution or any other point set registration methods. The similarity measure ϵ is the average point mapping error. Each candidate in A is matched against each candidate in B. Corresponding descriptors are those with minimal ϵ . This approach, however, is computationally very demanding as it has a complexity of $O(n^2)$ regarding the number of detections.

Therefore a variation of geometric hashing was employed to speed up the matching process. Instead of using one reference coordinate system for the complete scene a local coordinate system for each of the descriptors is used. It means each bead is the center of a coordination system that could be defined with the help of 3 closest beads (Figure 3.1). All remaining bead coordinates not used for defining the local coordinate system become rotation invariant which makes comparing the descriptors very efficiently using K-nearest neighborhood to index remaining bead coordinates in the local coordinate system.

Figure 3.1 shows the descriptor build-up of the method.

3.2 Gaussian mixture parameters

Since we are matching images of nuclei rather than just points, we can use the additional information from the shape of each nucleus. We add this information to the representation of each nucleus by adjusting the amplitude and standard deviation of the Gaussian. The Gaussian mixture representation of an image is given by, (The following derivation is reproduced from Ref [20].)

$$f(\xi, x) = \sum_{i=0}^{N-1} A_i \exp\left(-\frac{\|\xi - x_i\|^2}{2(\lambda\sigma_i)^2}\right), \quad (0.2.0.1)$$

Where A_i , x_i , and σ_i are the amplitude, mean, and standard deviation of the i-th Gaussian.

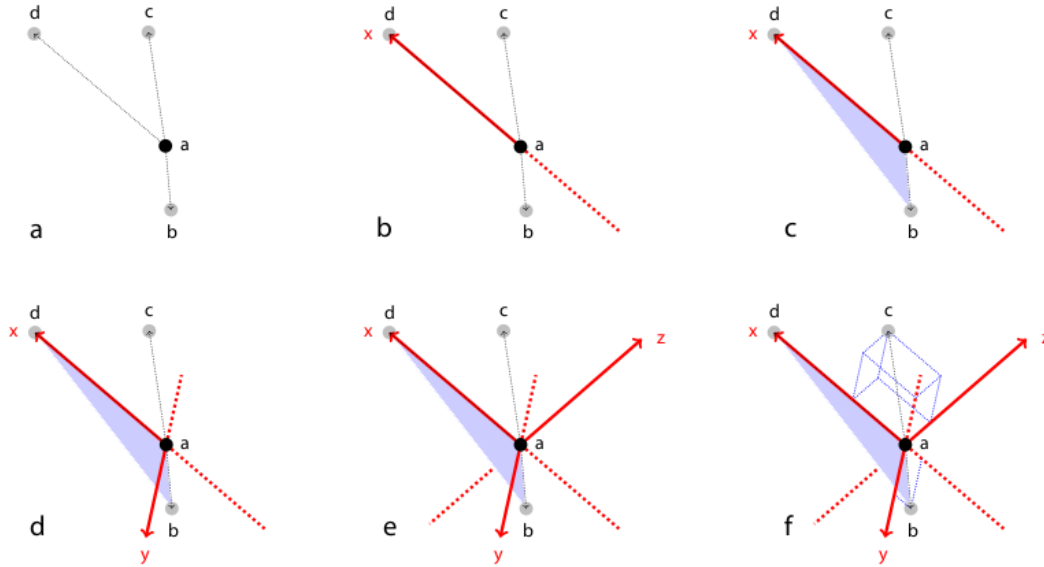


Figure 3.1. Rotation invariant local geometric descriptor. (a) Bead a is described by the constellation of its three nearest neighbors b , c , and d . (b) The local x -axis is defined by the vector pointing from a to d . Optionally scale invariance can be achieved by normalizing the length of the local x -axis, which is, however, for this particular registration case not necessary. (c) The first-nearest neighbor b and the third-nearest neighbor d define a plane. (d) The local y -axis lies perpendicular to the local x -axis in this plane pointing towards b . (e) The local z -axis lies perpendicular to the local x and y -axes pointing towards c . (f) Six rotation invariant values are extracted from the local geometric descriptor using the defined coordinate system; the length of d , xy -coordinates of b , and xyz -coordinates of c . For scale invariant descriptors, only five values are extracted as the length of d on the x -axis is always one. Figure reproduced from Ref [5].

These parameters are derived from the brightness, centroid, and size of the segmented neuron, while ξ is the 3D spatial coordinate. A scale factor λ is added to the standard deviation to scale the size of each Gaussian. This will be used later during gradient descent. The sample constellation of nuclei is then represented by the Gaussian mixture $f(\xi, X)$. Similarly, the reference constellation's own nuclei are represented as a $f(\xi, R)$.

To match a sample constellation of nuclei X with a reference constellation of nuclei R , non-rigid transformation $u : IR^3 \rightarrow IR^3$ could be used. The transformation maps X to $u[X]$ such that the L_2 distance between $f(\xi, u[X])$ and $f(\xi, R)$ is minimized with some constraint on the amount of deformation.

This can be written as an energy minimization problem, with the energy of the transformation, $E(u)$, written as

$$E(u) = \int [f(\xi, u[X]) - f(\xi, R)]^2 d\xi + E_{Deformation}(u). \quad (3.2.0.1)$$

Note that the point-sets X and R are allowed to have different numbers of points. The deformations as a thin-plate spline (TPS) could be modeled.

The minimization of E is found by gradient descent. Working with Gaussian mixtures as opposed to the original images gives the model of the deformations and analytically compute the gradients of eqs.(3.2.0.1) making gradient descent more efficient [20].

Since for C.elegans data rigid transformation is needed and differences in the shape of the nuclei in different views are larger than ones in the tracking procedure, this registration is not a good solution, however Gaussian parameter of each nucleus could be used for feature selection.

3.3 Feature Vectors

The features which are added to feature vector included:

Centre of mass of the region, returned as a 1-by-3 vector of the form [*centroid-x centroid-y* and *centroid-z*]. The first element, *centroid-x*, is the horizontal coordinate (or x-coordinate) of the centre of mass. The second element, *centroid-y*, is the vertical coordinate (or y-coordinate). The third element, *centroid-z*, is the planar coordinate (or z-coordinate).

Volume is the count of the actual number of 'on' voxels in the region, returned as a scalar. Volume represents the metric or measure of the number of voxels in the regions within the volumetric binary image.

Dot product of the first eigenvectors

Eigenvectors

Eigenvectors are eigenvectors of the voxels representing a region, returned as a 3-by-3 vector. Eigenvectors could be used to calculate the orientation of the ellipsoid that has the same normalized second central moments as the region. For 3D labeled area there are three eigenvectors, To save time and computation the first eigenvector is chosen to represent the orientation of the labeled area with respect to each other.

First eigenvalue

Eigenvalues are eigenvalues of the voxels representing a region, returned as a 3-by-1 vector.

Eigenvalues could be used to calculate the principal axes lengths of the ellipsoid that has the same normalized second central moments as the region. For 3D labeled area there are three eigenvalues, to save time and computation the first eigenvalue is chosen to represent the Length (in voxels) of the major axes of the labeled area.

Eccentricity

In mathematics, the eccentricity denoted e or ϵ , is a parameter associated with every conic section. It can be thought of as a measure of how much the conic section deviates from being circular. In particular,

- The eccentricity of a circle is zero.
- The eccentricity of an ellipse which is not a circle is greater than zero but less than 1.
- The eccentricity of a parabola is 1.
- The eccentricity of a hyperbola is greater than 1 (Table 3.1).

Table 3.1. Eccentricity of all types of conic sections.

Conic section	Equation	Eccentricity (e)	Linear eccentricity (c)
Circle	$x^2 + y^2 = r^2$	0	0
Ellipse	$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$ or $\frac{y^2}{a^2} + \frac{x^2}{b^2} = 1$ where $a > b$	$\sqrt{1 - \frac{b^2}{a^2}}$	$\sqrt{a^2 - b^2}$
Parabola	$x^2 = 4ay$	1	
Hyperbola	$\frac{x^2}{a^2} - \frac{y^2}{b^2} = 1$ or $\frac{y^2}{a^2} - \frac{x^2}{b^2} = 1$	$\sqrt{1 + \frac{b^2}{a^2}}$	$\sqrt{a^2 + b^2}$

The eccentricity of nuclei is the eccentricity of the ellipse that has the same second-moments as the region and is returned as a scalar. The eccentricity is the ratio of the distance between the foci of the ellipse and its major axis length.

For 3D objects like nuclei, the meridional and equatorial eccentricity are calculated.

$$\text{Meridional Eccentricity} = \sqrt{-(\text{shortest radius}/\text{longest radius})^2};$$

$$\text{Equatorial Eccentricity} = \sqrt{1 - (\text{shortest radius}/\text{second longest radius})^2};$$

Feature Correction

For 3D objects like nuclei, we have 3 eigenvectors and eigenvalues. For feature vector, the first eigenvector and first eigenvalue were used. The value of eigenvectors and eigenvalues usually are sorted from the highest to the lowest. It means the first eigenvector and eigenvalue are the largest ones so if the nuclei have a different direction they can have the same first eigenvector. As you can see in Figure 3.2, two ellipsoids have the same first eigenvalue but one is stretched along -x-direction (left) and the other one is stretched along -z-direction (right).

To find nuclei with similar shapes by using this feature, we should consider the direction of nuclei too. To overcome this problem another feature called BoundingBox was used. This feature is smallest cuboid containing the region, returned as a 1-by-6 vector of the form $[ulf_x \ ulf_y \ ulf_z \ width_x \ width_y \ width_z]$. $ulf_x, ulf_y,$ and ulf_z specify the upper-left front corner of the cuboid. $width_x, width_y,$ and $width_z$ specify the width of the cuboid along each dimension. Based on the BoundingBox feature especially the last ones (4:6 elements) we can find in which direction nuclei have the largest elongation. (This feature was n't used directly in the feature vector.)

Based on this correction, it can be shown each eigenvector (or eigenvalue) belong to which direction -x -y and -z (approximately).

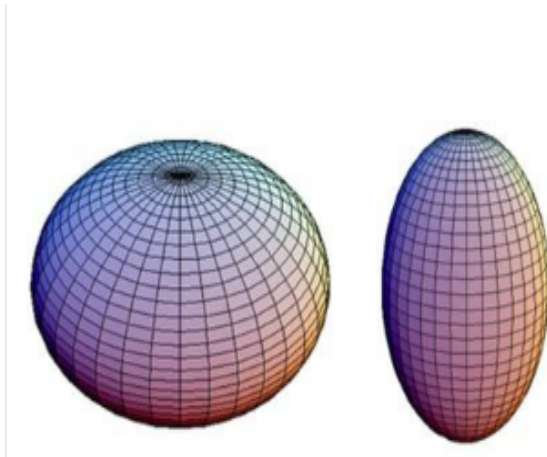


Figure 3.2. Two ellipsoid with the same first eigenvector and eigenvalue. Figure reproduced from Matlab Image processing toolbox.

3.4 Producing Feature Vectors

Considering the methods above, the algorithm that is used for feature selection is the following:

Step 1. Nuclei will be 3D-labelled using a 26-neighborhood.

Step 2. For each labeled object an 11 element feature vector is made as explained above.

Step 3. For each labeled object 3 closest objects are found by the KNN algorithm based on section 3.1. KNN used 3D centres of feature vector at this step. The output of this step is the local geometric descriptors (Figure 3.3a).

Step 4. For each local geometric descriptor of second view 4 closest local geometric descriptors of the reference, the view is chosen by the KNN algorithm. The result is then sorted from the closest to the farthest.

Step 5. Of 4 local geometry descriptors of reference view one will be chosen that has the most similarity in other features than centres that include: volume, the dot product of the first eigenvectors, first eigenvalue and equatorial eccentricity and meridional eccentricity (Figure 3.3b).

Step 6. Centres of local geometry descriptors of two views are the result of feature selection or control points.

The nuclei that are found by the feature selection algorithm are shown in Figure 3.3. These nuclei are the 12 most similar nuclei in 2 views from the first pair nuclei to 12th pair nuclei altogether. As it is shown in Figure 3.3 (c)-(d) the pair nuclei are found that are similar in shape. The feature selection is successful in representing the shape and density of nuclei in C.elegans embryo too as it is shown in Figure 3.3 (e)-(g).

Nuclei have higher density in the head part than the tail part, so the most similar nuclei are mostly distributed in head (and the area between head and tail) part. If we reduce the number of nuclei in the head part by using 3D histogram binarization, the result of feature selection could show the shape of the embryo better.

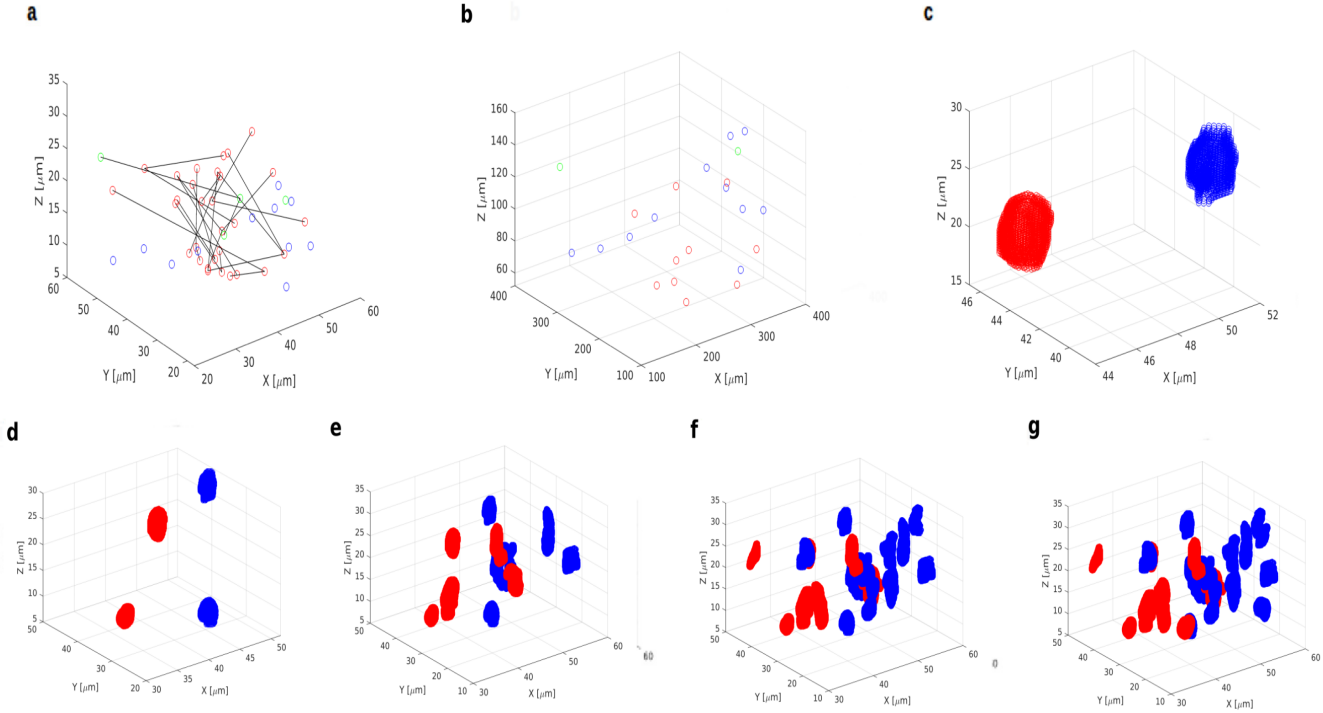


Figure 3.3. 12 most similar nuclei/centers from reference view (red) and second view (blue) a) After geometry hashing of local descriptors. b) After Feature vectors. The best matches are shown with green color. (c)-(g) The pair nuclei which are result of feature selection shown from the first pair nuclei to 12th pair . (c)-(d) The pair nuclei which are selected, are similar in shape. (e)- (g) The feature selection is successful in representing the shape and density of nuclei in c-elegans embryo too.

3.5 Optimizing feature vector

The first part of feature selection is to find the nuclei based on their neighborhood nuclei. The second part of feature selection is finding the nuclei based on feature vectors include: volume, the dot product of the first eigenvectors, first eigenvalue, and equatorial eccentricity and meridional eccentricity (defined by 3 diameters of nuclei: section 3.3.3: Eccentricity).

Nuclei with the minimum distance between their feature vector values are the most similar ones.

$$Df_{ij} = \sum_k |x_{ki} - y_{kj}| \quad (3.5.0.1)$$

with Df_{ij} being the distance between feature vector values, x_{ki} is the value of k -th feature of i -th reference nucleus and y_{kj} the value of k -th feature of j -th transformed nucleus of the second view after registration.

i -th and j -th nucleus with the minimum Df_{ij} are the same nuclei from 2 sets.

These features have values in different ranges, for example, the volume is in the range of 1000 to 8002 $voxel^3$ but eccentricity is in range of .067 to 1.03.

To be able to compare these features, coefficients or parameters for these feature should be defined by normalization of different feature values.

$$Df_{ij} = \sum_k a_k |x_{ki} - y_{kj}| \quad (3.5.0.2)$$

After setting these parameters the similarity between features of two set after registration increased significantly by three methods: ICP, LMICP, EM.

3.6 Summery

To extract the features first local descriptor/geometry hashing is used on centers and then global feature descriptors are extracted from the nuclei shapes (S1Fig). The feature vector consists of information for both local descriptor/geometry hashing and global descriptors. Feature vector contains 3D center, volume, the dot product of the first eigenvectors, first eigenvalue, eccentricities. The pair nuclei are found by this algorithm are similar in shape and the result of the feature selection is successful in representing the shape and density of nuclei in C.elegans embryo.

Chapter 4

Point Set Registration

Point set registration, is the process of finding a spatial transformation that lines up two point sets. The reason of finding such a transformation comprises merging multiple data sets into a globally steady model, and mapping a new measurement to a known data set to identify features. In medical image analysis, point set registration is essential to match points or landmarks in 3D images for disease diagnosis, point-set-based image registration, image fusion, and construction of image atlases [50].

Point set registration is used in optical character recognition,[51] augmented reality [52] and aligning data from magnetic resonance imaging with computer aided tomography scans [53].

This document closely follows the notation introduced in the cited papers whenever possible. For efficiency and make the different part relevant the same notation is used sometimes.

4.1 Rigid registration

Given two point sets, rigid registration yields a rigid transformation which maps one point set to the other. A rigid transformation is defined as a transformation that does not change the distance between any two points. Typically such a transformation consists of translation and rotation [51]. In rare cases, the point set may also be mirrored.

4.1.1 Iterative closest point

Iterative closest point (ICP)[49] is an algorithm used to minimize the difference between two sets of points. ICP iteratively assigns correspondences based on a closest distance criterion and finds the least-squares rigid transformation relating the two point sets. The algorithm then redetermines the correspondences and continues until it reaches the local minimum.

If there are two sets of points in R^n ; which could be denoted by model and data, with their elements being denoted by $\{m_i\}_{i=1:N_m}$ and $d_{i=1:N_d}$: The task of registration is to determine the parameters of a transformation T which, when applied to the data points, best aligns model and data.

The parameters of T are represented by a p-vector a . The value of p for a 3d affine transformation is 12 (3*4 matrix).

Alignment is measured by an error function $\epsilon^2(|x|)$; and a typical choice is to define

$$\epsilon^2(|x|) = ||x||^2$$

In order to measure alignment, It is required that correspondence between the model and data points is specified. This correspondence is denoted by the function $\phi(i)$; which selects, for each data point, the corresponding model point. In order to cope with data points for which no correspondent is found, weights w_i is also introduced; which are set to zero for points with no match, and one otherwise. Thus, the error to be minimized is

$$E(a, \phi) = \sum_{i=1}^{N_d} w_i \epsilon^2(m_{\phi(i)} - T(a; d_i)) \quad (4.1.0.1)$$

In its simplest form, the ICP algorithm iterates two steps. Beginning with an initial estimate of the registration parameters, a_0 , the algorithm forms a sequence of estimates a_k which progressively reduce the error $E(a)$. Each iteration of the algorithm comprises the following two steps:

1. Compute correspondences, ϕ : Set

$$\phi(i) = \underset{j \in \{1, \dots, N_m\}}{\operatorname{argmin}} \varepsilon^2(m_j - T(a_k; d_i)) \quad i = 1 \dots N_d$$

So that $m_{\phi(i)}$ is the closest model point to the datum d_i transformed by the current estimate a_k .

2. Update transformation, a : Set

$$a_{k+1} = \underset{a}{\operatorname{argmin}} \sum_{i=1}^{N_d} \varepsilon^2(|m_{\phi(i)} - T(a; d_i)|)$$

In many common cases, step 2 can be performed in closed form, or via well understood, numerically stable, procedures such as singular value decomposition.

When the set of correspondences does not change in step 1, the value of a_{k+1} will be set equal to a_k in the step 1, so no further change is possible.

Many variants of ICP have been proposed that affect all phases of the algorithm from the selection and matching of points to the minimization strategy. ICP requires that the initial position of the two point sets be adequately close [27,51].

For c-elegans data to have a better initial position feature selection was used to find the best correspondences in previous steps.

Figure 4.1 is an example of ICP registration method for point cloud of a bunny and a dragon [30].

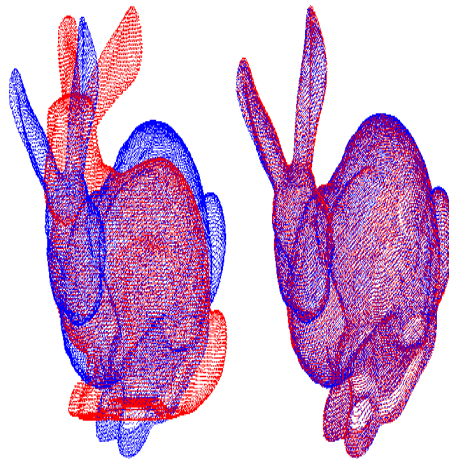


Figure 4.1. The point cloud data of a bunny (left) before, right) after registration by ICP.

4.1.2 Robust ICP by Trimming

Trimmed ICP [31] is a robust version of ICP and identifies possible outliers and removes them.

