AN INTEGRATION FRAMEWORK FOR UNDERSTANDING MULTI-MODAL 3D MEDICAL VOLUMES

BY

SHADI ASHNAI

B.S., Sharif University of Technology, 2004

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2007

Urbana, Illinois

Adviser:

Professor John C. Hart

Co-Adviser:

Dr. Peter Bajcsy

Abstract

In this thesis we have proposed an integration framework for understanding multi-modal 3D medical volumes. The integration framework consists of a sequence of operations that are designed to support the transfer of raw data to knowledge, and to enable learning and exploration of medical hypotheses about imaged specimens. The framework includes algorithms for reconstructing, integrating, analyzing and visualizing multi-modal medical data.

In our work, raw medical data are represented by 2D and 3D images. The images correspond to the same tissue and are acquired using different imaging instruments and processes. Different imaging modalities provide a variety of properties of the tissue of interest. Some raw data sets, such as magnetic resonance (MR) or computer tomography (CT) images, form already a 3D volume. Other data sets, such as a set of 2D microscopy images of histological cross sections, have to be aligned to reconstruct a 3D volume. When multiple multi-modal data sets form 3D volumes, volumes of different modalities can be spatially integrated into the same coordinate system using computer assisted techniques. During the integration, the following challenges have to be addressed: (a) the spatial resolution of multi-modal volumes might differ in every dimension, (b) the appearance of the same physical tissue varies across modalities, (c) the modality specific measurements represent grayscale (MRI, CT, or Neutron Beam) or color (Histology) or vector (diffusion tensor (DT) images) values, and (d) the file size of 3D volumes requires significant computational resources and scalable algorithms. In addition to integration, there is a need to support the end users of integrated data by providing 3D visualization and quantitative feedback about the estimated integration accuracy

Due to the large variety of specimens, imaging techniques and preparation methods to obtain raw data, the current framework has been designed

as computer assisted rather than as fully automated. The full automation of each algorithm is outside of the scope of this work. The main contribution of this work is in designing and prototyping an integration framework that includes algorithms for detecting and clustering of features, extraction of foreground in volumes, reconstruction of 3D volumes from 2D cross sections, 3D-to-3D registration and 3D visualization of multi-modal information. The framework could be used not only for transforming raw data to knowledge about the imaged specimens but also for better understanding of the uncertainty introduced by integration.

The prototype was applied to a specific study focusing on understanding multi-modal correlations of gender specific patterns and stuttering patterns of myelinated fibers in animal brain models.

To my parents, Mahvash & Mohammad
To my sister, Bahar
To Maziar

Acknowledgments

This project would not have been possible without the support of many people. Thanks to John C. Hart and Peter Bajcsy for advising this work and providing limitless insight. Also, thanks to Kenneth L. Watkin for his constant encouragement and feedback and providing us with all of this data. Thanks to all the people in the ISDA group at National Center for Supercomputing Applications for sharing their code and knowledge. And finally, thanks to my family and friends who endured this process with me, always offering support and love.

Table of Contents

List of Tables . . . . . **viii**

List of Figures . . . . . **ix**

Chapter 1 Introduction **1**

Chapter 2 Related Work **7**
2.1 Feature Detection 7
2.2 Volume Registration 8

Chapter 3 3D Volume Reconstruction **10**
3.1 Background Subtraction 11
3.1.1 Color-based Background Subtraction 11
3.1.2 Noise Removal 13
3.1.3 Hole Filling 14
3.2 Volume Reconstruction 14
3.2.1 Slice Correspondence 15
3.2.2 Slice Alignment 16

Chapter 4 Feature Detection in 3D Volumes **18**
4.1 Feature Detection in Scalar Volumes 18
4.2 Feature Detection in RGB Volumes 19
4.3 Feature Detection in Vector Volumes 20

Chapter 5 3D Volume Registration of Multi-Modal Data . **23**
5.1 Properties of 3D Volumes 23
5.1.1 Centroid 23
5.1.2 Bounding Ellipsoid 24
5.2 Primitive Transformations 24
5.2.1 Centroid-based Translation 24
5.2.2 Bounding Ellipsoid Based Scaling 25
5.2.3 Bounding Ellipsoid Based Rotation 25
5.3 Automatic Pre-alignment 26
5.4 Registration Quality Evaluation 28
5.4.1 Overlapping Metric 29

Chapter 6 Results 30
6.1 Integration of 2D Images 30
6.2 Integration of 3D Volumes 33