Trimming algorithm: if there are two data point sets S and M

1. Initialize set S to contain all n data points.
2. Repeat until convergence:
 - a) Use S to compute M (possibly with robust error function).
 - b) Compute misfit between each data point and M .
 - c) Remove k data points with the largest misfit from S .
3. Use S to compute M .

This method is a generalization of approximate trimmed least squares. The LM-ICP method use different kernels to get robus but it just consider the points. To choose the nuclei that are closer to eqach other trimmed ICP could be used . To have an 3d affine transformation at least 4 points are needed and to have a robust estimation 3 times of this number should be used [5].

4.1.3 The Levenberg-Marquardt algorithm

This method replaces the Euclidean distance with the Chamfer distance and uses a Levenberg-Marquardt algorithm to compute T_k , which accelerates the convergence of data registration, while keeping high accuracy. Especially in the treatment of high overlapping ratios, the LM-ICP method is superior to the standard ICP method [33,49].

$$E(a) = \sum_{i=1}^{Nd} E_i^2(a), \quad E_i(a) = \sqrt{\omega_i} \min_j \varepsilon(m_{\varphi(i)} - T(a; d_i))$$

An important concept in the derivation of LM will be the vector of residuals so that $E(a) = \|e(a)\|^2$.

The Levenberg-Marquardt algorithm combines the gradient descent and Gauss-Newton approaches to function minimization. Using the notation above, the goal at each iteration is to choose an update to the current estimate a_k , say x , so that setting $a_{k+1} = a_k + x$ reduces the error $E(a)$. Expanding $E(a + x)$ around a , equation below is obtained

$$E(a + x) = E(a) + (\nabla E(a).x) + \frac{1}{2!}((\nabla^2 E(a).x).x) + h.o.t.$$

Expressing this in terms of e , There are

$$E(a) = e^T e$$

$$\nabla E(a) = 2(\nabla e)^T e$$

$$\nabla^2 E(a) = 2(\nabla e)^T \nabla e + 2(\nabla e)^T \nabla e$$

It could be denoted the $N_d \times p$ Jacobian matrix ∇e by J , with ij th entry . Introducing the Gauss-Newton approximation , i.e. neglecting $(\nabla^2 e)e$, It could be arrived at

$$E(a + x) \approx e^T e + x^T J^T e + x^T J^T J x$$

The task at each iteration is to determine a step x which will minimize $E(a + x)$. Using the approximation to E which it has been just derived, It will be differentiated with respect to x and equate with zero, yielding

$$\nabla_x E(a + x) = J^T e + J^T J x = 0$$

solving this equation for x yields the Gauss-Newton update, and gives the algorithm for one iteration of Gauss-Newton ICP:

1. Compute the vector of residuals $e(a_k)$, and its $N_d \times p$ matrix of derivatives J with respect to the components of a . (For a 2D rigid-body transformation, a has 3 components, and J is $N_d \times 3$).
2. Compute the update $x = (J^T J)^{-1} J^T e$.
3. Set $a_{k+1} = a_k + x$.

Of course, the above strategy does not guarantee that the step taken will result in a reduced error at $E(a_{k+1})$. Whether or not it does so depends on the accuracy of the second-order Taylor series expansion at a_k , and on the validity of the Gauss-Newton approximation. However, it can be shown that when these approximations are good, as they tend to be when near the minimum, convergence is rapid and reliable.

By comparison, an accelerated gradient descent approach as used by some previous registration algorithms is obtained by replacing step 2 with

2. Compute the update $x = \lambda^{-1} J^T e$.

Where the value of λ controls the distance travelled along the gradient direction. For small λ , the iteration moves a long way along the downhill direction; large λ implies a short step. In contrast to Gauss-Newton, gradient descent does guarantee to reduce E , providing λ is sufficiently large. However, its convergence near the optimum is dismally slow.

The Levenberg-Marquardt algorithm combines both updates in a relatively simple way in order to achieve good performance in all regions. Step 2 is replaced by

2. Compute the update $x = (J^T J + \lambda I)^{-1} J^T e$.

Now large λ corresponds to small, safe, gradient-descent steps, while small λ allows fast convergence near the minimum. The art of a good Levenberg-Marquardt implementation is in tuning λ after each iteration to ensure rapid progress even where the Gauss-Newton approximations are poor.

4.1.4 Robust estimation

Many attempts have been made to widen the basin of convergence of the ICP algorithm, and these largely amount to introducing robust estimation. This is difficult with standard ICP, as no closed-form robust estimate of the T step is known, so authors have used either a nonlinear or RANSAC-based estimator, or exclude (Winsorise) points with large errors. With LM-ICP, it is trivial to modify the error function to include a robust kernel. One must take a little care to ensure that the Levenberg-Marquardt algorithm behaves well if the kernel chosen is not smooth.

For LM-ICP either of the following kernels could be used:

$$\text{Lorentzian : } \epsilon(r) = \log\left(1 + \frac{r^2}{\sigma}\right) \quad \text{or } \text{Huber : } \epsilon(r) = \begin{cases} x & r < 0 \\ 2\sigma|r| - \sigma^2 & \text{else} \end{cases}$$

LM-ICP using these kernels is compared against ICP with Winsorised residuals, as this is the most common way of robustifying ICP. Figure 4.2 shows that LM-ICP with the Huber kernel has a basin of convergence twice as large as that of Winsorised ICP.

Figure 4.2 shows an example of these different methods. For the example shown in Figure 4.2. b) different basins of convergence is shown in Figure 4.2. b). Algorithms are initialised at one-degree intervals between $\pm 120^\circ$ of the true solution, and the value of the minimum is plotted as a function of initial guess. LM-Huber is significantly wider than all others, at a cost of a factor of 2 in number of iterations.

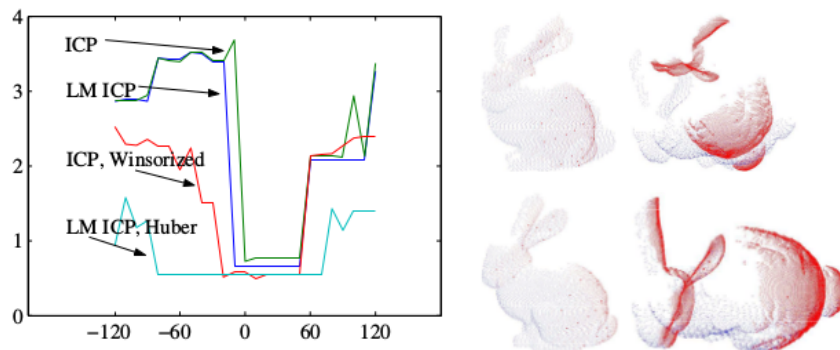


Figure 4.2. (a) 2D Basins of convergence. (b) 3D Example. Top: Initial alignment for 3D registration. Bottom: LM-Huber optimum.

4.1.5 EM(Expectation Maximization)

The vast majority of state-of-the-art registration techniques select one of the sets as the “model” and perform pairwise alignments between the other sets and this set. The main drawback of this mode of operation is

that there is no guarantee that the model-set is free of noise and outliers, which contaminates the estimation of the registration parameters. The method based on EM treats all the point sets on an equal footing: they are realizations of a Gaussian mixture (GMM) and the registration is cast into a clustering problem. An EM algorithm that estimates both the GMM parameters and the rotations and translations that map each individual set onto the “central” model is formally derived. The mixture means play the role of the registered set of points while the variances provide rich information about the quality of the registration [34,52] (Figure 4.3).

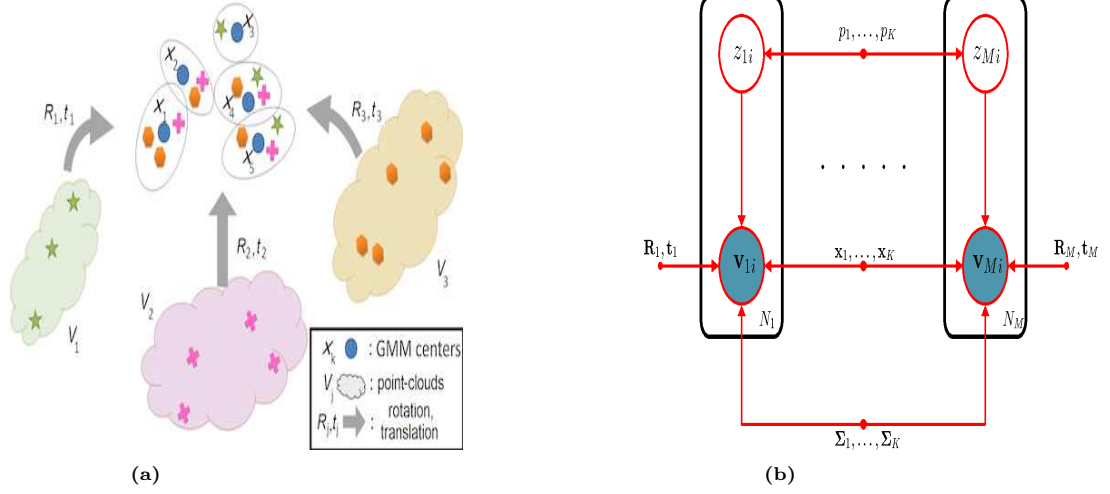


Figure 4.3. The proposed generative model for joint registration of multiple point clouds (a) and the associated graphical model (b). Unlike pairwise registration strategies, the proposed model simultaneously registers an arbitrary number of point clouds with partial or total overlap and optimally estimates both the GMM and registration parameters. Hence, the solution is not biased towards a particular cloud.

4.1.5.1 Problem Formulation

Let $V_j = [v_{j1} \dots v_{ji} \dots v_{jN_j}]$ be a $R^{3 \times N_j}$ matrix of N_j points associated with the j -th point set and let M be the number of sets. The union of all the data points is denoted with $V = V_{j=1}^M$. It is assumed that there is a rigid transformation $\phi_j : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ that maps a point set j onto a scene-centered model. The objective is to estimate the set-to-scene transformations under the constraint that the sets are jointly registered. It is assumed that the point sets are rigidly-transformed realizations of an unknown “central” GMM. Hence, one can write

$$P(\mathbf{v}_{ji}) = \sum_{k=1}^K p_k \mathcal{N}(\phi_j(\mathbf{v}_{ji}) | \mathbf{x}_k, \Sigma_k) + p_{K+1} \mathcal{U}(a-b), \quad (4.1.0.2)$$

where $\phi_j(\mathbf{v}_{ji}) = \mathbf{R}_j \mathbf{v}_{ji} + \mathbf{t}_j$ (a 3×3 rotation matrix \mathbf{R}_j and a 3×1 translation vector \mathbf{t}_j), p_k are the mixing coefficients $\sum_{k=1}^{K+1} p_k = 1$, \mathbf{x}_k and Σ_k are the means and covariance matrices, and \mathcal{U} is the uniform distribution parameterized by $a-b$. Here we take $a-b = h$, where h is the volume of the 3D convex hull encompassing the data. Now γ as the ratio between outliers and inliers is defined, that is,

$$p_{K+1} = \gamma \sum_{k=1}^K p_k. \quad (4.1.0.3)$$

This allows to balance the outlier/inlier proportion by choosing γ . To summarize, the model parameters are

$$\Theta = (\{p_k, \mathbf{x}_k, \Sigma\}_{k=1}^K, \{\mathbf{R}_j, \mathbf{t}_j\}_{j=1}^M). \quad (4.1.0.4)$$

The deterministic nature of ϕ_j does not affect the statistical properties of the mixture model. Figure 4.3 shows a graphical representation of the proposed model. This problem can be solved in the framework of expectation-maximization. In particular, hidden variables $\mathcal{Z} = \{z_{ji} | j = 1 \dots M, i = 1 \dots N_j\}$ such that $z_{ji} = k$ assigns observation $\phi_j(\mathbf{v}_{ji})$ to the k -th component of the mixture model could be defined, and maximizing the expected complete-data log-likelihood is aimed

$$\mathcal{E}(\Theta|\mathbf{V}, Z) = \mathbb{E}_{\mathcal{Z}}[\log P(\mathbf{V}, \mathcal{Z}; \Theta)|\mathbf{V}] = \sum_{\mathcal{Z}} P(\mathcal{Z}|\mathbf{V}, \Theta) \log(P(\mathbf{V}, \mathcal{Z}; \Theta)) \quad (4.1.0.5)$$

in order to estimate the parameters Θ .

4.1.5.2 Joint Multiple-Set Registration

Assuming that the observed data V are independent and identically distributed, it is straightforward to write Eq. 4.1.0.5 as

$$\mathcal{E}(\Theta, \mathbf{V}|, Z) = \sum_{i,j,k} \alpha_{jik} \left(\log p_k + \log P(\phi_j(\mathbf{v}_{ji})|z_{ji} = k; \Theta) \right) \quad (4.1.0.6)$$

where $\alpha_{jik} = P(z_{ji} = k|v_{ji}; \Theta)$ are the posteriors. By replacing the standard expressions of the likelihoods and by ignoring constant terms, (1.1.5) can be written as an objective function of the form

$$\begin{aligned} f(\Theta) = & \frac{-1}{2} \sum_{j,i,k} \alpha_{jik} \left(\|\phi_j(\mathbf{v}_{ji}) - \mathbf{x}_k\|_{\Sigma_k}^2 + \log |\Sigma_k| - 2 \log p_k \right) \\ & + \log p_{K+1} \sum_{j,i} \alpha_{ji(K+1)} \end{aligned} \quad (4.1.0.7)$$

where $|\cdot|$ denotes the determinant and $\|y\|_A^2 = y^T A^{-1} y$. Therefore, one has to solve the following constrained optimization problem:

$$\begin{cases} \max_{\Theta} & f(\Theta) \\ \text{s.t.} & \mathbf{R}_j^T \mathbf{R}_j = \mathbf{I}, |\mathbf{R}_j| = 1, \forall j = 1 \dots M. \end{cases} \quad (4.1.0.8)$$

Direct optimization of $f(\Theta)$ via a closed-form solution is difficult owing to the induced non-linearities. Therefore, an expectation conditional maximization (ECM) scheme is adopted to solve (4.1.0.8). ECM is more broadly applicable than EM, while it is well suited for this problem owing to the extended parameter set. Notice that ECM replaces the M-step of EM with a series of conditional maximization (CM) steps, that is, an M-substep for each parameter. It has been referred to this algorithm as joint registration of multiple point clouds (JR-MPC); JR-MPC maximizes $f(\Theta)$, and hence $(\Theta|\mathbf{V}, Z)$, sequentially with respect to each parameter, by clamping the remaining ones to their current values. Commonly, such an iterative process leads to a stepwise maximization of the observed-data likelihood as well. At each iteration, it first estimates the transformation parameters, given the current GMM parameters, and then it estimates the new GMM parameters, given the new transformation parameters. It is of course possible to adopt a reverse order, in particular when a rough alignment of the point sets is provided. However, no prior information on the rigid transformations is considered, so that the pre-estimation of the registration parameters favors the estimation of the GMM means, x_k , that should be well distributed in space.

4.1.6 Mixture Models and Point Matching

The main idea is, to measure the similarity between two finite point sets by considering their continuous approximations. In this context, one can relate a point set to a probability density function [53]. Considering the point set as a collection of Dirac Delta functions, it is natural to think of a finite mixture model as representation of a point set. As the most frequently used mixture model, a Gaussian mixture is defined as a convex combination of Gaussian component densities $\Phi(x|\mu_i, \Sigma_i)$, where μ_i is the mean vector and Σ_i is the covariance matrix. The probability density function is explicitly given as $p(x) = \sum_{i=1}^k w_i \Phi(x|\mu_i, \Sigma_i)$ where w_i are weights associated with the components. If the number of components, k , is quite large, then almost any density may be well approximated by this model. Here, the Gaussian mixture model is used to represent the point set explicitly [55]. In a simplified setting, the number of components is the number of the points in the set. And for each component, the mean vector is given by the location of each point. Without prior information, it can be assumed each component has same weight and each Gaussian is spherical i.e. the covariance is proportional to identity matrix. If the the orientation and anisotropy information are available, the shape and orientation of the covariance matrix can be determined accordingly. For example, when the point set is acquired from an intensity image, additional information can be obtained from the gradient vectors. For a dense point cloud, a mixture model-based clustering or grouping may be performed as a preprocessing procedure. the two point-sets, model and scene, are represented by two mixtures of Gaussians. Intuitively, if these two point sets are aligned properly

enough, the two resulting mixtures should be statistically similar to each other. Consequently, this raises the key problem: How to measure the similarity/closeness between two Gaussian mixtures?

4.1.6.1 L_2 Distance between Gaussian Mixtures

Many measures have been proposed to quantify the similarity between two arbitrary probability distributions[55]. Here we suggest the L_2 distance for measuring similarity between Gaussian mixtures because: (1) The L_2 distance is strongly related to the inherently robust estimator L_2E ; (2) There is a closed-form expression for the L_2 distance between Gaussian mixtures, which in turn permits efficient implementation of the registration algorithm.

Density Power Divergence—Let f and g be density functions, define the divergence $d_\alpha(f, g)$ to be

$$d_\alpha(f, g) = \int \left\{ \frac{1}{\alpha} f^{1+\alpha} - \left(1 + \frac{1}{\alpha} f g^\alpha \right) + g^{1+\alpha} \right\} dz \quad (4.1.0.9)$$

In the case of $\alpha \rightarrow 0$:

$$d_0(f, g) = \lim_{\alpha \rightarrow 0} d_\alpha(f, g) = \int f(z) \log\{f(z)/g(z)\} \quad (4.1.0.10)$$

which gives the well known Kullback-Leibler (KL) divergence. And the minimizer corresponds to maximum likelihood estimation (MLE). On the [integraltext] other hand, when $\alpha = 1$, the divergence $d_1(f, g) = \int \{f(z) - g(z)\}^2 dz$ becomes exactly the L_2 distance between the densities, and the corresponding estimator is called L_2E estimator. For general $0 < \alpha < 1$, the class of density power divergences provides a smooth bridge between the KL divergence and the L_2 distance. Furthermore, this single parameter α controls the trade-off between robustness and asymptotic efficiency of the parameter estimators which are the minimizers of this family of divergences. The fact that the L_2E is inherently superior to MLE in terms of robustness can be well explained by viewing the minimum density power divergence estimators as a particular case of Mestimators [56].

Next, one can easily derive the closed-form expression for the L_2 distance between two mixtures of Gaussians by noting the formula:

$$\int \Phi(x|\mu_1, \Sigma_1) \Phi(x|\mu_2, \Sigma_2) dx = \Phi(0|\mu_1 - \mu_2, \Sigma_1 + \Sigma_2) \quad (4.1.0.11)$$

The Eq. 4.1.0.11 does not have a closed form for $0 < \alpha < 2$ except for the L_2 distance at $\alpha = 1$. Hence, this affords an advantage to the L_2 distance since there is no need to the numerical integration or approximation which is a practical limitation not only in computation time but also for obtaining sufficient accuracy to perform numerical optimization.

4.2 Non-rigid point set registration

Given two point sets, non-rigid registration yields a non-rigid transformation which maps one point set to the other. Non-rigid transformations include affine transformations such as scaling and shear mapping. However, in the context of point set registration, non-rigid registration typically involves nonlinear transformation. If the eigenmodes of variation of the point set are known, the nonlinear transformation may be parametrized by the eigenvalues [27]. A nonlinear transformation may also be parametrized as a thin plate spline.

Compared with rigid transforms, non-rigid transforms such as affine, spline or radial functions can well-represent the shape variations of 3D surfaces. The Thin Plate Spline transform (TPST) [57] is a widely used non-rigid transform for 3D registration. It decomposes the deformation between two subjects into affine and non-affine components. The famous TPS-RPM method proposed by Chui and Rangarajan [58] incorporates TPS into the framework of ICP for matching two point sets. They iteratively performed a soft-assign and deterministic annealing optimization to compute point correspondence and used a binary correspondence matrix to record point matching. This method is sensitive to initial alignment and is not fit for the dense registration of 3D points because running TPS with the whole dense point sets as control points is impractical [59]. Some people suggest using control points to overcome this problem.

Hutton et al. [60] manually chose 9 facial landmarks as the control points to compute the TPS deformation between two 3D points.

There is a probabilistic method for point set registration called the Coherent Point Drift (CPD) method [27]. Similar to [58], given two point sets, a GMM is fit to the first point set, whose Gaussian centroids are initialized from the points in the second set. However, unlike [58,55] which assumes a thin-plate spline transformation, no explicit assumption of the transformation model will be made. Instead, the process of adapting the Gaussian centroids from their initial positions to their final positions as a temporal motion process will be considered, and

impose a motion coherence constraint over the velocity field. Velocity coherence is a particular way of imposing smoothness on the underlying transformation. The concept of motion coherence was proposed in the Motion Coherence Theory [61]. The intuition is that points close to one another tend to move coherently. This motion coherence constraint penalizes derivatives of all orders of the underlying velocity field (thin-plate spline only penalizes the second order derivative). Examples of velocity fields with different levels of motion coherence for different point correspondences are illustrated in Figure 4.4.

In [27] has been shown that CPD outperforms RPM in the presence of noise and outliers.

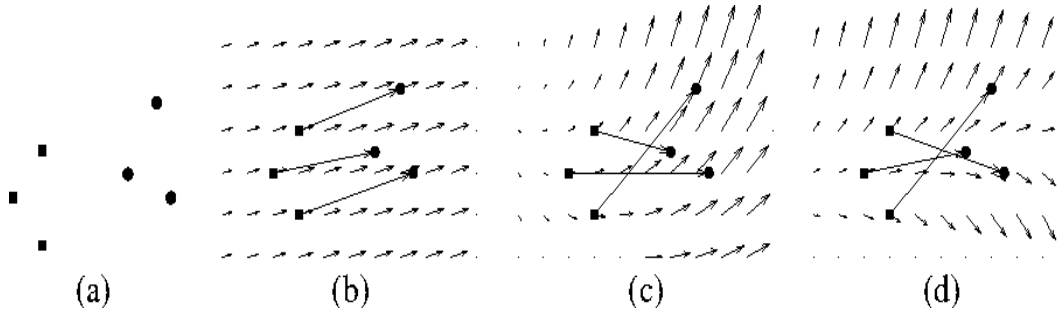


Figure 4.4. (a) Two given point sets. (b) A coherent velocity field. (c, d) Velocity fields that are less coherent for the given correspondences.