Chapter 7 Conclusion 36

Appendix A Graphical User Interface 37
A.1 Open Multiple Volumes 38
A.2 Volume Control Panel 38
A.2.1 Information Panel 38
A.2.2 Change Label 39
A.2.3 Select Visualization Mode 39
A.2.4 3D Transformations 39
A.2.5 Filtering Visible Voxels 40
A.2.6 Filtering Fiber Clusters 40
A.3 Volume Visibility Selection 41
A.4 Volume Registration 41
A.5 Volume Comparison 41

References 43

List of Tables

6.1 Quantitative Results from Registering Lung Images 32

List of Figures

1.1 Corpus Callosum Correspondence in Different Modalities of Human's Brain 1
1.2 Workflow of our Integration System 5

3.1 Results of the Color-based Background Subtraction 13
3.2 Illustration of 3D Volume Reconstruction 15
3.3 Affine vs. Rigid Transformation for a single slice 16
3.4 Final Results of 3D Volume Reconstruction 17

4.1 Grayscale Features in MR Volume 19
4.2 Color Features in Histological Volume 20
4.3 Directional Features in DTI Volume 22

5.1 Result of Rotation only Based on Bounding Volumes 26
5.2 Result of Rotation while Maximizing Overlapping 27

6.1 Images from Different Modalities of Lung Specimen 31
6.2 Unregistered Lung Images After Background Subtraction . . . 31
6.3 Pair-wise Registration Results of Lung Images 32
6.4 Integration Results of All Lung Images 32
6.5 Original Rat Brain Slices and Reconstructed Volumes 34
6.6 Feature Detection in Rat Brain Data 34
6.7 Integration Results of Rat Brain Data 35

A.1 A Snapshot of the Graphical User Interface 37

Chapter 1

Introduction

Medical volumes are usually constructed from a series of images. Variety of image modalities are acquired by different devices and processes. The acquired data volumes vary in their resolution, orientation, coordinate system and attributes (measurements) of the tissue. Since the attributes represent the same tissue, the problem of integrating multiple volumes and finding the correspondences between different data acquisition modalities had always been of a great interest to the medical society. As an example, figure 1.1 shows four different images from four data acquisition that come from corpus callosum of a human. Figure also shows the correspondences between the different modalities.

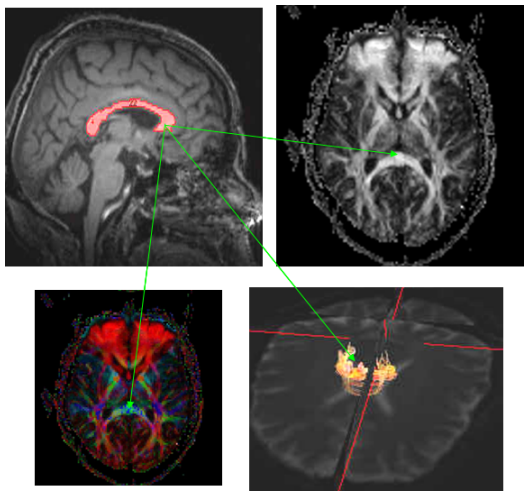

Figure 1.1: Corpus callosum correspondence in the human's brain modalities: MRI (top left), fractional anisotropy (top right), directional water diffusion (bottom left), CAT scan (bottom right).

In order to perform data integration of 3D volumes from different modalities, one needs to first reconstruct the medical volumes from cross sections, detect the foreground by separating from the background, and then bring all the 3D volumes into the same coordinate system. Moreover, there are always several important features in each tissue, for instance myelinated fibers in brain, that doctors would like to inspect visually and understand the correspondence across modalities. Thus, there is a need to provide feature detection capabilities. Our work focuses on designing a framework for integration and analysis of 3D radial volumes including the algorithms and techniques for reconstructing volumes, detecting the foreground and features, and quantitatively analyzing the correspondences.

From 2D Slices to 3D Volumes: Volume reconstruction might not be necessary for all modalities, since most of the acquisition devices generate parallel and well-aligned images. However, when the images are not parallel to each other or aligned to form a 3D volume, the need to a volume reconstruction algorithm from acquired images would become inevitable. This is true for example for histological cross sections which are captured by embedding, slicing and staining the tissue obtained by sectioning 3D specimen of interest. These slices are then placed on glass slides by hand. Therefore, the orientation of one slice might not match with the other slices after this process and a reconstruction algorithm has to be developed to take care of these transformations. We have implemented a technique that reconstructs the volume by finding slice correspondences with one of the existing 3D volumes and registering the images, slice-to-slice, and finally stacking up the registered images to form the 3D volume.

Foreground and Feature Detection: Separating foreground from the background is also one of the essential steps through this integration,

since volume registrations and correspondence analysis take into account in their computations only the foreground voxels. Multiple approaches such as edge or surface detection can be performed on the volumetric data in order to obtain the desired separation of structures of interest. Since the medical images usually have very dark or bright backgrounds, we found it more efficient to build the filter using a simple color-based filtering approach which is by far faster than surface detection. The drawback of color-based filtering is that voxels from the structures of interest might be removed if their color match the background color. In order to compensate the drawbacks of color-based filtering, we have an additional step to our foreground detection which finds small segments, aggregations of the neighboring voxels, and toggles the mask for those segments. If the small segment is detected as a foreground, it would be set to be a background and thus noises are removed from the background. On the other hand, if the segment is already marked to be background, the algorithm toggle the mask for the whole segment and thus a small hole is filled in the structure of interest. We have also used a similar approach for finding features in some of our volumes, since the features were either colored with a known color or had a higher/lower value of brightness. This approach could be used when voxels of the volume are represented using colors. However, for feature tracking other modalities, such as in the directional data, we had taken another approach which traced the vectors and clustered them as they were detected to be a part of one single fiber.

Registration of 3D Volumes: Volume registration is one of the most challenging parts of the above integration since captured data have different spatial resolutions, arbitrarily fine or coarse in each dimension which adds to the complexity of defining a proper quantitative measure for the quality of registration. Some data, such as MRI, are captured at fine spatial resolution

on a regular grid. Some data, are generated from post-processing several views of a tissue, such as the directional diffusion data, which are captured at coarse spatial resolution on a regular grid. On the other hand, there are some data modalities that have different resolution in each dimension, such as histological cross sections. Each single histological cross section is digitized using a bright field optical microscope. These images of cross sections have much finer spatial resolution in x and y dimensions than in z dimension due to slicing limitations. Another major challenge is that the representation of the voxels are not the same among different modalities: some are gray scale, some are RGB colors, and some other are vectors. Moreover, recently, some multi-directional data are generated as well [7].

Correlation of Multi-Modal Features: Once all data modalities have been acquired, reconstructed and integrated into a single system, the main goal is to support understanding of the specimens by correlating multi-modal features. We have proposed a fast measure for the percentage of overlapping between volumes. This measure can either be computed on the volumes as a whole, or on the features detected in different volumes.

The Proposed Integration Framework: Figure 1.2 represents the design of our integration framework for multi-modal medical volumes. Our system supports three major data types: scalar, RGB, and vector. Any scalar voxel is presented by a single attribute; RGB voxels store three attributes, which can be views as a vector from the origin; and vector voxels assign an arbitrary direction to each point in the 3D space. We prototyped the framework into a computer assisted system that could be fully automated as several algorithms for integration would become more robust. As shown in the figure, after the data is acquired, one can reconstruct volumes from 2D images, separated foreground from the background, detected features, and

register volumes such that they all sit in the same coordinate system. Results are visually shown to the user in a single 3D canvas and the user can ask for any quantitative volume comparisons to perform later analysis.