4.2.1 Coherent Point Drift: Method

Assume two point sets are given, where the template point set $\mathbf{Y} = (y_1, \dots, y_M)^T$ (expressed as a $M \times D$ matrix) should be aligned with the reference point set $\mathbf{X} = (x_1, \dots, x_N)^T$ (expressed as a $N \times D$ matrix) and D is the dimension of the points [27]. The points in \mathbf{Y} as the centroids of a Gaussian Mixture Model will be considered, and will be fit to the data points \mathbf{X} by maximizing the likelihood function. \mathbf{Y}_0 as the initial centroid positions will be denoted and a continuous velocity function v will be defined for the template point set such that the current position of centroids is defined as $\mathbf{Y} = v(\mathbf{Y}_0) + \mathbf{Y}_0$.

Consider a Gaussian-mixture density $p(x) = \sum_{m=1}^M \frac{1}{M} p(x|m)$ with $x|m \sim \mathcal{N}(y_m, \sigma^2 \mathbf{I}_D)$, where \mathbf{Y} represents D -dimensional centroids of equally-weighted Gaussians with equal isotropic covariance matrices, and \mathbf{X} set represents data points. In order to enforce a smooth motion constraint, the prior $p(\mathbf{Y}|\lambda) \propto \exp(\frac{\lambda}{2} \phi(\mathbf{Y}))$ will be defined, where λ is a weighting constant and $\phi(\mathbf{Y})$ is a function that regularizes the motion to be smooth. Using Bayes theorem, goal is to find the parameters \mathbf{Y} by maximizing the posteriori probability, or equivalently by minimizing the following energy function:

$$E(\mathbf{Y}) = - \sum_{n=1}^N \log \sum_{m=1}^M e^{-\frac{1}{2} \left\| \frac{x_n - y_m}{\sigma} \right\|^2} + \frac{\lambda}{2} \phi(\mathbf{Y}) \quad (4.2.0.1)$$

Assume the data i.i.d. and ignore terms independent of \mathbf{Y} . Eq. 4.2.0.1 has a similar form to that of Generalized Elastic Net (GEN), which has shown good performance in non-rigid image registration; note that there \mathbf{Y} is directly penalized, while here the transformation v is penalized. The ϕ function represents prior knowledge about the motion, which should be smooth. Specifically, the velocity field v generated by template point set displacement should be smooth. Smoothness is a measure of the “oscillatory” behavior of a function. Within the class of differentiable functions, one function is said to be smoother than another if it oscillates less; in other words, if it has less energy at high frequency. The high frequency content of a function can be measured by first high-pass filtering the function, and then measuring the resulting power. This can be represented as $\phi(v) = \int_{R^d} |\tilde{v}(\mathbf{s})|^2 / \tilde{G}(\mathbf{s}) d\mathbf{s}$, where \tilde{v} indicates the Fourier transform of the velocity and \tilde{G} is some positive function that approaches zero as $\|\mathbf{s}\| \rightarrow \infty$. Here \tilde{G} represents a symmetric low-pass filter, so that its Fourier transform G is real and symmetric. Following this formulation, the energy function is rewriting as:

$$E(\tilde{v}) = - \sum_{n=1}^N \log \sum_{m=1}^M e^{-\frac{1}{2} \left\| \frac{x_n - y_m}{\sigma} \right\|^2} + \frac{\lambda}{2} \int_{R^d} (|\tilde{v}(\mathbf{s})|^2 / \tilde{G}(\mathbf{s})) d\mathbf{s} \quad (4.2.0.2)$$

It can be shown using a variational approach that the function which minimizes the energy function in Eq. (4.2.0.2) has the form of the radial basis function:

$$v(\mathbf{z}) = \sum_{n=1}^M w_n G(\mathbf{z} - y_{0n}) \quad (4.2.0.3)$$

Substituting the solution obtained in Eq. (4.2.0.3) back into Eq. (4.2.0.2), we obtain

$$E(\mathbf{W}) = - \sum_{n=1}^N \log \sum_{m=1}^M e^{-\frac{1}{2} \left\| \frac{x_n - y_{0m} - \sum_{k=1}^M w_k \mathbf{G}(y_{0k} - y_{0m})}{\sigma} \right\|^2} + \frac{\lambda}{2} \text{tr}(\mathbf{W}^T \mathbf{G} \mathbf{W}) \quad (4.2.0.4)$$

where $\mathbf{G}_{M \times M}$ is a square symmetric Gram matrix with elements $g_{ij} = e^{-\frac{1}{2} \left\| \frac{y_{0i} - y_{0j}}{\beta} \right\|^2}$ and $\mathbf{W}_{M \times D} = (\mathbf{w}_1, \dots, \mathbf{w}_M)^T$ is a matrix of the Gaussian kernel weights in Eq. (4.2.0.3).

Optimization. Following the EM algorithm derivation for clustering using Gaussian Mixture Model, the upper bound of the function in Eq. 4.2.0.4 can be found as (E-step):

$$E(\mathbf{W}) = \sum_{n=1}^N \log \sum_{m=1}^M P^{old}(m|x_n) \frac{\|x_n - y_{0m} - \mathbf{G}(m, \cdot) \mathbf{W}\|^2}{2\sigma^2} + \frac{\lambda}{2} \text{tr}(\mathbf{W}^T \mathbf{G} \mathbf{W}) \quad (4.2.0.5)$$

where P^{old} denotes the posterior probabilities calculated using previous parameter values, and $\mathbf{G}(m, \cdot)$ denotes the m^{th} row of \mathbf{G} . Minimizing the upper bound Q will lead to a decrease in the value of the energy function E in Eq. 4.2.0.4, unless it is already at local minimum. Taking the derivative of Eq. (4.2.0.5) with respect to \mathbf{W} , and rewriting the equation in matrix form, it will obtain (M-step)

$$\frac{\delta Q}{\delta \mathbf{W}} = \frac{1}{2} G(\text{diag}(\mathbf{P}\mathbf{1}))(\mathbf{Y}_0 + \mathbf{G}\mathbf{W}) - \mathbf{P}\mathbf{X} + \lambda \mathbf{G}\mathbf{W} = 0 \quad (4.2.0.6)$$

where \mathbf{P} is a matrix of posterior probabilities. The $\text{diag}(\cdot)$ notation indicates diagonal matrix and $\mathbf{1}$ is a column vector of all ones.

Solving the system for \mathbf{W} is the M-step of EM algorithm. The E step requires computation of the posterior probability matrix \mathbf{P} . The EM algorithm is guaranteed to converge to a local optimum from almost any starting point. Eq. (4.2.0.6) can also be obtained directly by finding the derivative of Eq. 4.2.0.4 with respect to \mathbf{W} and equating it to zero. This results in a system of nonlinear equations that can be iteratively solved using fixed point update, which is exactly the EM algorithm shown above. The computational complexity of each EM iteration is dominated by the linear system of Eq. (4.2.0.6), which takes $O(M^3)$. If using a truncated Gaussian kernel and/or linear conjugate gradients, this can be reduced to $O(M^2)$.

4.3 Summery

We have analysed different point set registration methods in this chapter. While there are many registration methods has been developed by now, they still have the problem of accuracy and robustness. ICP [29] is one of the most general algorithm proposed in this field. Other methods like CPD are for more specific applications like non-rigid registration in the presence of noise and outliers.

Chapter 5

Multi Step Registration

In the previous chapter (Point set registration), different methods of point registration (Rigid or Non-Rigid) were discussed. To be able to use these methods for multi view c-elegans registration, a multi step registration method was proposed. Different methods mentioned before were used and the one with the least error was LMICP+ICP huber (chapter 4 section 1.2 and section 1.2.1). The result of feature selection approach was the input to multi-step registration. Feature selection approach suggested some assignment of nuclei that were improved during registration. In other words, this method did the assignment and mapping simultaneously. After registration the common nuclei were obtained, could be used to register 2 views in one step. Registration of first step was done in 3 steps: registration of centers, nuclei and embryos. For the method evaluation we used 5 different approaches. Evaluation of proposed method were also done on 3 different synthetic data and one different data set.

5.1 5.1 Multi-step multi-view registration of c-elegans embryo by LMICP+ICP huber

5.1.1 Step 1: Registration of control points or centres after feature selection

At the first step control points (or centers after feature selection) are given to the LMICP+ICP huber algorithm (running consecutively). The result of registration is the transformation, transformed points, and mean distance between centres. We find that the Mean distance between centers after registration is 19.86 pixels (Table 5.1). If the transformation matrix obtained at this step is applied to the initial state of the embryos, the nuclei which are common can not be completely overlapp (Figure 5.1).

5.1.2 Step 2: Rejecting the false correspondences by Trimming or Ransac

LMICP+ICP huber has been robusted due to its formula explained in previous chapter (section 4.1.4). This robustness is useful when data is simple or when we just need to register the centres.

Descriptors (or nuclei centers) composed of only four fiducials are not completely distinctive and similar descriptors can occur by chance.

To have most accurate multi view registration, all true correspondences consent on one transformation model, In this case each false correspondence support a different transformation. Therefore, using the minimal descriptor size (4 fiducials) and rejected false correspondences from candidate sets with the Random Sample Consensus (RANSAC) or Trimming on the affine transformation model followed by robust regression is necessary.

This approach is useful when number of nuclei centers are ideally 1,000–2,000 fiducials per imaged volume[5]. Here we propose to use RANSAC or Trimming on nuclei to have more features than the position of centres. Due to computational challenges here, RANSAC is avoided as much as it is necessary and trimming is used.(previous chapter, section 4.1.2).

As explained in that section, at this step, the nuclei which their centers are the closest are chosen to find an affine transformation. A nucleus has more features than the position of centres, so instead of using just centres for registration, the whole nuclei can be used to find an affine transformation. To have an affine transformation based on nuclei at least 4 nuclei are needed. With this number of nuclei, the rmse obtained with the LMICP method is 11.21 pixels, which is less than the maximum diameter of a nucleus (15.5 pixels). Since the goal is to have a robust affine transformation rather 3 times of the minimum number should be used is 12 nuclei. At the first step a non-precise transformation was obtained. This transformation is applied to the embryos and the 12

closest nuclei are chosen for the second step. This makes the closest nuclei closer and improves overlap between common nuclei. The transformation is obtained with mean distance between centres of nuclei of 20 pixels (Table 5.1). The common nuclei got closer and uncommon nuclei got farther from each other (Figure 5.1c). The area between head and tail in the embryo is an area that can be chosen as a common area by eyes and could be used to show the progress of registration (Figure 5.1d).

Step 3: Alignment of the embryo after Trimming by LM-ICP Huber

Step 2 was used to improve the overlap between common nuclei. This cause to lose alignment between embryos and non-overlap nuclei are not in the frame of common embryos.

In every step after finding the transformation matrix, it is applied to second view to obtain transformed of second view embryo. LMICP+ICP huber registration was used to align reference and 2 times (from previous steps) transformed of second view embryo at this step. The mean distance between centres of nuclei is 14.8 pixels (Table 5.1). This error is less than mean of diameters of nuclei (which is 17.5 based on the histogram of diameters of nuclei, S1Fig) in two embryos. After this step not only nuclei overlap got better (Figure 5.2D) the embryos got aligned in 3D space (Figure 5.1d). Figure 5.2 shows the difference between alignment in different steps.

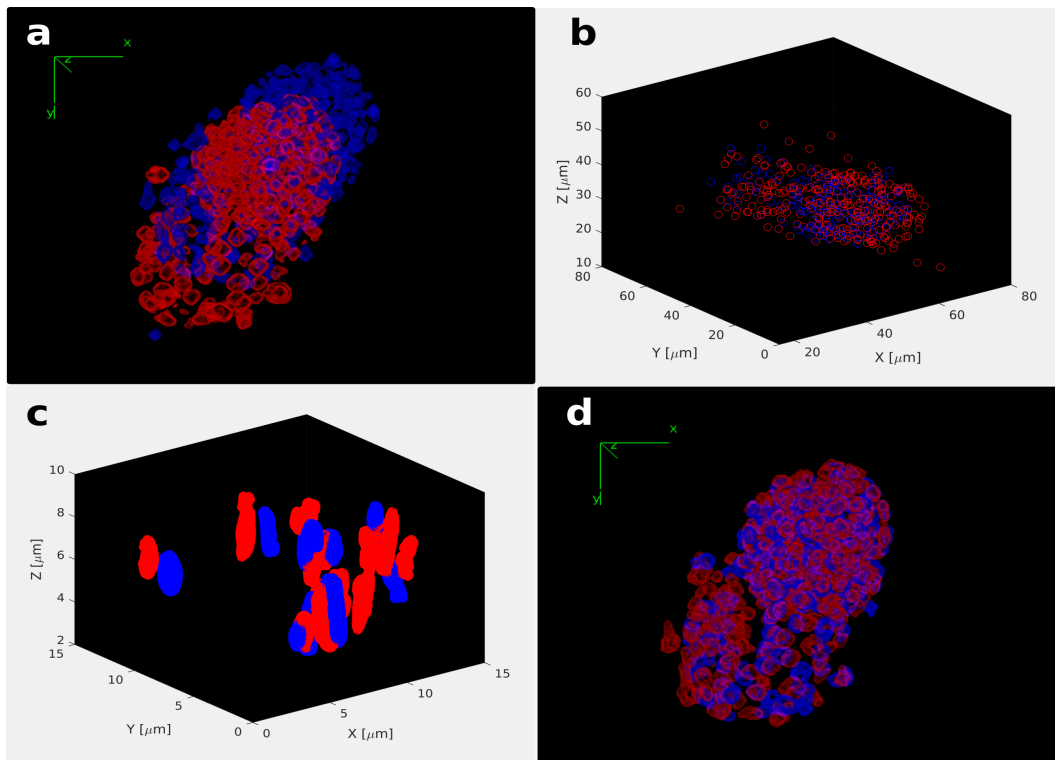


Figure 5.1. The Overlap of the reference view (red) and the transformed view (blue) result of: a)initial state b)step 1: Registration of centers after feature selection, c)step 2: Rejecting the false correspondences nuclei by Ransac, d)step3: Alignment of embryos after Ransac. (a) and (d) are overlap of embryos. (b) center of nuclei and (c) 12 closest nuclei. 3D views of embryos are volume rendering of segmented nuclei [60].

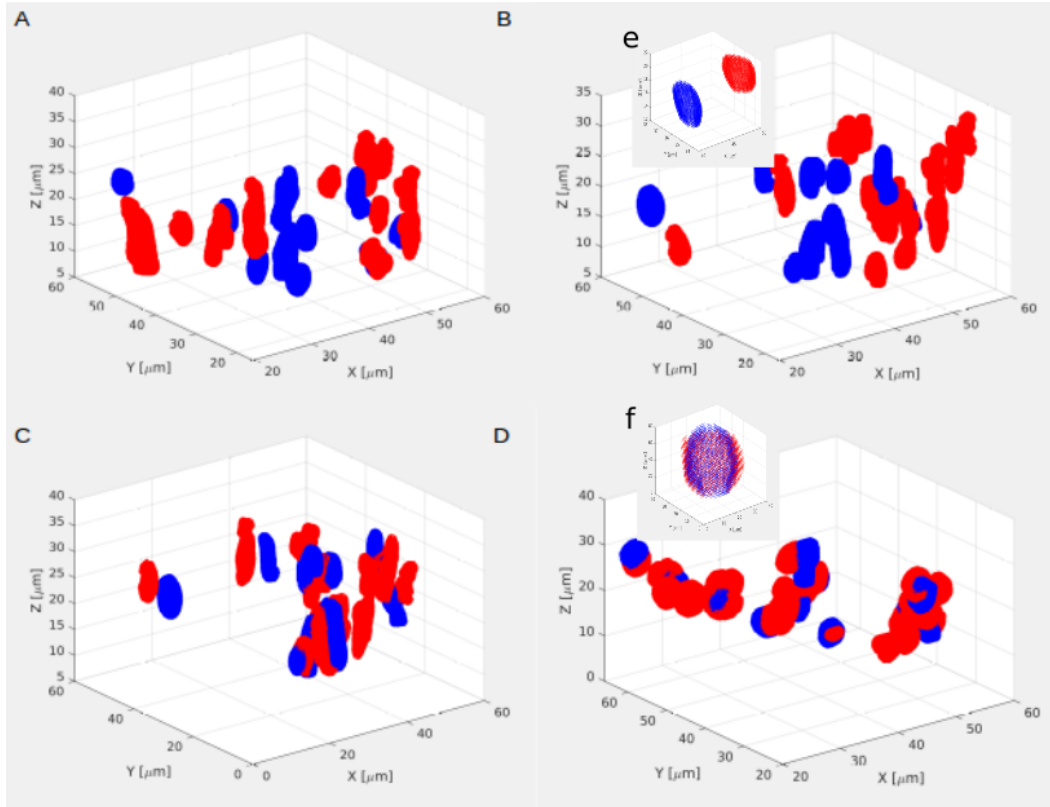


Figure 5.2. The Overlap of the reference view (red) and the transformed view (blue) of nuclei. A)after feature selection B)step 1, C)step 2, D)step3. The progressive overlap of nuclei in different step shows the accuracy of registration. e,f) one pair of nuclei chosen from relative figures.

Table 5.1. Mean of euclidean distance between all nuclei centres of embryos at each step for Multi-step registration

	Step 1	Step 2	Step 3
Multi-Step Registration	19.12	27.00	14.8

Feature Correlation

Histogram of diameters of nuclei in x,y and z direction is calculated for both views (S1Fig A,B). The peak value for diameter of nuclei in x,y and z direction for the reference view are 23.7, 15.7 and 15.7 pixels and for the second view is 17.3, 17.3 and 16.3 pixels.

The average value that could be considered as diameter of nuclei for fused image is 17.5 $((15.7+17.3)/2)$ pixels. Since this value is smaller than the peak of distance between nuclei in 2 views, it is a valid measurement (S1Fig D). After registration, nuclei which have distances less than 17.5 pixels ($2.45 \mu m$) between their centres have been chosen as common or overlap nuclei. The number of common nuclei found by multi-step registration is 325 nuclei with the mean error of euclidean distance between nuclei 11.39 pixels (Table 5.2).

Table 5.2. Quantification of result of registration

	Overlap nuclei	Non-overlap nuclei	Mean error of euclidean distance
Multi-Step Registration	325	60	11.39

To validate the result features of these nuclei are calculated such as: spatial positions, volume, dot product of first eigenvectors, first eigenvalue and eccentricities of 2 views of two methods after registration (Figure 5.3). To compare these feature for different methods the error function e is used.

$$e_i = \frac{|x_{1i} - x_{2i}|}{x_{1i} + x_{2i}}$$

(5.1.0.1)

which x_{1i} is the value of feature of i -th reference nuclei and x_{2i} the value of feature of i -th transformed nuclei of the second view after multi step registration. This error function makes is easier to compare all features together and also the units of measurement is not necessary to be used. For dot product minus of log is calculated to have a better consistency in comparing all the features together.

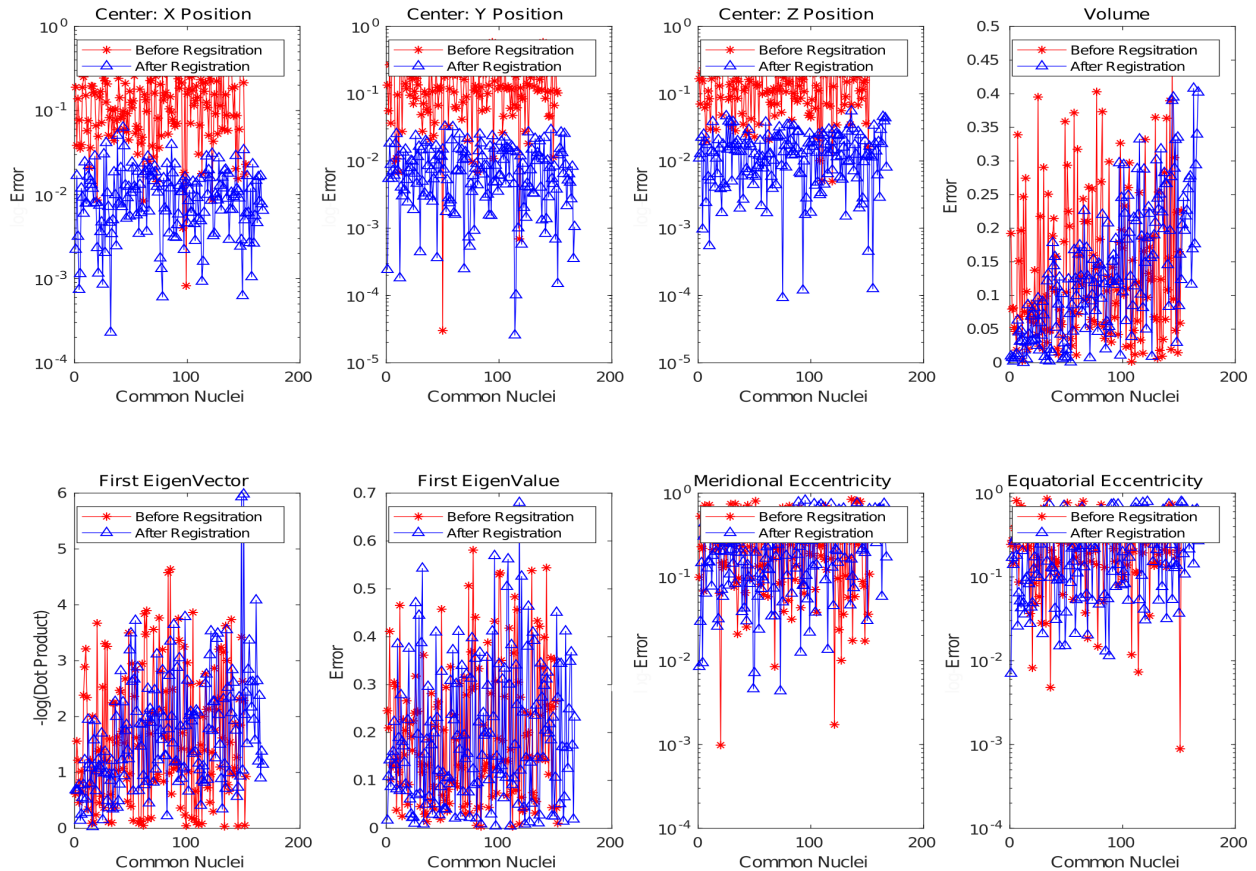


Figure 5.3. 8 features of overlap nuclei after registration for multi-step registration before and after registration. Spatial positions, volume, dot product of first eigenvectors, first eigenvalue and eccentricities are calculated. The log of differences between these values are based on the error function shown above (except than volume and first eigenvalue). Smaller the values are more similar the nuclei are.