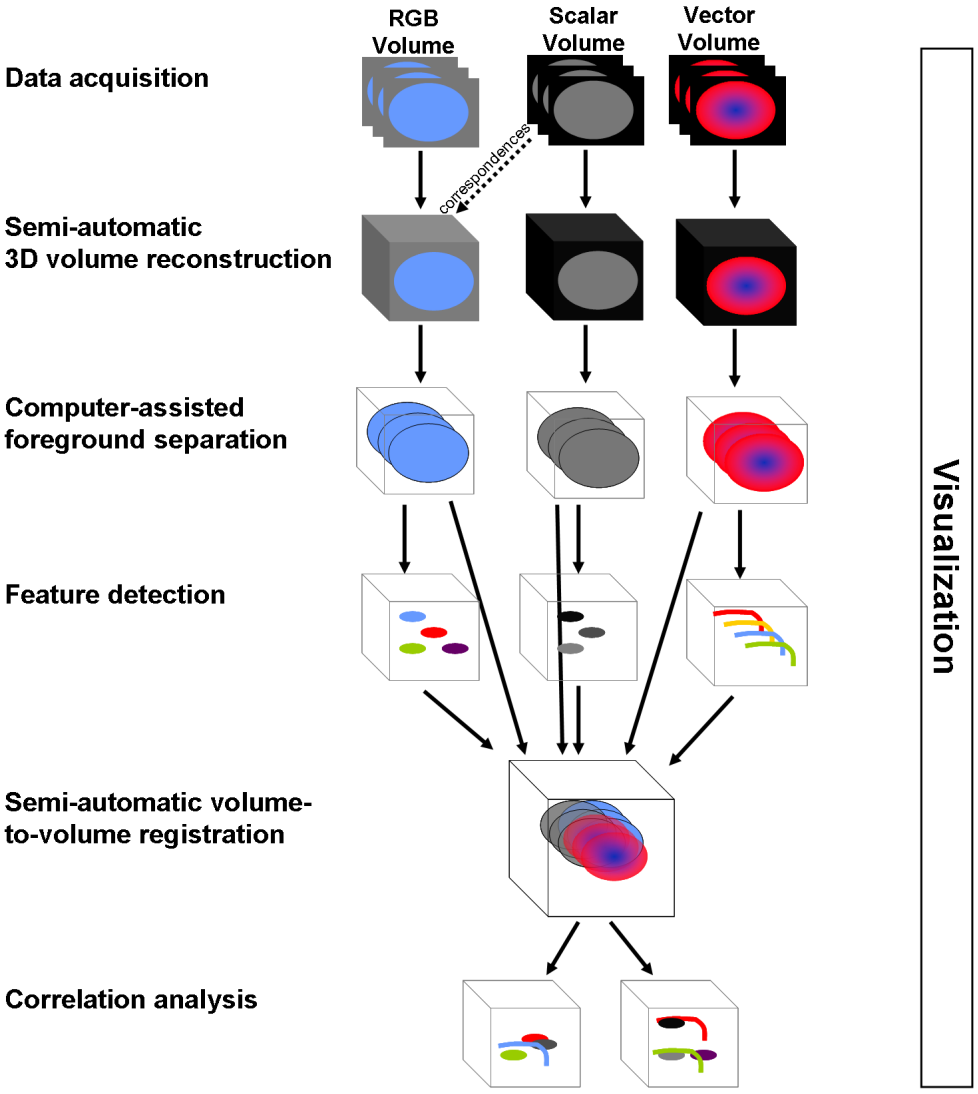

Figure 1.2: The proposed integration framework consists of data acquisition, 3D volume reconstruction, foreground separation, feature detection, and volume-to-volume registration stages.

The integration framework had to address not only the design of the integration sequence of operations but also several basic computer science problems related to: different spatial resolutions, different measurement rep-

resentations (grayscale, RGB, and single or multiple Vectors), very large size data which required significant computational resources and scalable algorithms. Moreover, visualizing several volumes showing different information about the same physical tissue, is usually challenging because not only we were interested in the inner features of the volumes, but also we had occlusion caused by the other volumes.

In order to overcome the above integration challenges, we have proposed a new approach for reconstructing a medical volume from un-aligned images based on its correspondences with other existing volumes. We have also implemented a novel pre-alignment algorithm which registers the bounding ellipsoids of the volumes while maximizing the pair-wise overlapping. Moreover, we have used our fast overlapping measure to evaluate the quality of the pair-wise registration. All of our suggested algorithms are linear in time to the number of voxels in each volume. The linearity not only allows easy and fast modifications to the suggested registration but also makes the algorithms scalable to very large size data.

Chapter 2

Related Work

Over the years many people have worked on suggesting algorithms for integrating data and analyzing the relationships between different data modalities each representing a property of an organ. There are several steps and challenges in finding a good solution to the integration problem. After the data is acquired, if the volumes are defined by a set of unaligned images, a volume reconstruction technique should be used to generate a proper volumetric data. Next, due to the difference in coordinate systems of the volume, they should be registered to each other so that further analysis and comparisons can be performed on the volumes. In order to find the correspondences, one might want to look into the important features of each data modality and compare the detected features against each other and therefore the problem of feature detection in each data modality is also of a special concern to scientists.

This chapter represents previous work on algorithms and integration framework approaches including detection of features and fibers in the volumetric data, and registration of surfaces or point cloud data.

2.1 Feature Detection

Fibers are usually one of the most important features in several tissues. There are a wast amount of work and research on tractography of tissue fibers.

Specifically for finding the brain's white matter detection, Susumu Mori has proposed an algorithm to follow the main diffusion direction in MRI [13]. The algorithm described in this paper, mainly determines the main diffusion direction at each point to be the average of the three principal axes, scaled by the principal eigenvalues λ_1, λ_2, and λ_3. Computed vector field is then processed in order to obtain clusters of connected vectors. An axonal projection is tracks as long as the consecutive vectors are well-aligned until the fiber reaches to an isotropic region. Mori has later written a thorough survey on main algorithms and techniques proposed for fiber tracking in a review article [12]. Over the past years people have used these algorithms to track fibers and visualize the fiber structure and to analyze the correspondences of those structures in order to prove or reject medical hypothesis. As an example, a recent work completed at Stanford University, has used standard tracking algorithms to dynamically visualize pathways of the brain's white matter [1].

2.2 Volume Registration

Most of the shape registration algorithms are based on the Iterative Closest Point step [2]. ICP is so far the most well known algorithm for performing a volume registration. So many people have proposed variants to ICP in order to make it more efficient and faster [14, 4]. Implementing ICP is so simple; however, the efficiency of the algorithm strongly depends on the initial correspondence guess. A main drawback of ICP is that each step of it is relatively expensive when it is running on a large data set. Moreover, finding a good initial correspondence is not always trivial.

Multi-modal volume registration had been one of the most interesting

problems over years. Williams Wells et. al. [16] have proposed a pair-wise registration algorithm by maximizing the mutual information between the corresponding images of different sets. Later on registration algorithms were proposed based on intensity distribution of the images as in the work completed in MIT [9]. Registration of volumes can also be completed by minimizing the squared distance between the underlying surfaces as done by Mitra et. al. [11]. These kind of minimizations are usually done by developing and solving linear systems to find rigid transformations. However, we did not find these approaches suitable for our application since they would be relatively slow for our large size data.

On the other hand, there exist some marker-based algorithms compared to the fully-automated algorithms discussed above. In these algorithms, a number of correspondence pairs have to be marked on the two data sets and then the best transformation from one volume to the other is detected [3]. More recently, Lee and Bajcsy [8] have proposed a centroid-based registration algorithm that would tile large images and registers the corresponding images based on the region correspondences defined by the user.

Rather than finding a perfect registration for pairs of volumes, we preferred finding an approximate initial guess for the registration since this is sufficient for the users in most of the cases. In addition, we provide the user with control widgets to easily modify the suggested registration. People have already developed systems and algorithms for finding such initial alignments. For example, an on-going project in Vanderbilt University is mainly focused on image-guided liver surgery [5, 6]. In this project, system aligns the volumes using the orientation information provided by the user. Pre-alignment is completed using these input data which is later used as an initial guess to a more accurate iterative ICP registration.

Chapter 3

3D Volume Reconstruction

In most medical data modalities, as in MRI and CAT scans for example, acquired images are aligned with respect to the imaged specimens; thus, no volume reconstruction is necessary. However, this is not true for some data modalities, such as histological cross sections. These slices are obtained by embedding the tissue such that it turns into a hard block, then slicing it and staining each slice. Each slice is later placed on a glass slide and digitized for further analysis. Not only that these cross sections might not be parallel, they might even stretch or shrink during this process. If a slice is composed of several disconnected segments, the location of the segments with respect to each other might not be correct after placing the segments on the slice and therefore, slices no more line up with each other. In order to build the 3D model of the tissue one needs to take into account all of these transformations and perform an appropriate 3D volume reconstruction.

We have proposed a technique for reconstructing 3D volumes, from unaligned slices, using a slice-to-slice alignment approach with one of the other existing 3D volumes. In this algorithm, we assume that we have a corresponding volume composed of several images which was captured by another technique. For reconstructing a volume from unaligned slices, we need to find a corresponding slice for each unaligned slice. After finding these correspondences and subtracting the background from the image, we run an alignment algorithm on the slice pairs to line up the unaligned slices with a reference

volume. Once this step is completed, we can simply stack up the slices on top of each other and build the new volume. In the following sections we describe the details of this algorithm.

3.1 Background Subtraction

Captured images in all modalities are rectangular images containing both the foreground (the actual tissue) and the background (the substrate pixels). One needs to run a background subtraction technique to extract the pixels that belong to the foreground. In some modalities such background is usually perfect black or perfect white. Detecting such a background can be easily done by removing black/white pixels.