To compare the result of two sets the mean of each plot is displayed in Table 5.3. The euclidean distance between nuclei for these data set after each step of registration is shown in Table 5.4. First step is registering the centers as initial step. The second step is to use Ransac or trimming for registering nuclei, so the best match nuclei will be closer to each other but distance between non-overlap nuclei will be worse. The third step is to correct this distance based on shape of embryo. When number of non-overlap nuclei are more than overlap nuclei the third distance value will be larger otherwise it will be smaller. (multi-step registration: 325 overlap, 60 non-overlap)

Table 5.3. Mean of Log difference of feature values for data set1 before and after registration plotted in Figure 5.3

	Before Registration	After Registration
Center: X Position	0.13	0.011
Center: Y Position	0.17	0.0091
Center: Z Position	0.16	0.015
Volume	0.13	0.13
First Eigen Vector	1.63	1.60
First Eigen Value	0.19	0.19
Meridional Eccentricity	0.29	0.27
Equatorial Eccentricity	0.32	0.29

In the second and third columns values are calculated for the same number of nuclei which were used for feature selection. Table 5.3 enable us to compare the performance of feature selection. The value of feature parameters after registration are either close or less than the value before registration. Registration were successful in overlapping similar nuclei and even improved the result of feature selection part.

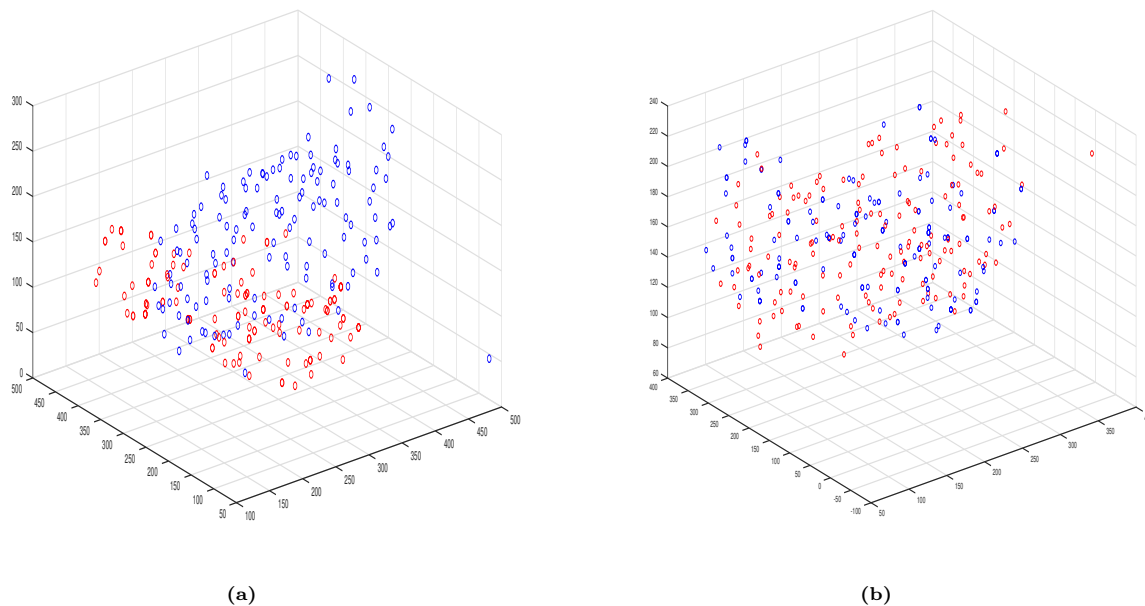
5.2 Evaluation of registration for synthetic data 1

To evaluate the method above a synthetic data 1 is made. The proposed method was used on this data set and. Result of registration is quantified in Tables 5.4 and 5.5 and Figures 5.5.

Transformation

It has three rotation around -x, -y and -z direction. Jitter was added to centers and outline of nuclei. Twenty nuclei were removed/added to data too.

This data is registered first by center registration (Figure 5.4) and then by multi-step registration.

**Figure 5.4.** The Overlap of the reference view and the transformed reference view nuclei centres a)before registration, b)after registration.

Feature Correlation

After registration, nuclei which have distances less than 12.6 pixels ($1.8 \mu\text{m}$) (based on the histogram of diameters of nuclei.) between their centres have been chosen as common or overlap nuclei. The number of common nuclei

found by multi-step registration for synthetic data 1 is 134 nuclei with the mean error of euclidean distance between nuclei 1.8 pixels and for centres registration 42 nuclei with the mean error of euclidean distance between nuclei 10.00 pixels (Figure 5.5, 5.6, S4Fig).

Features of these nuclei such as: spatial positions, volume, dot product of first eigenvectors, first eigenvalue and eccentricities of 2 views of two methods after registration are calculated (Figure 5.5).

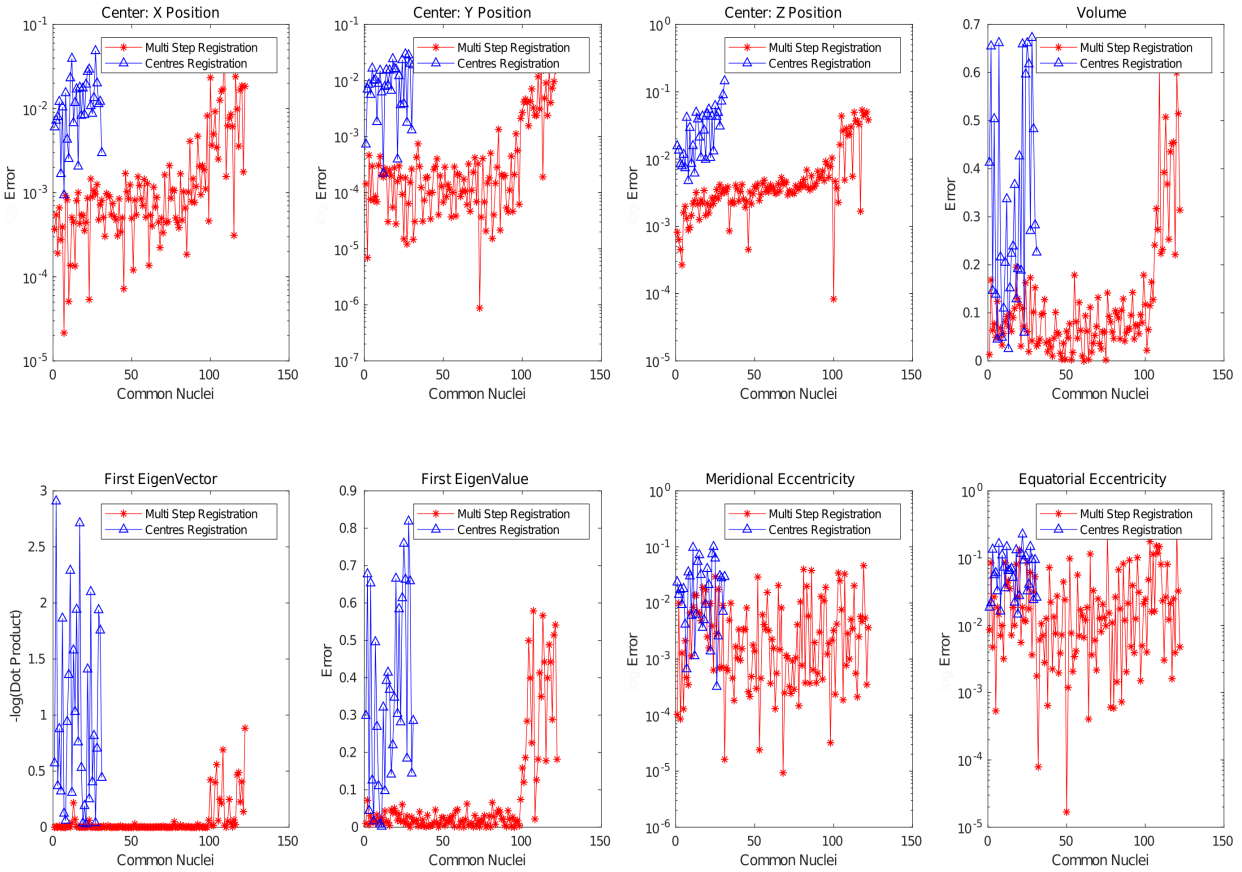


Figure 5.5. 8 features of overlap nuclei after and before registration. Spatial positions, volume, dot product of first eigenvectors, first eigenvalue and eccentricities are calculated. The log of differences between these values based on the error function are shown above (except than volume and first eigenvalue). Smaller the values are more similar the nuclei are.

To compare the result of two sets the mean of each plot is displayed in Table 5.4.

Table 5.4. Mean of Log difference of features for syntethic data2 based on centres and multi step registration plotted in Figure 5.5

	Centres Regsitratio	Multi-Step Registration
Center: X Position	0.010	0.0029
Center: Y Position	0.010	0.0015
Center: Z Position	0.03	0.0079
Volume	0.32	0.11
First Eigen Vector	0.98	0.05
First Eigen Value	0.35	0.07
Meridional Eccentricity	0.02	0.006
Equatorial Eccentricity	0.07	0.03

The euclidean distance between nuclei for these data set after each step of registration is shown in Table 5.5. First step is registering the centers as initial step. The second step is to use Ransac or trimming for registering nuclei,so the best match nuclei will be closer to each other but distance between non-overlap nuclei will be worse. The third step is to correct this distance based on shape of embryo . When number of non-overlap nuclei are more than overlap nuclei the third distance value will be larger otherwise it will be smaller.(centre registration : 42 overlap, 88 non-overlap)(multi-step registration: 134 overlap, 24 non-overlap)

Table 5.5. Mean of euclidean distance between all nuclei centres of embryos at each step for 2 methods

	Step 1	Step 2	Step 3
Centres Registration	19.12	30.66	7.12
Multi-Step Registration	19.122	-	-

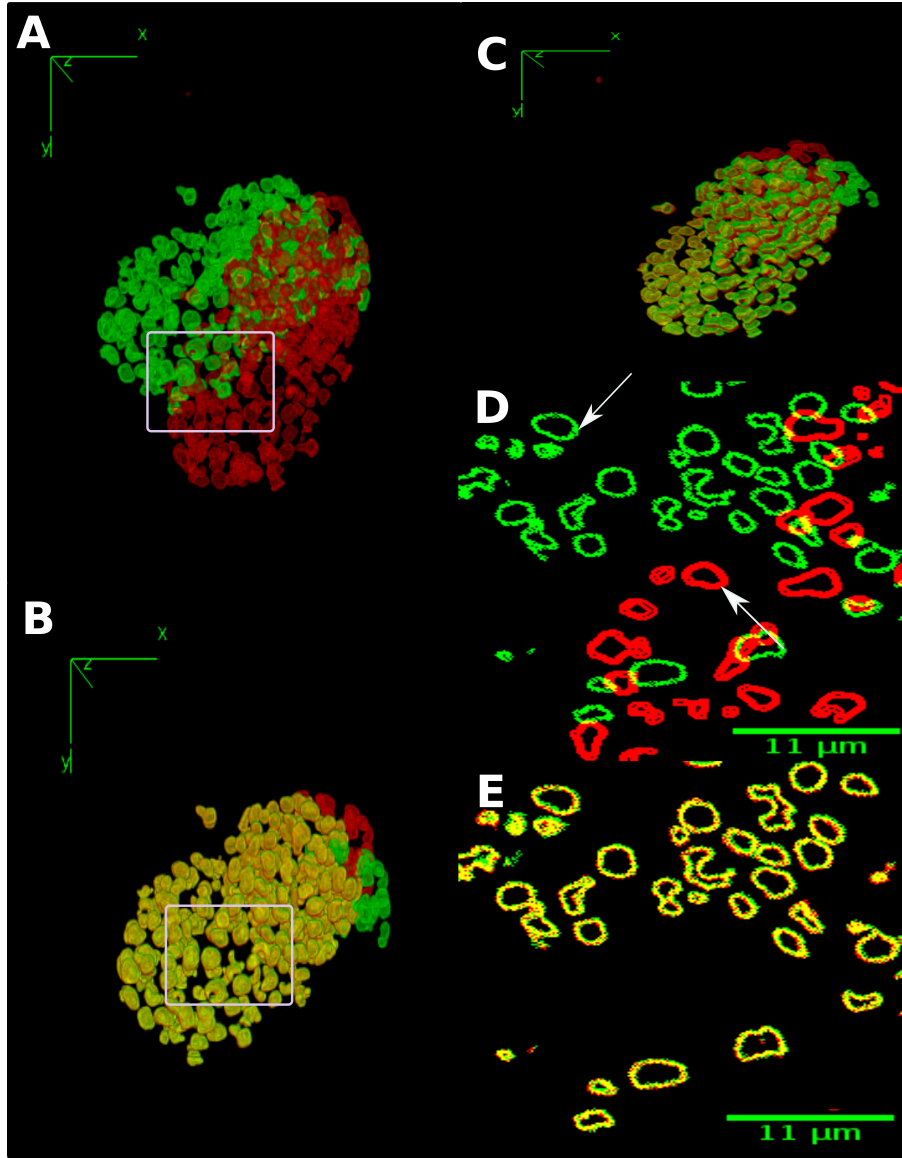


Figure 5.6. The Overlap of the reference view and the transformed reference view. A) before registration, B) after multi-step registration, C) after centre registration. All nuclei are matched except than 20 ± 1 nuclei which were added/removed. (D)-(E) 2D contour of the reference view nuclei (green) and transformed reference view nuclei (red). The overlap of nuclei is shown by yellow. The arrows show the same nuclei with and without jitters.

5.3 Evaluation of registration for synthetic data 2

To register the different embryos, embryos first should be segmented (or labeled) to extract the centres. After that feature selection part is used to extract features. Multi step registration is the third step in this pipeline. To remove the errors of other part of the pipeline, the synthetic data 2 is made. For synthetic data 2 transformation is applied to centres not embryo. It means here we just evaluate the last two parts of pipeline. The proposed method was used on this data set and. Result of registration is quantified in Tables 5.6 and 5.7 and Figure 5.9.

Applying transformation to centres after feature selection

For synthetic data 2 transformation is applied to centres not embryo. It has three rotation around $-x$, $-y$ and $-z$ direction. Jitter was added to centres and outline of nuclei. Twenty centres were removed/added to data too. This data is registered first by center registration (Figure 5.7) and then by multi-step registration.

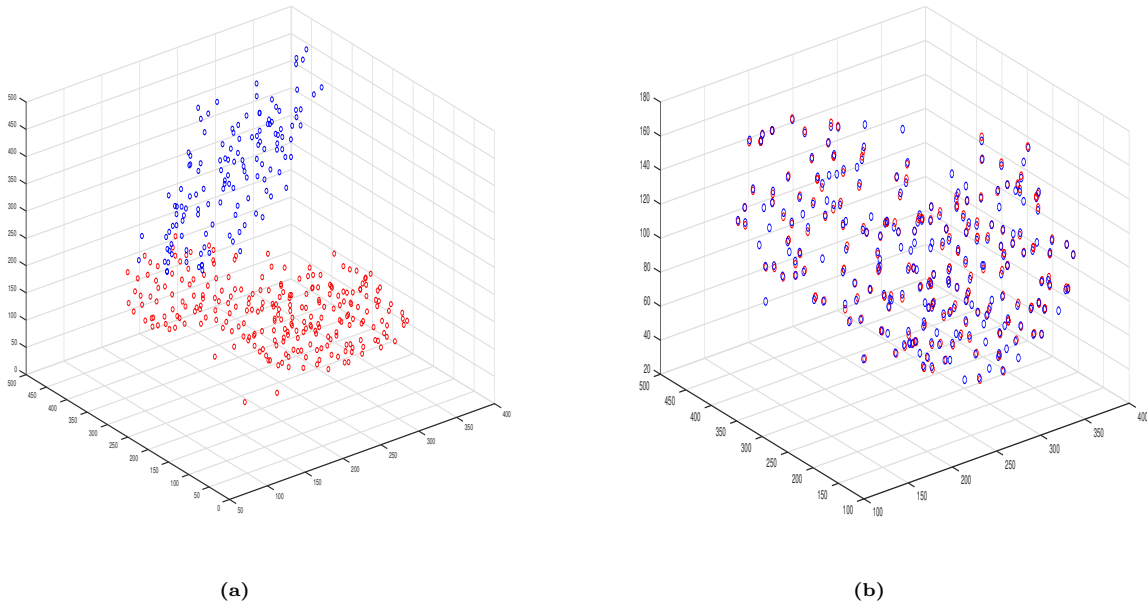


Figure 5.7. The Overlap of the reference view and the transformed reference view nuclei centres a) before registration, b) after registration.

The 3D view of embryos after registration is shown in Figure 5.8.

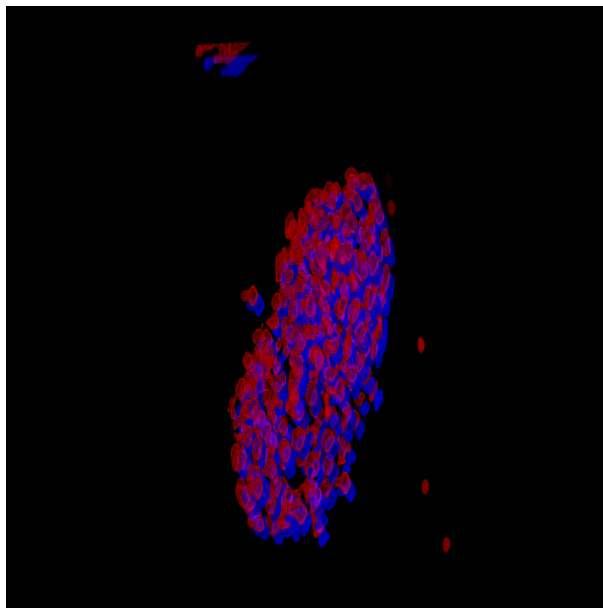


Figure 5.8. 3D view of overlap of the reference view and the registered transformed reference view nuclei.

Feature Correlation

After registration, nuclei which have distances less than 12.6 pixels ($1.8 \mu\text{m}$) (based on the histogram of diameters of nuclei.) between their centres have been chosen as common or overlap nuclei. The number of common nuclei found by multi-step registration for synthetic data 2 is 134 nuclei with the mean error of euclidean distance between nuclei 1.8 pixels and for centres registration 149 nuclei with the mean error of euclidean distance between nuclei 4.75 pixels (Figure 5.9).

Features of these nuclei such as: spatial positions, volume, dot product of first eigenvectors, first eigenvalue and eccentricities of 2 views of two methods after registration are calculated (Figure 5.9).

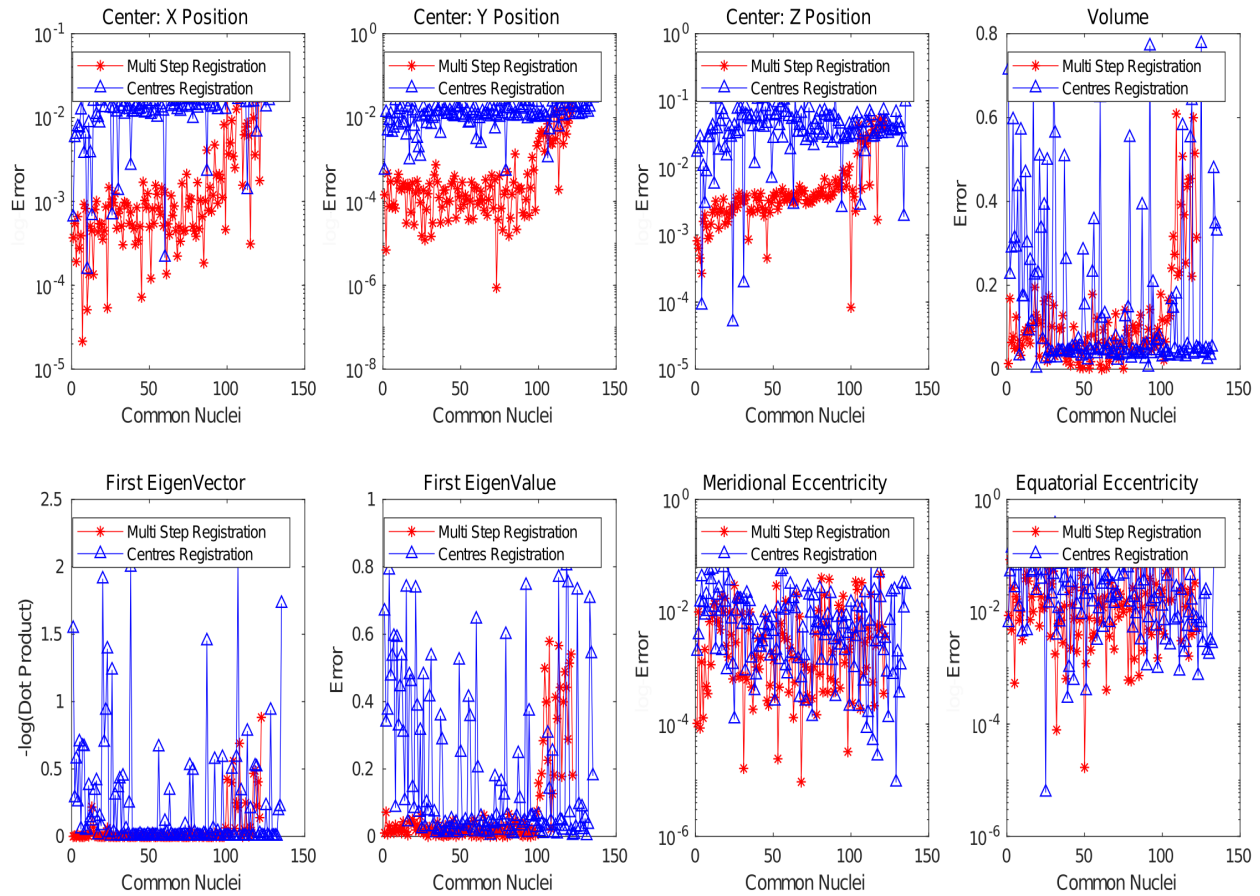


Figure 5.9. 8 features of overlap nuclei after registration for multi-step registration and centre registration. Spatial positions, volume, dot product of first eigenvectors, first eigenvalue and eccentricities are calculated. The log of differences between these values based on the error function are shown above (except than volume and first eigenvalue). Smaller the values are more similar the nuclei are.

To compare the result of two sets the mean of each plot is displayed in Table 5.6.

Table 5.6. Mean of Log difference of features for syntethic data2 based on centres and multi step registration plotted in Figure 5.9

	Centres Regsitratio	Multi-Step Registration
Center: X Position	0.010	0.0029
Center: Y Position	0.010	0.0015
Center: Z Position	0.03	0.0079
Volume	0.23	0.11
First Eigen Vector	0.7	0.05
First Eigen Value	0.20	0.07
Meridional Eccentricity	0.01	0.006
Equatorial Eccentricity	0.08	0.03

The euclidean distance between nuclei for these data set after each step of registration is shown in Table 5.7. First step is registering the centers as initial step. The second step is to use Ransac or trimming for registering nuclei,so the best match nuclei will be closer to each other but distance between non-overlap nuclei will be worse. The third step is to correct this distance based on shape of embryo . When number of non-overlap nuclei are more than overlap nuclei the third distance value will be larger otherwise it will be smaller.(centre registration : 149 overlap, 3 non-overlap)(multi-step registration: 134 overlap, 24 non-overlap)

Table 5.7. Mean of euclidean distance between all nuclei centres of embryos at each step for 2 methods

	Step 1	Step 2	Step 3
Centres Registration	19.12	30.66	7.12
Multi-Step Registration	6.75	-	-

5.4 Applying transformation to centres before feature selection: Synthetic data 3

For Synthetic data 3 transformation is applied to centres before feature selection not embryo. This means just the third part of the registration pipeline is evaluated here. The proposed method was used on this data set. Result of registration is quantified in Tables 5.8, 5.9 and Figure 5.12.

Data has three rotation around -x ,-y and -z direction. Jitter was added to centers and outline of nuclei. Twenty centres were removed/added to data too.

This data is registered first by center registration (Figure 5.10) and then by multi-step registration.

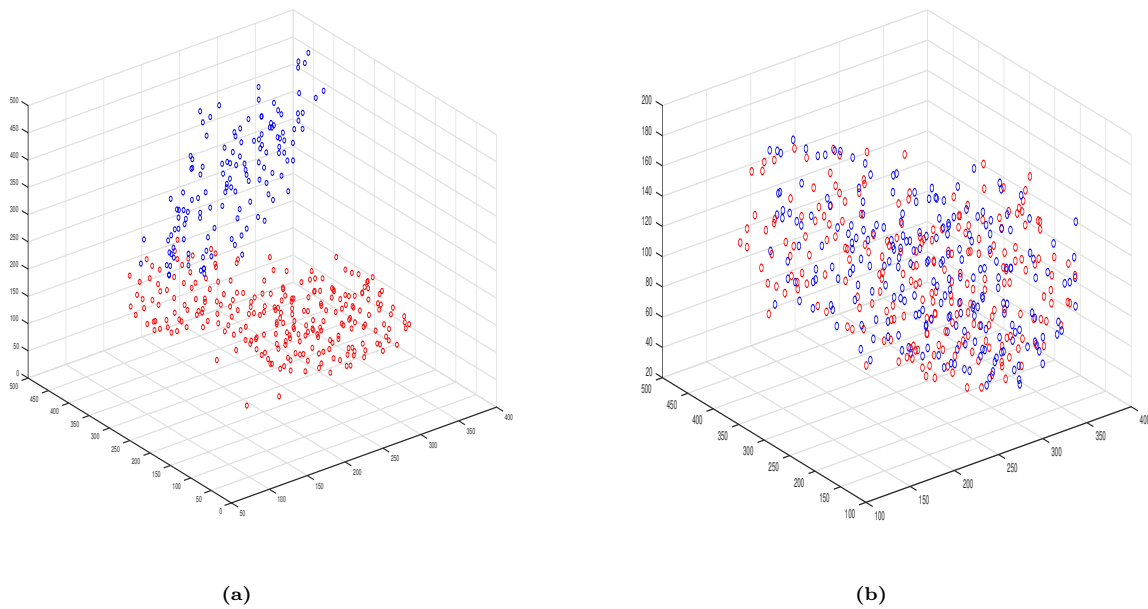


Figure 5.10. The Overlap of the reference view and the transformed reference view nuclei centres a) before registration, b) after registration.

The 3D view of embryos after registration is shown in Figure 5.11.

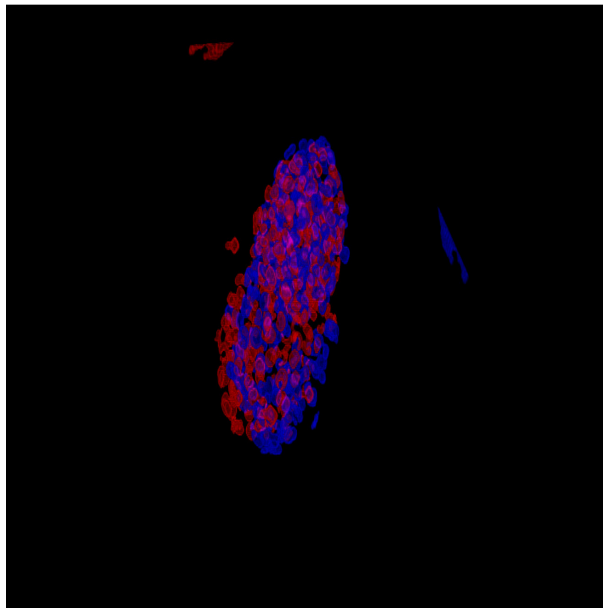


Figure 5.11. 3D view of overlap of the reference view and the registered transformed reference view nuclei.

Feature Correlation

After registration, nuclei which have distances less than 12.6 pixels ($1.8 \mu\text{m}$) (based on the histogram of diameters of nuclei.) between their centres have been chosen as common or overlap nuclei. The number of common nuclei found by multi-step registration for synthetic data 2 is 134 nuclei with the mean error of euclidean distance

between nuclei 1.8 pixels and for centres registration 109 nuclei with the mean error of euclidean distance between nuclei 9.35 pixels (Figure 5.12).

Features of these nuclei such as: spatial positions, volume, dot product of first eigenvectors, first eigenvalue and eccentricities of 2 views of two methods after registration are calculated (Figure 5.12).

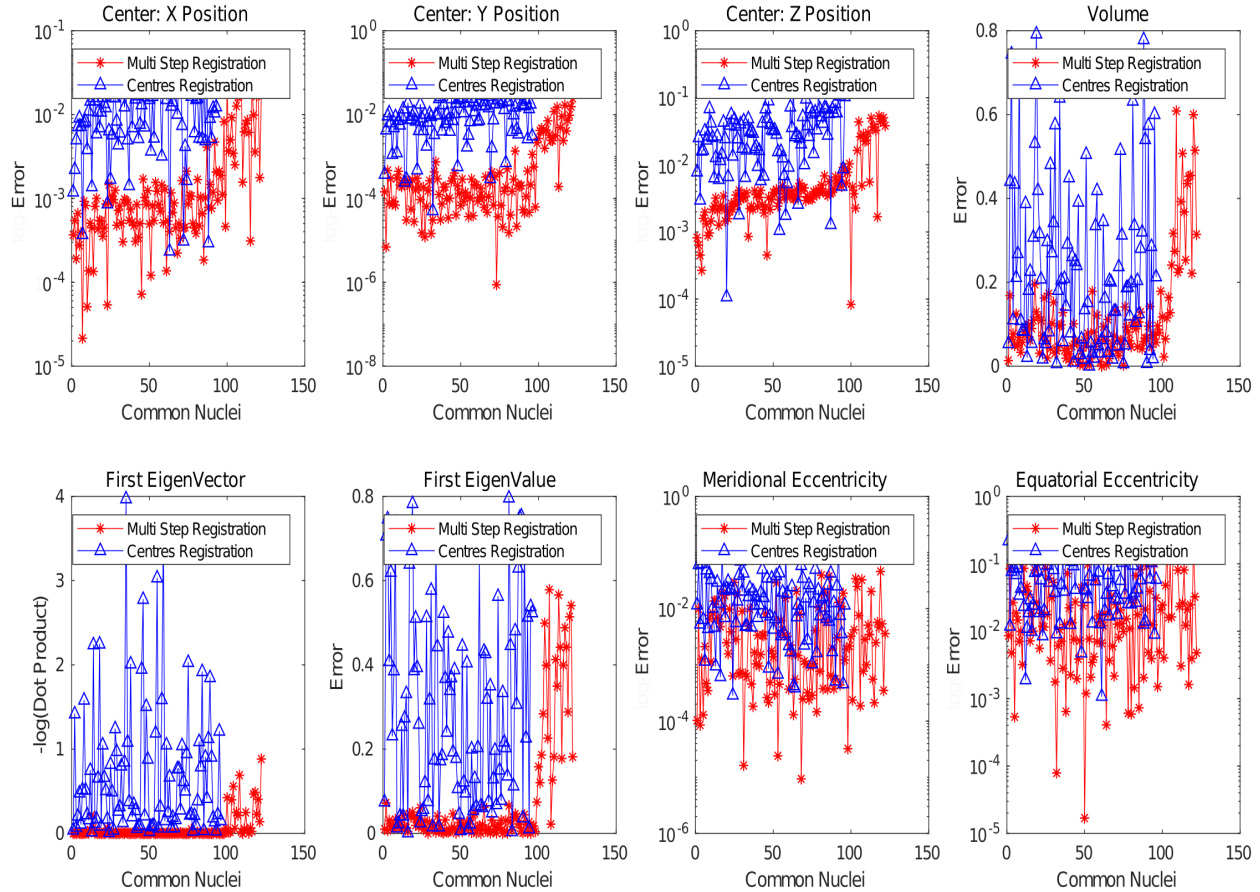


Figure 5.12. 8 features of overlap nuclei after registration for multi-step registration and centre registration (with out feature selection). Spatial positions, volume, dot product of first eigenvectors, first eigenvalue and eccentricities are calculated. The log of differences between these values based on the error function are shown above (except than volume and first eigenvalue). Smaller the values are more similar the nuclei are.

To compare the result of two sets the mean of each plot is displayed in Table 5.8.

Table 5.8. Mean of Log difference of features for synthetic data2 based on centres and multi step registration plotted in Figure 5.12

	Centres Registration	Multi-Step Registration
Center: X Position	0.010	0.0029
Center: Y Position	0.010	0.0015
Center: Z Position	0.04	0.0079
Volume	0.16	0.11
First Eigen Vector	0.22	0.05
First Eigen Value	0.18	0.07
Meridional Eccentricity	0.01	0.006
Equatorial Eccentricity	0.06	0.03

The euclidean distance between nuclei for these data set after each step of registration is shown in Table 5.9. (centre registration : 109 overlap, 38 non-overlap)(multi-step registration : 134 overlap, 24 non-overlap)

Table 5.9. Mean of euclidean distance between all nuclei centres of embryos at each step for 2 methods

	Step 1	Step 2	Step 3
Centres Registration	19.12	30.66	7.12
Multi-Step Registration	16.19	-	-

5.5 Method Evaluation

In this step multi step registration of c-elegans embryo (centers, nuclei, embryos) is done by five methods: ICP, LMICP, EM, Rigid-RPM and CPD.

Euclidean distance between nuclei at each step is calculated to compare the performance of algorithms.

After registration, nuclei which have more than 50% overlap have been chosen and the features of these nuclei: spatial positions, volume, orientations, principle axes length and eccentricities of 2 sets are calculated and compared.

The euclidean distance between nuclei at each step for ICP, LMICP and EM algorithms are (21, 13.8, 13.5), (4.3, 35.11, 11.67), (29, 32, 31) pixels respectively. Figure 5.13 shows the comparison of features of overlap nuclei after registration of 2 sets. Comparison with other methods (Coherent point drift and Rigid robust point matching) which provide non-rigid transformation is also done (Figure 5.14, 5.15).

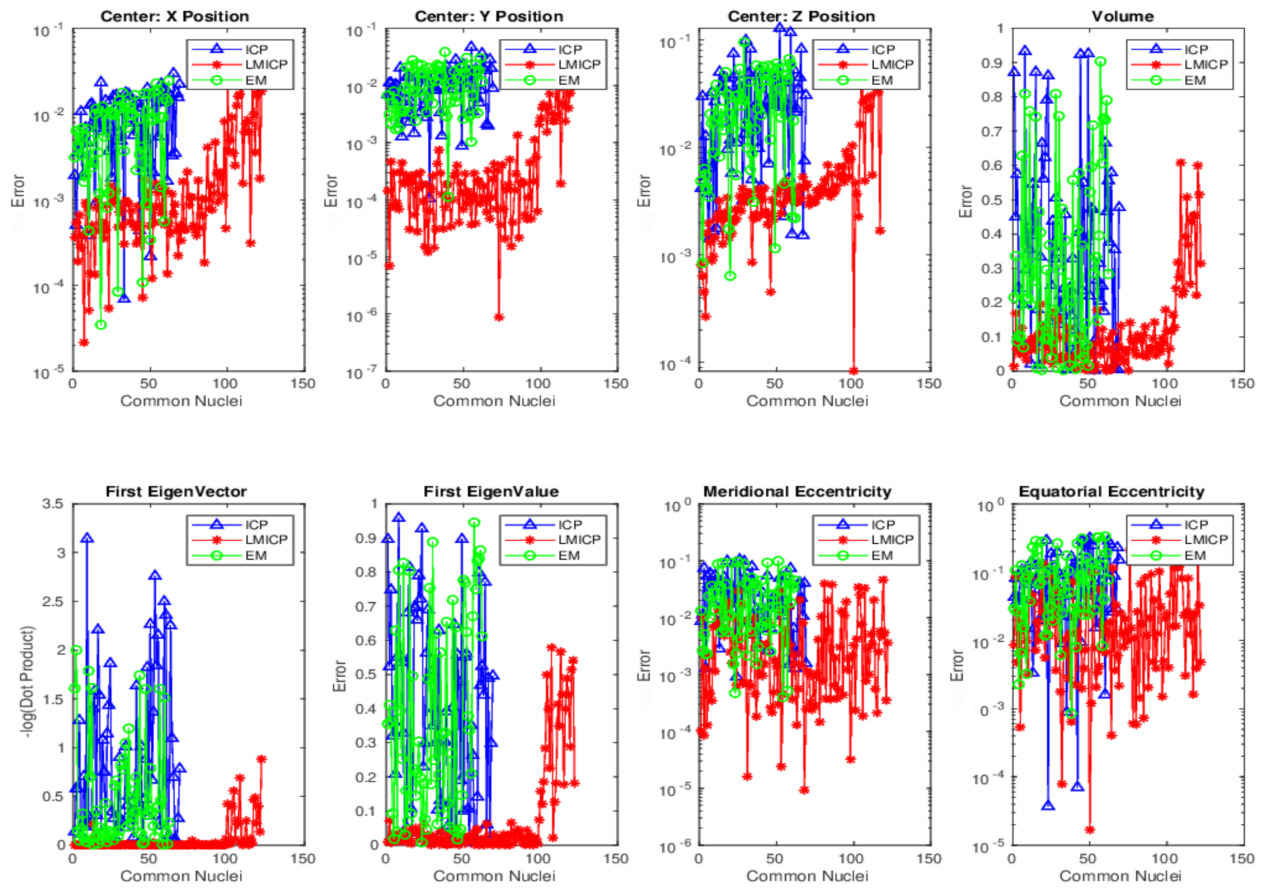


Figure 5.13. 8 features of overlap nuclei after registration for multi-step registration done by ICP, LMICP huber and EM methods. Spatial positions, volume, dot product of first eigenvectors, first eigenvalue and eccentricities are calculated. The log of differences between these values based on the error function are shown above (except than volume and first eigenvalue). Smaller the values are more similar the nuclei are.

Rigid-RPM

The euclidean distance between nuclei at each step for this algorithm are 29, 32, 31 pixels. Figure 5.14 shows the comparison of features of overlap nuclei after registration of 2 sets.

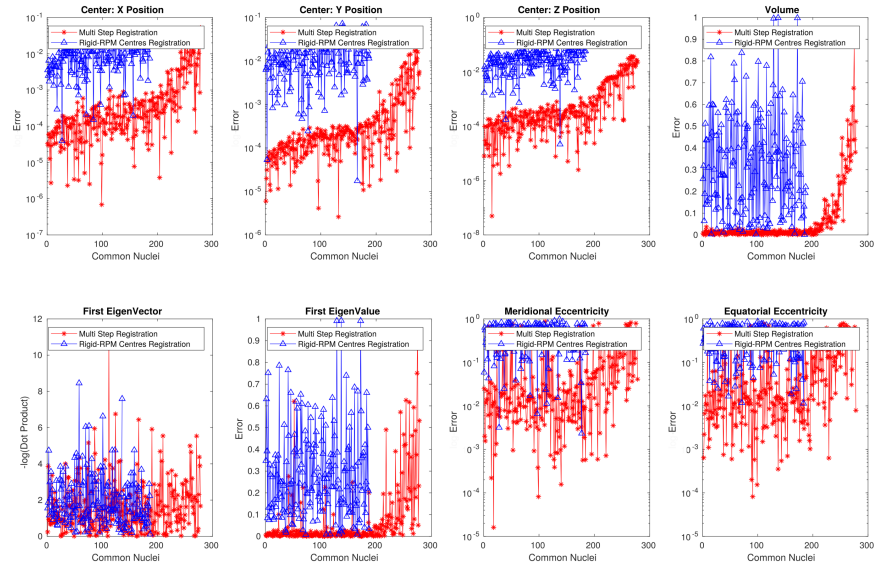


Figure 5.14. 8 features of overlap nuclei after Rigid-RPM registration: spatial positions, volume, orientations, principle axes length and eccentricities are calculated. The log of differences between these value are shown above. Smaller the values are more similar the nuclei are.

CPD

The euclidean distance between nuclei centres is 6.34 pixels. Figure 5.15 shows the comparison of features of overlap nuclei after registration of 2 sets.

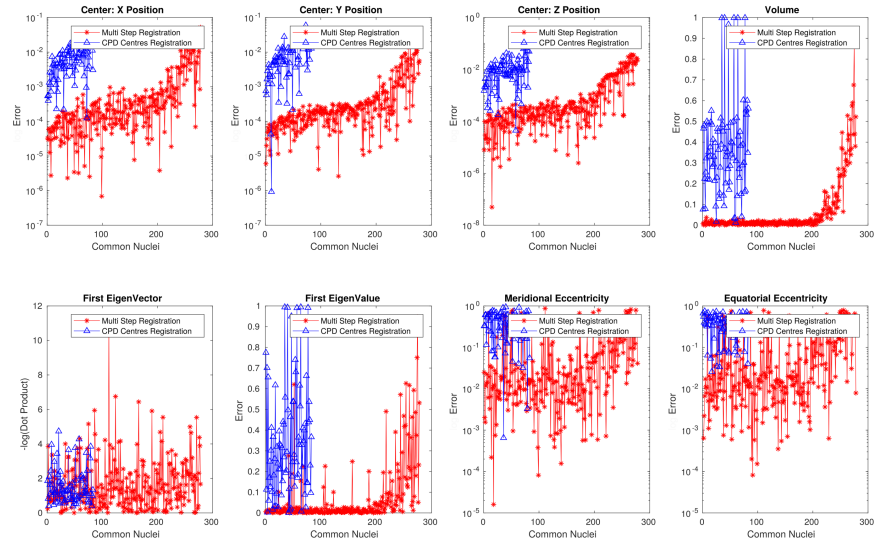


Figure 5.15. 8 features of overlap nuclei after CPD registration: spatial positions, volume, orientations, principle axes length and eccentricities are calculated. The log of differences between these value are shown above. Smaller the values are more similar the nuclei are.

5.6 Evaluation of registration of a different set

To evaluate multi step registration on real data, other data sets from c-elegans embryos were used. These sets or stacks are from the same embryo as the first set but at different angles. This data set contained different views of the head part of c-elegans embryos.

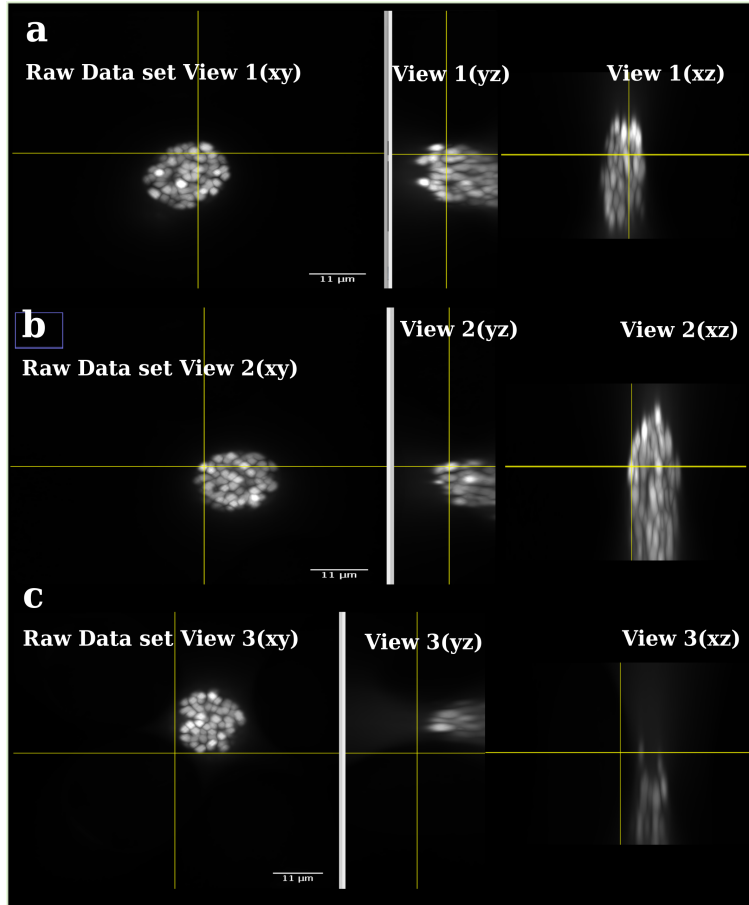


Figure 5.16. The Orthogonal views of raw data sets from a spinning-disk confocal microscopy of C-elegans head. a)The reference view, b)The second view, c)The third view at depth $z=80$ with YZ views and XZ views. The Orthogonal views of data displays the XZ and YZ planes at a given point in the 3D image. The intersection of the yellow lines on the stack indicating the point in the stack that is being analysed. the intersection of the yellow lines on the stack indicates the point in the stack that is being analysed.