In this chapter we describe a color-based filtering algorithm for separating the foreground and the background of each slice. This is a user-guided step since the user will define a brightness range for a foreground pixel. Similar ranges can be defined for the value of each R, G, or B channel. All pixels with color values not in the defined range are filtered and the remaining is the foreground. In order to detect the foreground with high accuracy, it is needed to modify the threshold for each data set. As we move the threshold to delete more and more pixels, we might start removing some of the foreground pixels that match the brightness or the color of the background. Therefore, the background subtraction is followed by a hole filling and similarly noise removal steps to produce a proper segmentation of the image.

3.1.1 Color-based Background Subtraction

As mentioned above, we have used a brightness-based filtering to subtract the background from the images. In this approach, we basically define a

lower and an upper threshold for the brightness of a foreground pixel. Any pixel with brightness out of this range is subtracted as a background pixel. Typically, medical images have perfect black or perfect white backgrounds. In these cases, the color-based algorithm performs well. However, in cases where the image is a digital scan of a tissue slice, or when captured image is dark (or bright) and is mapped on a black (or white) background, using this approach is no more reasonable. Using a brightness-based filter will remove some foreground pixels as well that match the brightness of the background.

In order to overcome the above problem, we add a similar threshold for each color channel separately. This will allow a user to filter the pixels based on their R, G, and B values. Having such a tool is useful in cases that the slices are stained, for example with Nissl stain, and therefore most of the pixels of the image have a high value in one channel, blue in this case, and lower values for the other channels. In the other words, if we define the $T_{brightness}$, T_R, T_G and T_B as the foreground ranges for brightness and the RGB channels, the mask can be formed as follows:

Algorithm 3.1.1: FILTER($\mathbf{p}$)

$mask(\mathbf{p}) \leftarrow 1$
if $brightness(\mathbf{p}) \notin \mathbf{T_{brightness}}$
  then $mask(\mathbf{p}) \leftarrow 0$
if $\mathbf{p.R} \notin \mathbf{T_R}$
  then $mask(\mathbf{p}) \leftarrow 0$
if $\mathbf{p.G} \notin \mathbf{T_G}$
  then $mask(\mathbf{p}) \leftarrow 0$
if $\mathbf{p.B} \notin \mathbf{T_B}$
  then $mask(\mathbf{p}) \leftarrow 0$

Moreover, we complete our background subtraction with two other steps. One for removing noise pixels from the background and the other for filling holes in the foreground. Details of these step can be found in the following sections.

In our current implementation, finding satisfying filtering thresholds are done manually by the user. However, one can use histograms of pixel brightness, or different color channels to find a good initial guess for foreground ranges automatically. The user will always gets the chance of modifying these thresholds.

Figure 3.1 shows the results of our color-based background subtraction algorithm for three different data modalities of the rat brain.

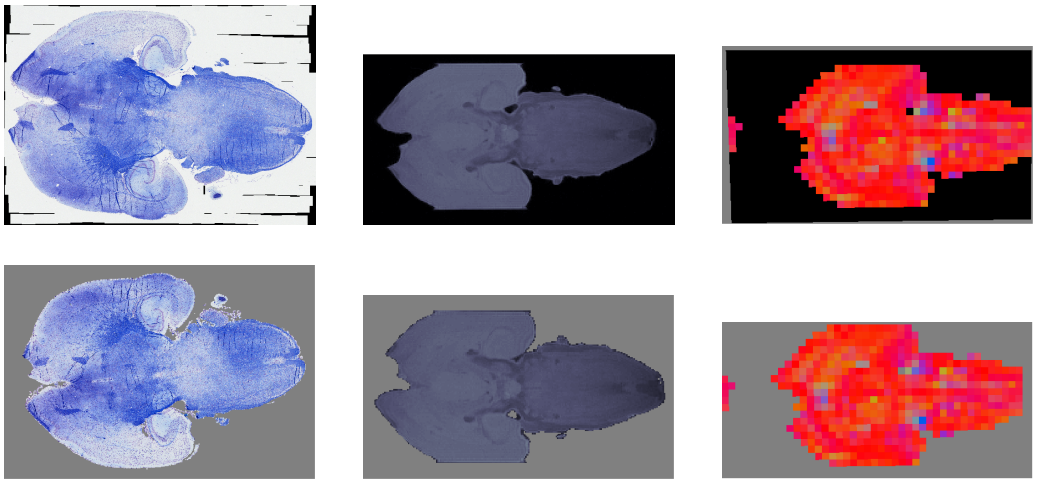

Figure 3.1: Original images vs. background subtracted images of a histology cross section, an MRI as well as a DTI slice.