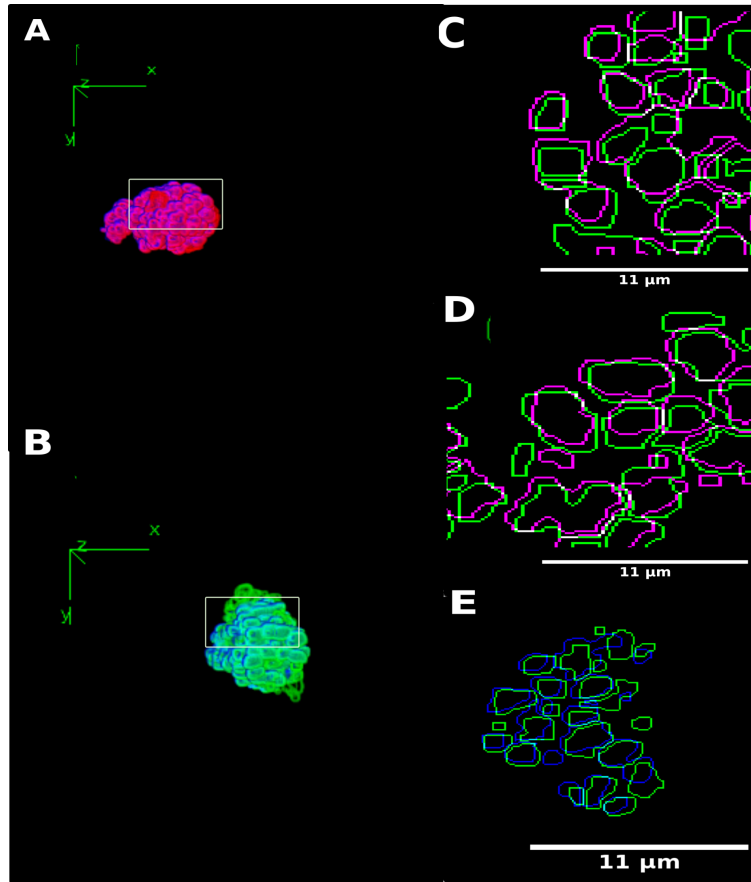


Figure 5.17. The Overlap of the reference view and the fused view of reference view and the second view of C-elegans head. First row 2D outline of reference view and the fused view. Pink colour shows fused view and green colour reference view. Common nuclei are shown by white. b) Green colour shows the second view. c) blue colour shows the third view. Second row: 3D view of overlap stack images. d) result of fusion step. e) result of groupwise registration step.

5.7 Summery

In this chapter, registration method was explained. We applied the proposed method on different synthetic data and real data. Quantification of the result of registration are shown in different figures and tables.

Chapter 6

Fusion

6.1 Fusion and Groupwise registration

6.1.1 Fusion

Image fusion is the combination of two or more different images to form a new image by using a certain algorithm [61]. In general, fusion techniques can be classified into different levels. They are signal level, pixel/data level, feature level, and decision level [62,63]. During the past two decades, several fusion techniques have been proposed. Most of these techniques are based on the compromise between the desired spatial enhancement and spectral consistency. Among the hundreds of variations of image fusion techniques, the widely used methods include, but are not limited to, intensity-hue-saturation (IHS) [64], high-pass filtering, principal component analysis (PCA) [65], different arithmetic combination (e.g. Brovey transform), multi-resolution analysis-based methods (e.g. pyramid algorithm, wavelet transform) [66,67], and Artificial Neural Networks (ANNs) [68], etc.

In our case, the fusion of two embryos is done by comparing the position of centres of nuclei in two views. If the centres are closer than the error of registration they are considered as one and won't be added to the fused embryo. To be able to do this KNN search is used to find the closest pair centres in two views. The result then will be sorted based on the distance between centres (Figure 6.1). Since nuclei have irregular shapes, another approach is added to the fusion part. The overlap of 2 (pair) closest nuclei will be calculated. If the overlap of smaller nuclei is more than 50% of its volume, the nuclei will be considered as overlap.

6.1.2 Groupwise registration

To use this registration for a group of images, all of them will be registered with respect to the reference view. Here we have 6 views of the embryo that need to be registered.

Groupwise registration has a similar structure like a reverse tree decision. For 6 views the registrations are pairwise and one fused image is obtained. These fused images are registered pairwise and the final fused image is made.

6.1.3 Result

For multi-view registration, there is another discussion that is better to segment first then register the image or vice versa [5]. For *C.elegans* embryo, since the registration is based on segmentation to fuse the images, it is easier to use the segmented images.

How to fuse nuclei depends on the size or diameter of the nuclei. Nuclei have a different distribution in shape and distance between them. Diameters (or Principle axis length) are length (in voxels) of the major axes of the ellipsoid that have the same normalized second central moments (covariance matrix) as the nucleus. In other words, we fit a multivariate probability distribution to the nuclei and calculates the central moments for that distribution and the returned second central moment (i.e. variance) is the diameter measurement. To obtain diameters of nuclei in -x,-y and -z-direction the same approach mentioned in chapter 3 section 3.3.4 can be used. Histogram of diameters of nuclei in x,y, and z-direction is calculated for both views (S1Fig A, B). The peak value for the diameter of nuclei in x,y, and z-direction for the reference view is 23.7, 15.7, and 15.7 pixels and for the second view is 17.3, 17.3, and 16.3 pixels. The average value that could be considered as the diameter of nuclei for the fused image is 17.5 $((15.7+17.3)/2)$ pixels. Since this value is smaller than the peak of the distance

between nuclei in 2 views, it is a valid measurement (S1Fig D).

After the fusion of two views number of nuclei added to the fused stack will be 50 nuclei (Figure 6.1 A, C, D), the third view after aligning will be registered with the fused image and result of the registration of 3 views will be obtained. The number of added nuclei will be 20 at this step that shows the next step of group-wise registration is not needed (Figure 6.1 B).

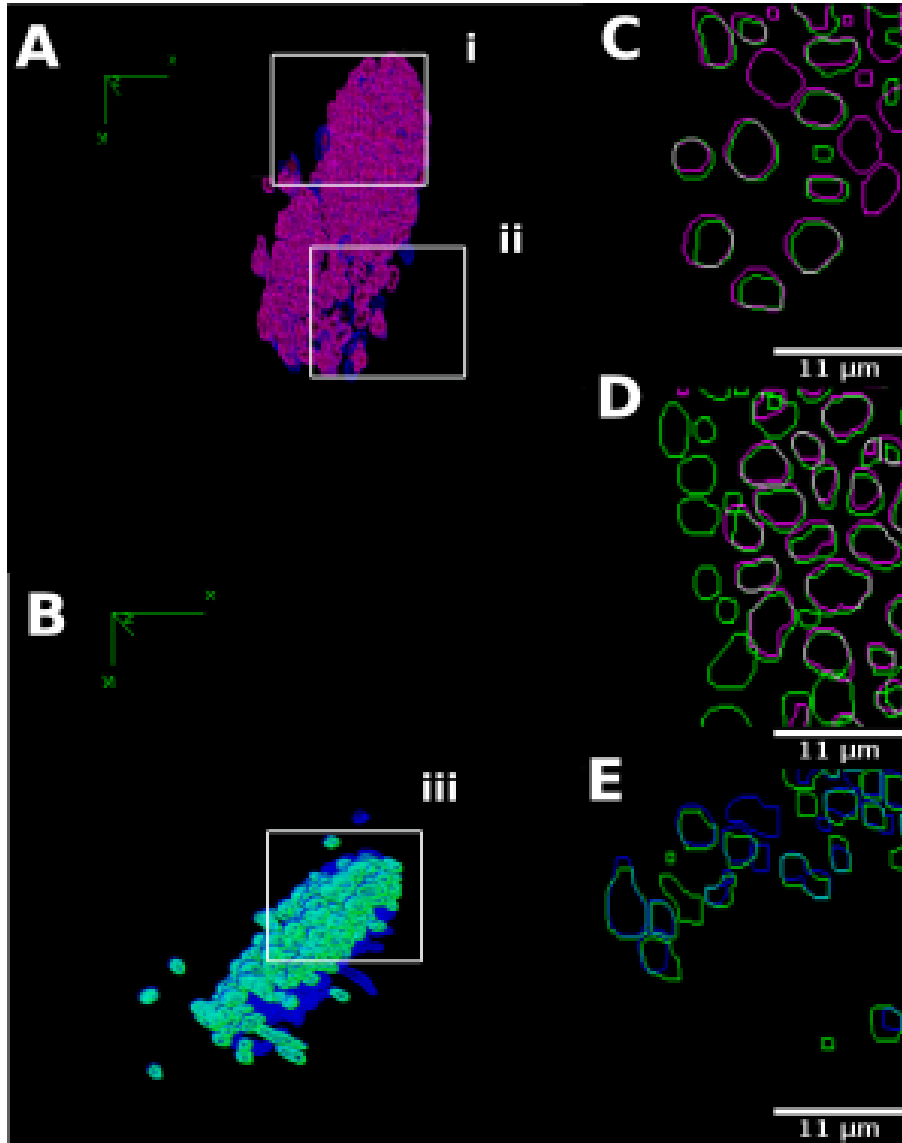


Figure 6.1. The Overlap of the reference view and the fused view of reference view and the second view. left column: 3D view of overlap stack images. A) the result of the fusion step. B) result of the groupwise registration step. The right column shows a 2D outline of the reference view and the fused view (chosen from ROIs). Pink colour shows a fused view and a green colour reference view. Common nuclei are shown by white. C, D) Green colour shows the second view. E) The blue colour shows the third view.

6.2 6.2 Summery

The images are 3 fixed data sets. The first and second data set comes from 3 different views of C.elegans embryos. The third data set contained different views of the head part of C.elegans embryos. We registered the

views pairwise. For Group registration the reverse decision tree method was used.

Registration of 2 views added some nuclei to the first view which were not in the first view. Since it was tried to image the samples in 180° (around $-z$ approximately) different views, the fused image of the first set contains nuclei in-depth z of image (where the outline of nuclei are not easy to quantify due to low resolution of the microscope). In addition to that number of overlap, nuclei are high which makes the registration more precise (Figure 6.1 A, C, D). For the second set fusion, added nuclei could be seen better in the back of the first image (Figure 6.1 B). The number of overlap nuclei is small for this set that suggests not to fuse any more views. If we fuse more views in this case we lose the precision of registration. The number of overlap nuclei is an indication of finishing the registration for groupwise registration. This parameter is used in Reg3D Gui for the user to stop the registration.

Chapter 7

GUIs

CS3D: Graphical user interface (GUI)

A user-friendly graphical user interface (GUI) for visualization and correction of the results was developed (S3Fig). Theoretically separation of nuclei could be repeated till all the nuclei are separated but practically due to limited sources and accuracy manual correction of the result is needed. The developed GUI could help the experimenters to manually correct the segmented nuclei. In addition to that CS3D, GUI is a tool to adjust the parameters of the proposed method to different data sets. To be able to have the code and GUI of the proposed method as a package, Matlab and Fiji were used together. Fiji can be accessed in Matlab by Miji/MIJ [46] or IJM [71] which are Java packages that link between these softwares. Here, IJM package is used since it is open-source and community-supported. To develop or even extend the functionality of the package, framework from various options such as ImageJ macros, Java, MATLAB scripts and C++ languages are used. The result of each step of this tool is saved as a tif file. This helps both developers and users to use this tool more efficiently either for adopting the parameters or skipping a step in the pipeline. After running the GUI, the raw image stack is imported. To be able to run the preprocessing step, parameters matched with data should be entered in the respective text-boxes. The separation step, parameters are shown in another tab in the GUI. The correction part is also included in case experimenters want to correct the segmentation result. The status label in the saving tab shows if each step is finished or not.

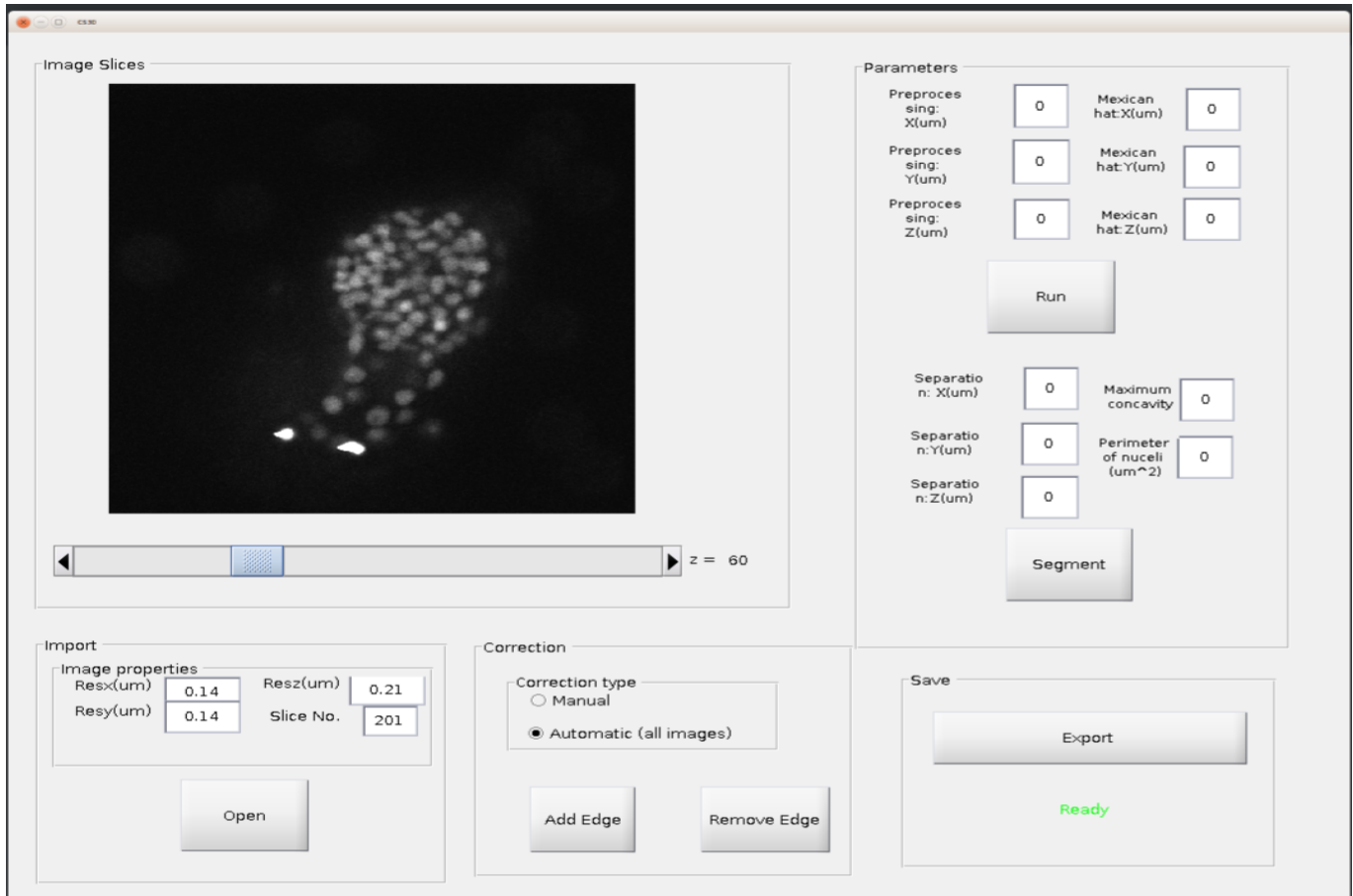


Figure 7.1. CS3D: Graphical user interface (GUI). The GUI includes different tabs for different task. The image should be first imported. After opening the raw image preprocessing parameters should be adjusted. The next step is separate the nuclei 3D and 2D. To correct the segmented image the radio button manual should be chosen and the segmented nuclei could be either separated or connected.

Segmentation

Preprocessing

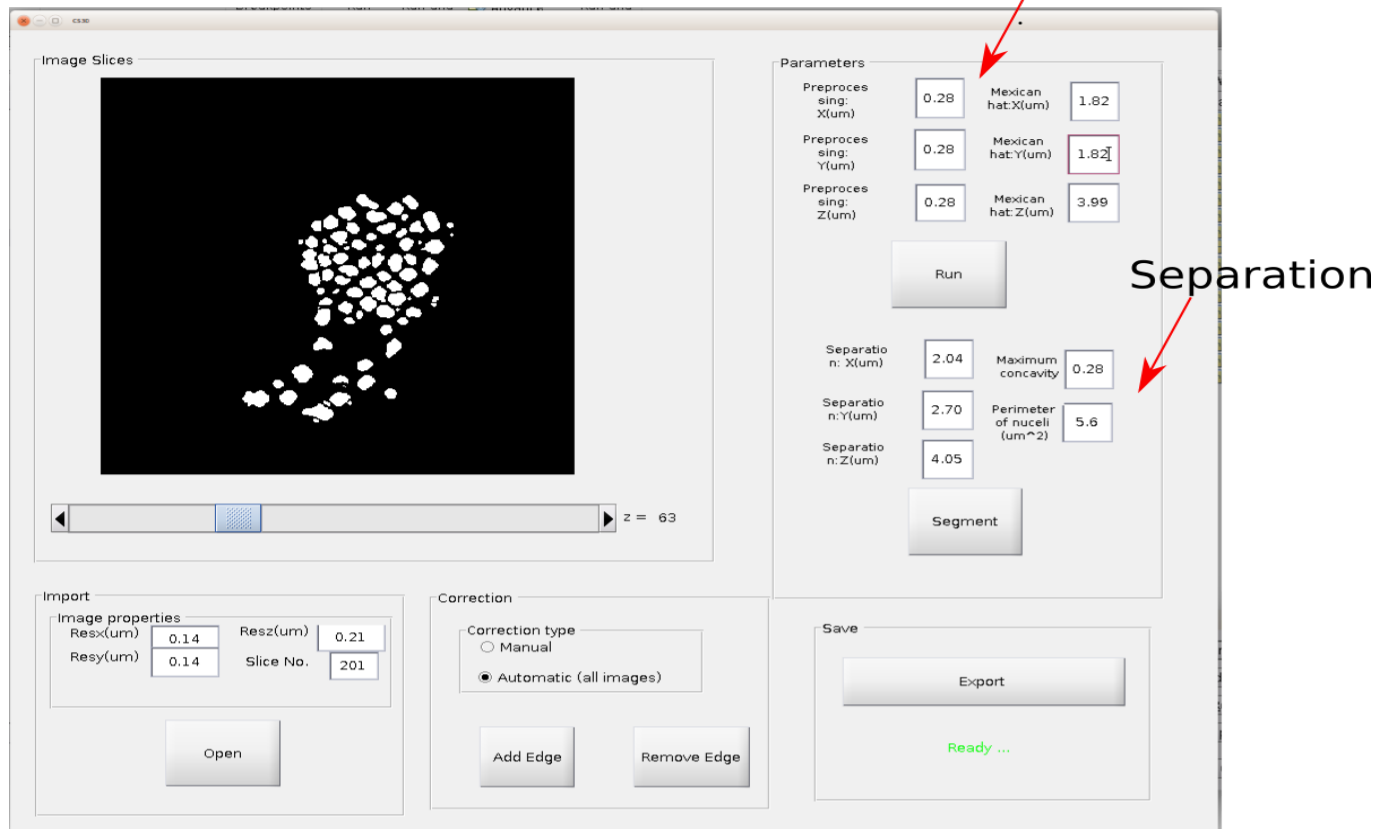


Figure 7.2. CS3D: Graphical user interface (GUI). The GUI includes different tabs for different tasks. After importing the image and preprocessing step, segmentation will be started by clicking the segment push button. The procedure will not start in case the segmented image already exists or the parameters of this section are not valid.

Correction

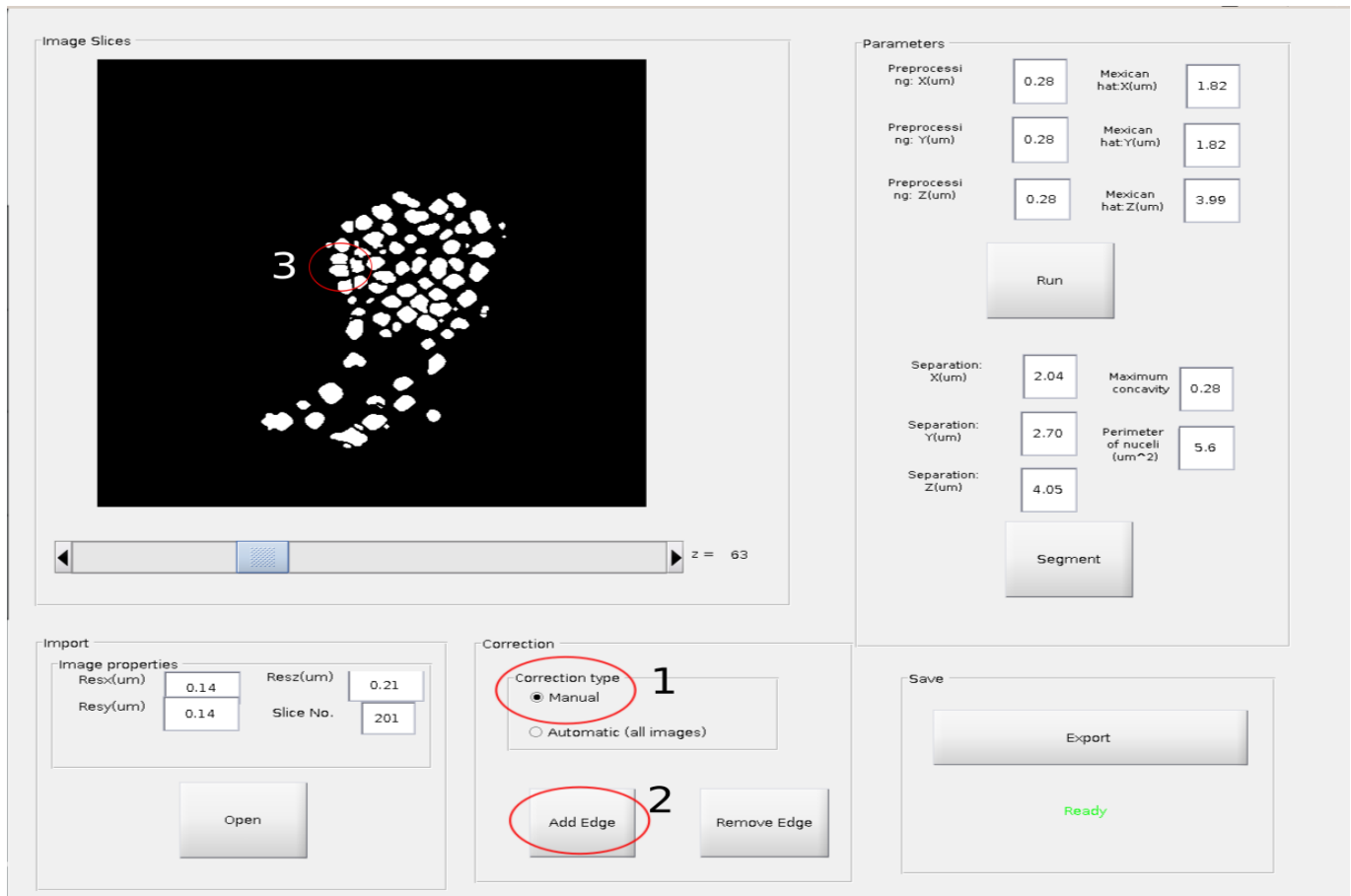


Figure 7.3. CS3D: Graphical user interface (GUI).The GUI includes different tabs for different task. After segmentation the user can correct the segmentation manually. (CS3D does that automatically too.) Correction type should be set to manual and then either Add Edge or Remove Edge push-buttons should be chosen. The user can modify the nuclei by drawing an edge or erasing an edge.

REG3D: Graphical user interface (GUI)

A user-friendly graphical user interface (GUI) for visualization and correction of the results is developed (Figure 7.4).

The developed GUI could help the experimenters to visualize the embryos and see the quantitative measurements of the registration result. In addition to that CS3D, GUI is a tool to adjust the parameters of the proposed method to different data sets. To be able to have the code and GUI of the proposed method as a package Matlab and Fiji were used together. To develop or even extend the functionality of the package, framework from various options such as ImageJ macros, Java, MATLAB scripts and C++ languages are used. REG3D includes different tabs: Importing and alignment, Registration, Displaying quantitative measurements, Fusion, and 3D view of result (next section).

7.1 How to use REG3D GUI.

- **Data directory:** Fill in the directory name that contains all the image files (or directories with equally sized 2d image planes). You can either drag drop the directory, browse for it or type the name directly.

- Pattern of files:** Define the naming convention of the input files by explaining how the angle and the timepoint are encoded into the filename. For example the file names are named in following way: *SegemntedImage_{view}{1}*, *SegemntedImage_{view}{2}*, ...
 That means the pattern of the file names corresponds to *SegemntedImage_{view}{1}.tif*, where {a} is replaced with the current view.
- Aligning Option:**
 Define the transformation applied to second view to get aligned with the reference view:
 FV: Flip Vertically
 FH: Flip Horizontally
 FZ: Flip z axis
 R90R: Rotate 90 right
 R90L: Rotate 90 left
- Image Properties:** Fill in the resolution and depth of the images needed to be registered.
- Quantitative measurements:**
Feature: choosing the feature values needed to be imported. These feature could be chosen after **feature selection** or **after Registration**. The display could show either the **average** of feature values of nuclei or the **individual** values. To be able to zoom in the figure y-axis scale **min** and **max** could be used.
- Fusion:** Nuclei fusion could be done either at the feature level or decision level. The nuclei which have overlap more then **overlap percentage** will be removed. The nuclei in which their distance is less then **distance between nuclei** will be removed.

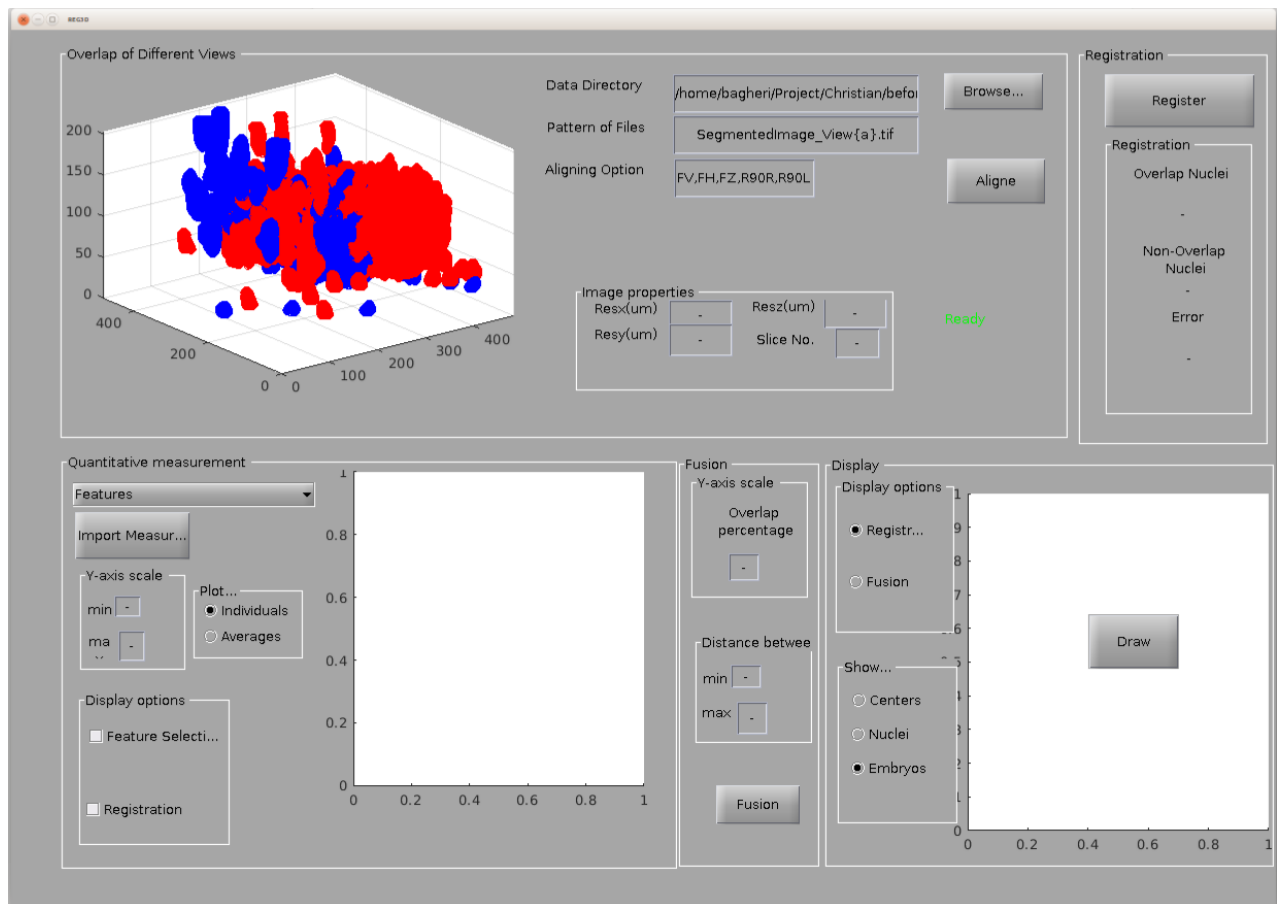


Figure 7.4. REG3D: Graphical user interface (GUI). The GUI includes different tabs for different task.

Chapter 8

Conclusion

This work is making the following contributions: 1) We proposed a 3D segmentation method for the low resolution, high dense distribution of nuclei. This method could be used as a tool for the segmentation of nuclei during developmental morphogenesis. 2) We improved the local geometry descriptor feature selection for using with nuclei registration. 3) A combination of point and voxel registration is proposed that is not only accurate for multi-view registration of *C.elegans* embryo, it does n't need any special experimental preparation such as tilted confocal acquisitions. 4) A machine learning-based approach (two fusion schemes) is developed to fuse the registered image. At decision-level fusion, merging non-overlap nuclei by means of KNN learning algorithm; and at feature-level fusion, concatenating nuclei of both sets by means of geometry features will be used.

For the segmentation part, we proposed a method that 3D segment the low resolution, high dense distribution of nuclei successfully. Nuclei during morphogenesis are small and have irregular shapes.

We were focused on separating the nuclei both on preprocessing steps and postprocessing steps. By using a 3D Mexican hat filter, the image got enhanced while denoised. The binary image was chosen by a method that optimizes the separation between foreground and background based on image histogram. If a 3D stack histogram is used, the brightest nuclei are foreground. A combination of 2D stack histogram binarization with a Mexican hat filter produces separate yet correctly detected nuclei. The rest of the overlap nuclei are separated by 3D Gaussian curvature criterion and 2D convex-concave analysis. A conventional method such as 3D watershed needs seeds and outlines (gradient magnitude of grey values) of nuclei to segment the nuclei. Finding seeds and outline of nuclei produce 2 uncertainties that leads to large error. In addition to that if we regard a 3D image as a stack of 2D images, the concept of the 3D border in 3D image introduces another uncertainty in z-direction that makes it harder for 3D watershed to segment nuclei successfully [72,70]. At first attempts, it was tried to solve the problem in 2D that is easier. Santella et al. [7] segmented 2D slices and then modeled the 3D segmentation of nuclei based on a trained probabilistic model of nuclei shape. Zhang et al. [9] Segmented 2D slices, labeled the nuclei, and then by using the Gaussian mixture and KNN 3D segmented nuclei of the embryo is obtained. To separate nuclei in 2D, the outline or border of the segmented area is used. The convex hull of the outline is calculated and based on that the nuclei will be separated [9,11]. To use this method it should be considered that it is sensitive to the noise on the outline of nuclei.

There were some attempts for 3D segmentation [70,17,15]. With the development of 3D algorithms for 3D image and hardware, more of these algorithms developed.

Toyoshima et al. [13,20] have developed a 3D segmentation method for nuclei. Their method uses 3D Gaussian curvature to separate nuclei of *C.elegans* adult worm. This method is not able to segment *C.elegans* embryo nuclei since the density of nuclei in embryos is higher than an adult *C.elegans*. LOS 3D segmentation [17,41] uses weak convex property for separation. This property is not precise enough to separate the nuclei of *C.elegans* embryo since they have irregular shapes rather than well-defined shapes.

3D multi-view registration of biological tissue to overcome the low limitation in the z-axis is a challenge now. Segmented nuclei could be used as markers for registration. The more accurate the segmentation is, the more accurate the registration will be. In addition to that shape of nuclei could also provide extended information for registration [5,11].

In addition to hardware and time limitation of 3D cell segmentation proposed here, there are some connected nuclei that can not be separated by this method due to having very smooth changes of curvature in the connected area. Theoretically, these nuclei could be separated but the size of the nuclei will be decreased drastically. There are some nuclei that cannot be binarized due to the low ratio of the intensity of nuclei/background. This happens due to the limitation of resolution in the z-axis. For this problem either Image acquisition should be an improved or more complicated method like multi-view registration should be developed.

The problem with this approach is that it can not be used for all types of data especially the ones with considerable size to the beads. To combine this idea with nuclei segmentation data, centre of nuclei is used as a local descriptor [72] to be used for geometric hashing [25]. The result of geometric hashing to the small number of nuclei is not efficient, So here we develop a method in feature selection part to use both local descriptor geometry hashing of centres and geometry information of nuclei like volume, first eigenvalue, first eigenvector, and eccentricities.

This combination of geometry information and local descriptor could be used for nuclei based feature selection.

Finding a robust registration algorithm to match the nuclei is the next step. Point set registration algorithms should be able to accurately model the transformation required to align the point sets with tractable computational complexity while robust to noise, outliers, and missing points. While there are many registration methods have been developed by now, they still have the problem of accuracy and robustness. ICP [29] is one of the most general algorithm proposed in this field. It alternates between building closest-point correspondences under the current transformation and estimating the transformation with these correspondences, until convergence. A good initial value is essential to reach an accurate model-data transformation. To improve the robustness of this method LMICP [33] has been used that suggests to use different kernels for the error function plus using a numerical method to make the algorithm faster. (Especially when the overlap is high.) EM[34] method uses a probabilistic method to register multiple point sets. This method doesn't performs pairwise alignments between sets and treats all the point sets on an equal footing. An extended version of this method could be used for non-rigid registration.

To reach an accurate registration, voxel-based registration inducing nuclei registration and embryo alignment should be considered here. Nuclei could be chosen by Ransac or Trimming method to avoid computational challenges.

The combination of guided point and voxel-based registration has been shown to be able to register the embryos accurately. The evaluation of the result of registration has been done by comparing the feature of common nuclei such as volume, eigenvalue, and eigenvector before and after registration. The similarity between common nuclei increases drastically after registration. For the method evaluation, part 3 methods were chosen. The result showed that registration based on LMICP+ICP Huber has the highest accuracy.

The proposed method is successfully used on different data set of C.elegans embryos and synthetic data sets. Even though our approach doesn't need any experimental preparation, it is dependent on segmentation of image that could produce errors while changing from pixel/voxel data to 3D stack image. Further work on the registration technique comprises improving the point set registration techniques. Unfortunately, the state of the art point set registration methods are not able to reach the demanded correctness needed for this approach. At the next step, we want to use this method for the time-lapse registration of the C.elegans embryo.

Appendix A

Appendix

Table A.1. S1: Names and parameter values of filters used in the proposed method.

Process	Filter	Parameters of Data1	Parameters of Data2
Preprocessing	3D Gaussian Blur	2*2*2 pixels	2*2*2 pixels
Preprocessing	3D Maximum	2*2*2 pixels	2*2*2 pixels
Preprocessing	LOG	13*13*19 pixels	11*11*15 pixels
Segmentation	3D Gaussian Curvature Separation	$S=5101$ voxel ³	$S6653$ voxel ³
Segmentation	2D Concave-Convex Separation	$T_d=2$ pixels— Perimeter Threshold=40 pixels.	$T_d=1.5$ pixels— Perimeter Threshold=50 pixels.

$S \simeq 4\pi(((ab)^{(1.6)} + (ac)^{(1.6)} + (bc)^{(1.6)})/3)^{1/1.6}$ a,b and c are ellipsoid axis lengths in x,y and z.

S1 Fig: Shape and spatial distribution of nuclei.

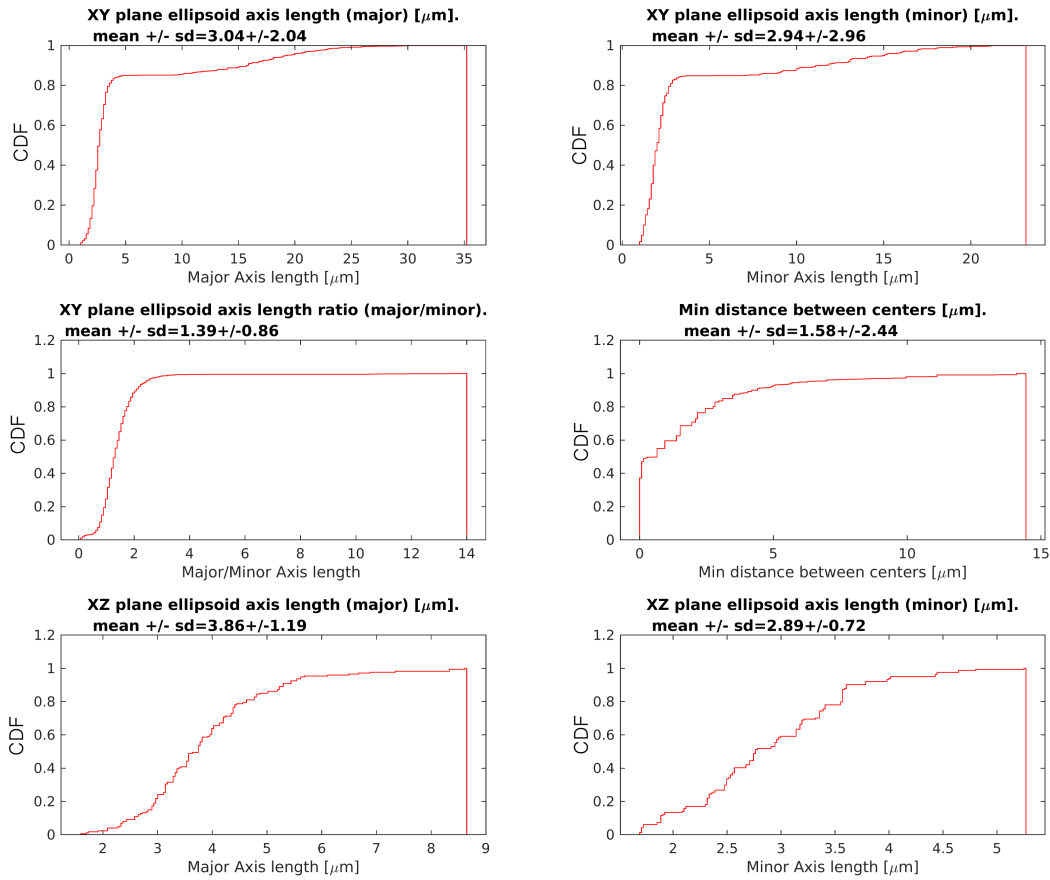


Figure A.1. Cumulative distribution of histogram of A,B)Major and minor diameters of nuclei in XY and XZ planes and C)ratio of major to minor diameters of nuclei in XY plane. D)Distance between nuclei of Dataset 1. E,F) The same procedure of (A) and (B) but for XZ plane. The value of XZ plane are just calculated for 20 slices while the XY plane for 201 slices. The measurements are calculated from manual ellipse fitting of original data set 1 after Binarization.

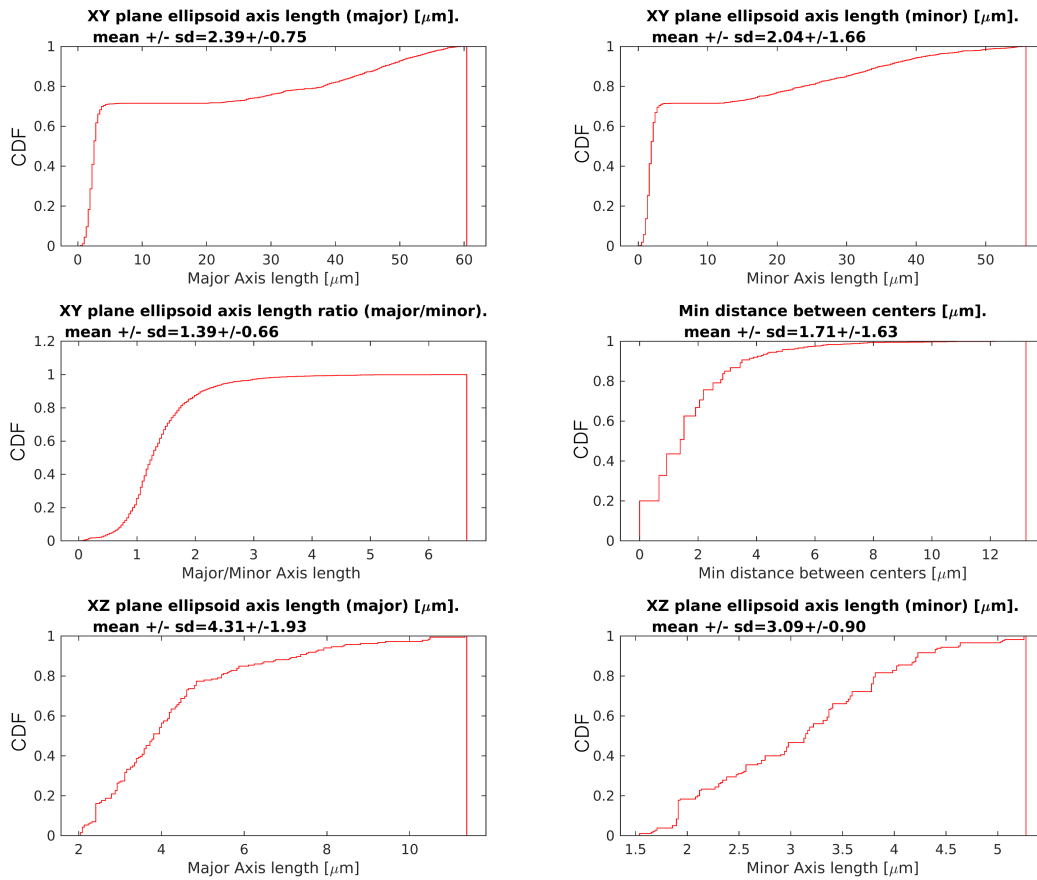


Figure A.2. Cumulative distribution of histogram of A,B)Major and minor diameters of nuclei in XY and XZ planes and C)ratio of major to minor diameters of nuclei in XY plane. D)Distance between nuclei of Dataset 1. E,F) The same procedure of (A) and (B) but for XZ plane. The value of XZ plane are just calculated for 20 slices while the XY plane for 201 slices. The measurements are calculated from manual ellipse fitting of original data set 2 after Binarization.