3.1.2 Noise Removal

After performing a color-based background subtraction the resulting image might contain some noise in the areas with most of their pixels detected as background. These noises are usually either single points or very small connected segments. In some cases we can assume that the foreground is

just one single connected segment and therefore remove all the other smaller segments from the foreground. However, this is not true for all the images. There are significant number of tissues that contain multiple segments in their parallel slices. Therefore, we can remove most of the noise by defining a threshold on the minimum number of pixels of each foreground segment. Having such a parameter, we can then remove all small components and return as a result a more accurate foreground.

3.1.3 Hole Filling

Once we have subtracted the background by a color-based filter, we might have introduced some holes to the foreground as well. These holes are introduced since in most of the images, the brightness or the color of foreground pixels cover wide ranges. We can take a similar approach as for the noise removal to fill-in these holes. Having a threshold on the minimum number of pixels of a hole, we can detect small holes and remove them from the foreground.

3.2 Volume Reconstruction

In the previous section we discussed the challenges for finding a good foreground from the captured images. Assuming that all the remaining voxels are foreground voxels, this section describes our proposed algorithm for reconstructing the 3D volume from a set of images that are not well-aligned and parallel. This algorithm is based on the assumption that there exist another 3D volume of the same tissue that can be used for reconstruction. This is however mostly true since capturing parallel and aligned data is mostly both faster and cheaper. This algorithm has two main steps. First, the

correspondences are resolved between the slices of the new volume and the existing one manually. Then, an alignment process registers each slice to its corresponding slice semi-automatically.

Figure 3.2 illustrates different steps of our reconstruction algorithm. For each acquired slice, the closest slice in a target volume is detected, the slice is then rigidly registered to the target slice after defining manual landmarks and finally, the slices are stacked to build the final volume.

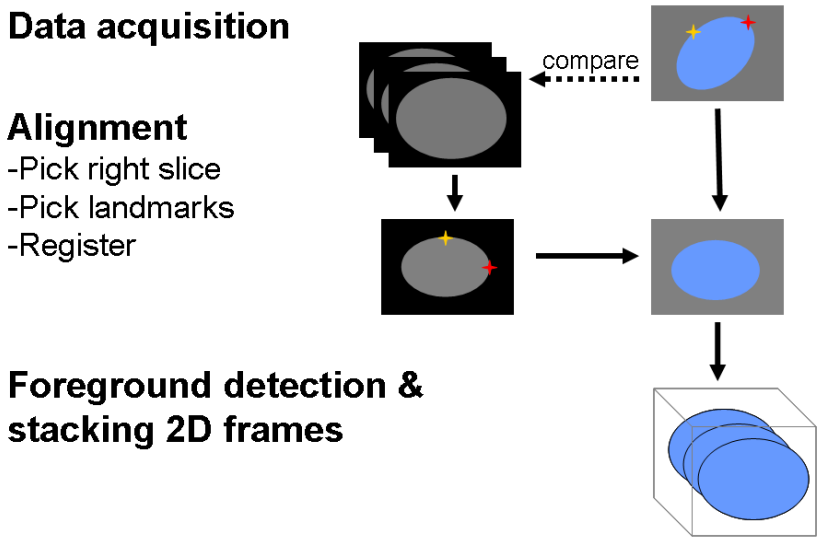

Figure 3.2: This figure shows the different steps of our 3D volume reconstruction algorithm.

3.2.1 Slice Correspondence

The first step of our volume reconstruction algorithm is to find the corresponding slices in another volume. This is a challenging step, since the spacing between the slices are not the same in different modalities. Also, in a data set such as the histology that the tissue is sliced by hand, the spacing might vary from slice to slice. If we had the correct and exact spacings, in

all data modalities, and many slices in the existing target volume, finding one slice correspondence would have been sufficient. However, the two main issues stay into effect. To solve the slice correspondence problem, for now, we have simply picked the visually closest slice from the target volume manually. In none of our data sets, we did not have an adequate number of slices in the target volume; therefore, no perfect correspondence was achieved. Still, the results will show that the final reconstructed 3D volumes are well-aligned.