S2 Fig: 3D Gaussian intensity distribution of nuclei.

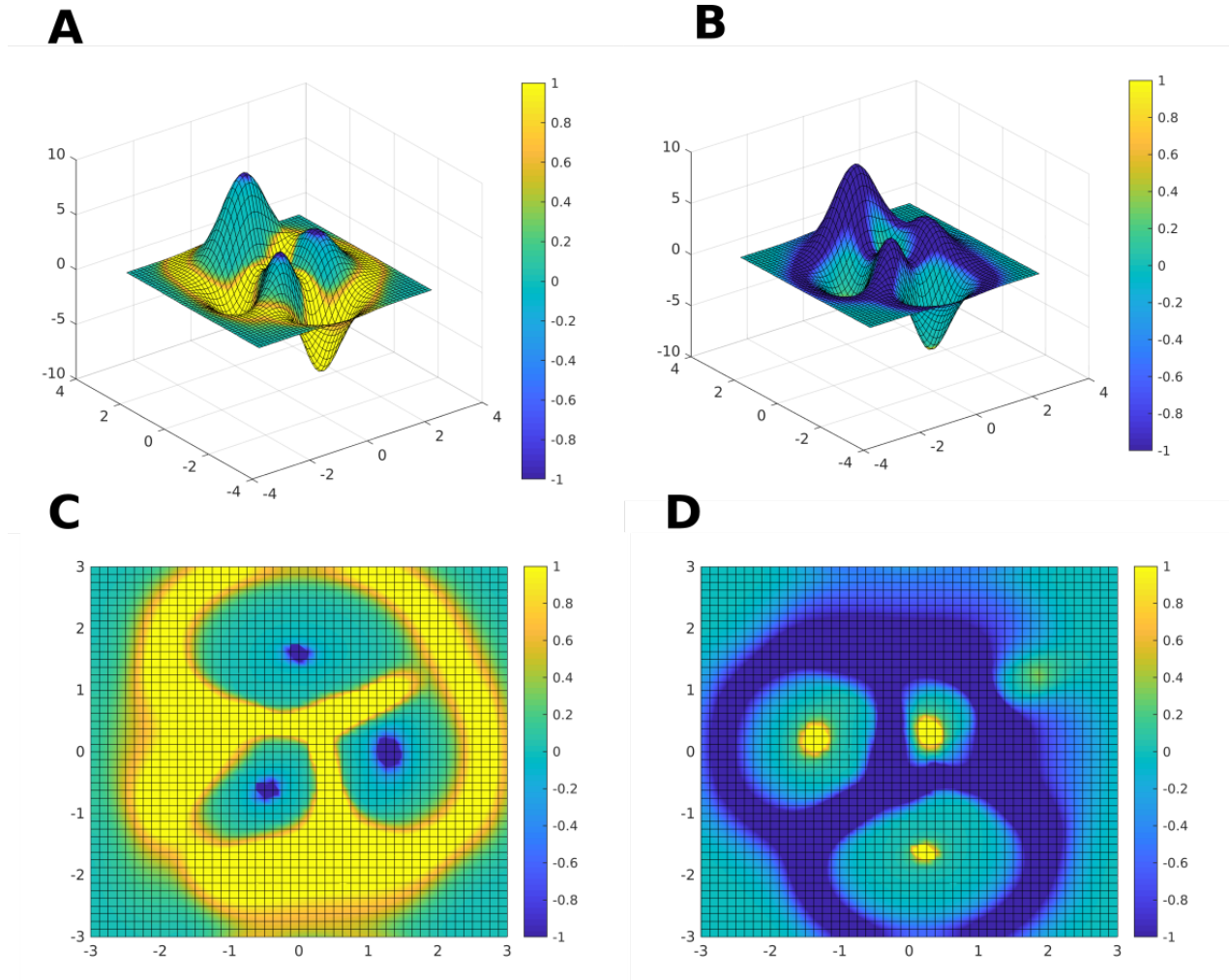


Figure A.3. 3D Gaussian intensity distribution of nuclei. A) Surf plot of larger principle of Gaussian curvature, B) Smaller principle of Gaussian curvature of a Gaussian mixture distribution. C) 2D view of the larger principle of Gaussian curvature. D) 2D view of the smaller principle of Gaussian curvature. The distribution used is $z = 3 * (1 - x)^2 * \exp(-(x^2) - (y + 1)^2) - 10 * (x/5 - x^3 - y^5) * \exp(-x^2 - y^2) - 1/3 * \exp(-(x + 1)^2 - y^2)$. With x and y being a 49*49 2D matrix. By using the positive value of larger principle the 3 Gaussians could be separated.

S4 Fig: Registration of centres.

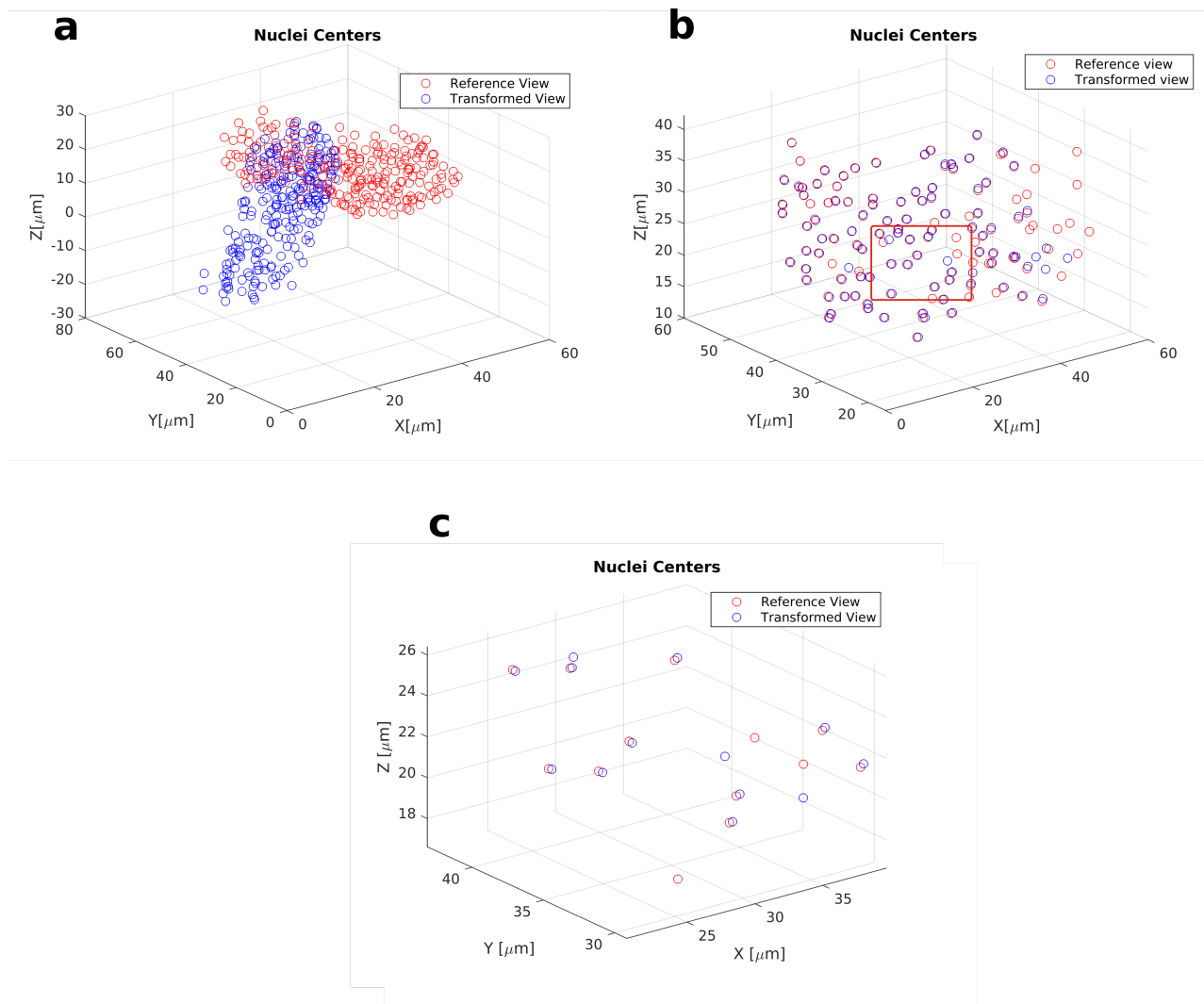


Figure A.4. Registration of centres. The overlap of the reference view (red) and the transformed reference view (blue) nuclei centres a) before registration, b) after registration. c) Enlarged area of ROI chosen in (b).

Bibliography

1. Coralie Fouquet, Jean-François Gilles, Nicolas Heck, Marc Dos Santos, Richard Schwartzmann, Vidjeacoumary Cannaya, Marie-Pierre Morel, Robert Stephen Davidson, Alain Trembleau, and Susanne Bolte. Improving Axial Resolution in Confocal Microscopy with New High Refractive Index Mounting Media. *Plos one* **10** (3), (2015). (1989).
2. Shaw, P, Agard, D, Hiraoka, Y Sedat, J. Tilted view reconstruction in optical microscopy. Three-dimensional reconstruction of *Drosophila melanogaster* embryo nuclei. *Biophysical Journal* **55**, 101–110 (1989).
3. Fitzpatrick, JM West, JB. The distribution of target registration error in rigid-body point-based registration. *IEEE Transactions on Medical Imaging* **20**(9): 917927 (September 2001).
4. Matul, P, Kozubek, M, Staier, F Hausmann, F. Precise 3D image alignment in micro-axial tomography. *Journal of microscopy*209: 12642 (2003).
5. S. Preibisch S. Saalfeld, JS Tomancak, P. Software for bead-based registration of selective plane illumination microscopy data. *Nature Methods* **7**, 418–419 (2010).
6. S. Preibisch F. Amat, ESMSRSEM Tomancak, P. Efficient Bayesian-based Multiview Deconvolution. *Nature Methods*, **11**(6):645-648 (2014).
7. Santella A., ZSA-KZ. hybrid blob-slice model for accurate and efficient detection of fluorescence labeled nuclei in 3D. *BMC Bioinformatics* **11**, 580 (2010).
8. Rajasekaran B Uriu K, VGTJ-YOA. Object Segmentation and Ground Truth in 3D Embryonic Imaging. *PLoS ONE* **11**(6) (2016).
9. Zhang C Sun C, SRPT. Segmentation of clustered nuclei based on curvature weighting. *Proceedings of the 27th Conference on Image and Vision Computing New ZealandIVCNZ 12. New York, New York, USA: ACM Press*; **49** (2012).
10. Wienert S Heim D, SKea. Detection and segmentation of cell nuclei in virtual microscopy images: A minimum-model approach. *Sci Rep* **2**, 503 (2012).
11. Qi, et al. *Drosophila* eye nuclei segmentation based on graph cut and convex shape prior. *20TH IEEE International Conference on Image Processing - ICIP 2013* **1**, 670–674 (2013).
12. J, Q. Dense nuclei segmentation based on graph cut and convexity-concavity analysis. *J Microsc* **253**, 42–53 (2014).
13. Toyoshima, Y et al. Accurate Automatic Detection of Densely Distributed Cell Nuclei in 3D Space. *PLoS Comput Biol* **12** (2016).
14. Stegmaier J Amat F, LWTG-MRKPea. Real-Time Three-Dimensional Cell Segmentation in Large-Scale Microscopy Data of Developing Technology Real-Time Three- Dimensional Cell Segmentation in Large- Scale Microscopy Data of Developing Embryos. *Dev Cell. Elsevier Inc* **36**, 225–240 (2016).

15. Ya-Hsin Cheng Tzu-Chieh Lin, J-JDY-FW Lin, S-J. Adaptive 3D cell segmentation and tracing algorithm using convex separation and histogram information for vivo images. *IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, 133–138 (Hong Kong 2017).
16. Harder N.; Bodnar, MERS-DRK. 3d segmentation and quantification of mouse embryonic stem cells in fluorescence microscopy images. *Proc ISBI'10*, 216–219 (2011).
17. Asafi, S, Goren, A Cohen-O, D. Weak Convex Decomposition by Lines-of-sight. *Comput Graph Forum*, 23–31 (2013).
18. Sommer C Straehle C, KUHFa. Ilastik: Interactive learning and segmentation toolkit. textit2011 IEEE International Symposium on Biomedical Imaging: From Nano to Macro. IEEE; 2011, 230–233 (2011).
19. Al-Kofahi Y Lassoued W, LWRB. Improved automatic detection and segmentation of cell nuclei in histopathology images. *IEEE Trans Biomed Eng* **57**, 841852 (2010).
20. Nguyen, J, Linder, A, Plummer, G, Shaevitz, J Leifer, A. Automatically tracking neurons in a moving and deforming brain. *PLoS Comput Biol* **13**(5) (2017).
21. D. Sage F.R. Neumann, FHSGMU. Automatic Tracking of Individual Fluorescence Particles: Application to the Study of Chromosome Dynamics. *IEEE Transactions on Image Processing* **14**, 1372–1383 (2005).
22. Kronenberger, M, Wirjadi, O, Freitag, J Hagen, H. Gaussian curvature using fundamental forms for binary voxel data. *Graphical Models* **82**, 123–136 (2015).
23. Weiguo Chen, Jinchuan Lu. Human Shape Recognition Algorithm Design Based on Hu Moments and Zernike Moments Authors. *4th International Conference on Machinery, Materials and Information Technology Applications*. (2016).
24. Chengcai Leng ; Hai Zhang ; Bo Li ; Guorong Cai ; Zhao Pei ; Li He. Local Feature Descriptor for Image Matching: A Survey. *IEEE Access*. **7**, 6424 - 6434. (2018).
25. A.S. Mian, MB Owens, R. Three-dimensional model-based object recognition and segmentation in cluttered scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **28**, 1584–601 (Oct.2006).
26. Moll Mark; Bryant, DHKLE. The LabelHash algorithm for substructure matching. *BMC Bioinformatics* **11**, 555.
27. Myronenko, A Song, X. Point set registration: Coherent point drift. *IEEE Trans. Pattern Anal. Mach. Intell.* **32**, 2262–2275 (2010).
28. S. Zhang Y. Yang, KYLSHO. Point set registration with global-local correspondence and transformation estimation. *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 2669-2677 (Oct. 2017).
29. Yang, J, Li, H, Campbell, D Jia, Y. Go-ICP: a globally optimal solution to 3D ICP point-set registration. *PAMI* (2016).
30. He Y.; Liang, BYJLSH. An Iterative Closest Points Algorithm for Registration of 3D Laser Scanner Point Clouds with Geometric Features. *Sensors*, 1862 (2017).
31. D. Chetverikov Comput. Autom. Inst. Budapest, HSDSPK. The Trimmed Iterative Closest Point algorithm, *Object recognition supported by user interaction for service robots* **3**, 545–548 (2002).
32. Yan Guo, YG Zhang, Y. Invariant Feature Point based ICP with the RANSAC for 3D Registration. *Information Technology Journal* **10**, 276–284 (2011).
33. Y.F.Wu, W.Wang, K.Q.Lu, Y.D.Weiz C.Chen. A New Method For Registration Of 3D Point Sets With Low Overlapping Ratios. *Procedia CIRP 27 (the 13th CIRP conference on Computer Aided Tolerancing)* (2015).

34. Evangelidis, G Horaud, R. Joint Alignment of Multiple Point Sets with Batch and Incremental Expectation-Maximization. *IEEE Transactions on Pattern Analysis and Machine Intelligence, Institute of Electrical and Electronics Engineers* **40**, 1397–1410 (2018).
35. Shapiro, LG Stockman, GC. Computer Vision. New Jersey, Prentice-Hall, 279–325 (2001).
36. Dilpreet Kaur, YK. Various Image Segmentation Techniques: A Review. *International Journal of Computer Science and Mobile Computing* **3**, 809–814 (May 2014).
37. Hildreth, DME. Theory of Edge Detection. *Proceedings of the Royal Society of London. Series B, Biological Sciences. The Royal Society.* **207**(1167): 215217. (29 February 1980).
38. Lindeberg. Image matching using generalized scale-space interest points. *Journal of Mathematical Imaging and Vision* **52**, 3–36 (2015).
39. Otsu, N. A threshold selection method from gray-level histograms. *IEEE Trans. Sys., Man., Cyber* **9**, 6266 (1979).
40. Atta-Fosu, T et al. 3D Clumped Cell Segmentation Using Curvature Based Seeded Watershed. *Imaging*, **2** (2016).
41. Legland D.; Arganda-Carreras, IAP. MorphoLibJ: integrated library and plugins for mathematical morphology with ImageJ. *Bioinformatics (Oxford Univ Press)* **32**, 3532–3534 (2016).
42. J.PVial. Strong and weak convexity of sets and functions. *Math. Oper. Res.*, **8**, pp.231-259 (1983).
43. Mathew, Ba. Robust and automated three-dimensional segmentation of densely packed cell nuclei in different biological specimens with lines-of-sight decomposition. *BMC Bioinformatics* **16**, 187 (2015).
44. Hodneland E.; Kgel, TFDGHLA. CellSegmA MATLAB toolbox for high-throughput 3D cell segmentation. *Source Code Biol. Med* **8** (2013).
45. Tibshirani, GJDWTHR. An Introduction to Statistical Learning. *Springer*. 204 (2014).
46. Lipson, SLM. Linear Algebra (Schaums Outlines) 4th (McGraw Hill, 2009).
47. Spellman, MSSLD. Vector Analysis (Schaums Outlines) 2nd (McGraw Hill, 2009).
48. Zhiyong Zhou Jianfei Tu, CGJHBTJJYD. Accurate and Robust Non-rigid Point Set Registration using Students-t Mixture Model with Prior Probability Modeling. *Scientific Reports*, 8742 (2018).
49. Fitzgibbon, AW. Robust Registration of 2D and 3D Point Sets. *Proceedings of the British Machine Vision Conference* (2001).
50. Allan Max; Kapoor, AMPMP. Non Rigid Registration of 3D Images to Laparoscopic Video for Image Guided Surgery in Springer International Publishing: (2015), 109–116.
51. Studholme C.; Hill, DLHDJ. Automated 3D Registration of Truncated MR and CT Images of the Head. *Proceedings of the Sixth British Machine Vision Conference*, 27–36 (1995).
52. Evangelidis, G, Kounades-Bastian, D, Horaud, R E.Z., P. A generative model for the joint registration of multiple point sets in European Conference on Computer Vision (ECCV) (2014).
53. Jian, B Vemuri, B. A robust algorithm for point set registration using mixture of gaussians. *ICCV*, 1246–1251 (2005).
54. Huber, P. Robust Statistics (eds Wiley, J Sons) (1981).
55. Brown BJ, RS. Global non-rigid alignment of 3-d scans. *ACM Transactions on Graphics (TOG)* **26**,

- 21 (2007).
56. Chui H, RA. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding* **89**, 114–141 (2003).
57. Chen Y Zhao J, DQDF. D craniofacial registration using thin-plate spline transform and cylindrical surface projection. *PLoS ONE* **12** (2017).
58. Hutton TJ Buxton BF, HP. Automated registration of 3d faces using dense surface models. *British Machine Vision Conference*, 439–448 (2003).
59. Yuille, A Grzywacz, N. The motion coherence theory. *Int. J. Computer Vision* **3**, 344–353 (1988).
60. Benjamin Schmid Johannes Schindelin, ACML Heisenberg, M. A high-level 3D visualization API for Java and ImageJ. *BMC Bioinformatics* **111**, 274 (2010).
61. Genderen, JLv Pohl, C. Image fusion: Issues, techniques and applications. *Intelligent Image Fusion, Proceedings EARSeL Workshop, Strasbourg, France, edited by J. L. van Genderen and V. Cappellini (Enschede: ITC)*, 18–26 (1994).
62. Jiang, D, Zhuang, D, Huang, Y Fu, J. Survey of Multispectral Image Fusion Techniques in Remote Sensing Applications. *Image Fusion and Its Applications Yufeng Zheng, IntechOpen*, (2011).
63. Dai, X Khorram, S. Data fusion using artificial neural networks: a case study on multitemporal change analysis. *Comput. Environ. Urban Syst* **23**, 19–31 (1999).
64. Blum, R Li u, Z. Multi-Sensor Image Fusion and Its Applications. *special series on Signal Processing and Communications; CRC Press: Boca Raton, FL, USA*, (2006).
65. Patel, K. Fusion Algorithms for Image Based on Principal Component Analysis and Discrete Wavelet Transform. *International Journal for Innovative Research in Science and Technology* **1**, 180–182 (2015).
66. Adelson, C Bergen, J. *Pyramid methods in image processing*. *RCA Eng* **29**, 33–41 (1984).
67. Mallat, S. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **11**, 674–693 (1989).
68. Louis, E Yan, X. A neural network model for estimating sea surface chlorophyll and sediments from thematic mapper imagery. *Remote Sens. Environ* **66**, 153–165 (1998).
69. Sage D Prodanov D, TJ-YSJ. Making Interoperability between ImageJ and Matlab Possible. *ImageJ User Dev Conf Poster* **4** (2012).
70. Ahonen T1 Hadid A, PM. Face description with local binary patterns: application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **28**, 2037–2041 (2006).
71. Cheng JR, Rajapakse JC. Segmentation of clustered nuclei with shape markers and marking function. *IEEE Trans Biomed Eng.* **56**, 741–748 (2009).
72. Li G, Liu T, Tarokh A, Nie J, Guo L, Mara A, et al. 3D cell nuclei segmentation based on gradient flow tracking. *BMC Cell Biol.* 8:40.(2007).
73. http://www.lix.polytechnique.fr/maks/Verona_M_PAM/TD/TD1/
74. <https://blogs.mathworks.com/steve/2016/06/14/image-binarization-otsus-method/>
75. <https://stackoverflow.com/questions/2545323/laplacian-of-gaussian-filter-use>