3.2.2 Slice Alignment

Slice alignment is the next step of our volume reconstruction. Once we finish this step, the volume can be represented by stacking up the aligned slices. We have tried several slice alignment approaches. We had to decide on two major steps of our slice alignment. The first choice was to decide using either *Rigid* or *Affine* transformation. And, the second choice was to decide whether we would want to align the individual slices to their corresponding slices in the target volume, or to align one slice of the new volume with the target volume and then align the other new slices with that reference slice.

Figure 3.3 shows some results from rigid and affine transformations of one single cross section of the rat brain.

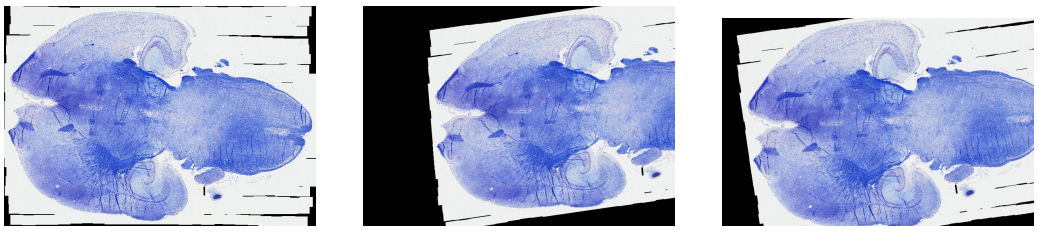

Figure 3.3: This figure shows one original histological cross section of the rat brain (left) a comparison of applying rigid (center) or affine (right) transformations on it.

We tried all of the following possibilities. By comparing the resulting

volume with other existing volumes, we decided to transform slices rigidly such that each individual slice is registered with its corresponding slice in a target volume.

Final results of our rat brain volume reconstruction is shown in Figure 3.4 which shows the background subtraction for MRI and DTI volumes and the reconstructed histological volume with respect to the MR volume.

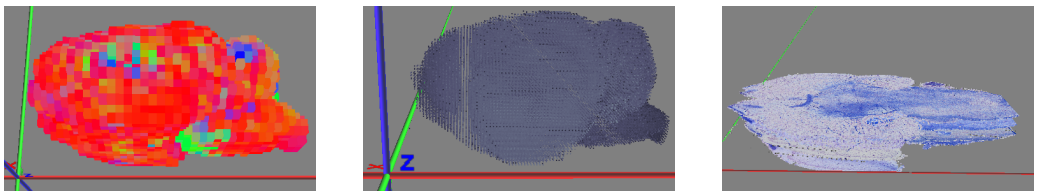

Figure 3.4: This picture shows final results of our volume reconstruction algorithm on three rat brain modalities. Images are the background subtracted DTI volume(left), background subtracted MRI volume(center), and the registered, background subtracted histology volume(right).

Chapter 4

Feature Detection in 3D Volumes

Medical experts often search for patterns and features in the medical volumes. For instance, in the brain data myelinated fibers are of high interest to medical doctors. Thus, the problem of detecting and comparing features in different modalities have been addressed by many scientists [1, 13, 12].

Solution to the feature detection problem is different from modality to modality. In this chapter we talk about the algorithms and user-assisted approaches that we have developed in order to find the features in each of the commonly used data modalities: grayscale, color, and directional.

4.1 Feature Detection in Scalar Volumes

In grayscale images the features are can be identified by the change of intensity. For example, the voxel intensities in MR images reflects amount of existing water at each point of the tissue. Soft tissues are usually lighter in these images, and harder organs such as bones are not at all visible in these images. On the other hand, CT images show harder tissues such as bones brighter and the soft tissues are dark in those images. Therefore, we can use our color-based filtering approach to filter pixels based on their brightness and extract features with desired intensity.

In the example of the brain data, based on structural and functional knowledge about brain, water flows in brain's fibers. Therefore, we are in-

terested in finding these tracks which are formed by the bright pixels and correspond them with the similar tracks in other data modalities. A recent work from Stanford University [1] talks about algorithms for finding fibers by exploring the brain's white matter pathways. They have used standard tractography methods to detect the fibers. These methods either follow the first principal diffusion direction or use a tensor defined on each voxel and multiply the incoming vector by the tensor to compute the outgoing vector at that voxel. However, these methods are based on knowing the diffusion direction on the volume. The directional data is usually of a coarse spatial resolution. Here, we have used the brightness-based filtering algorithm to detect the fibers in MR images. Figure 4.1 shows the result of our feature detection algorithm on an MRI data set from a rat brain.

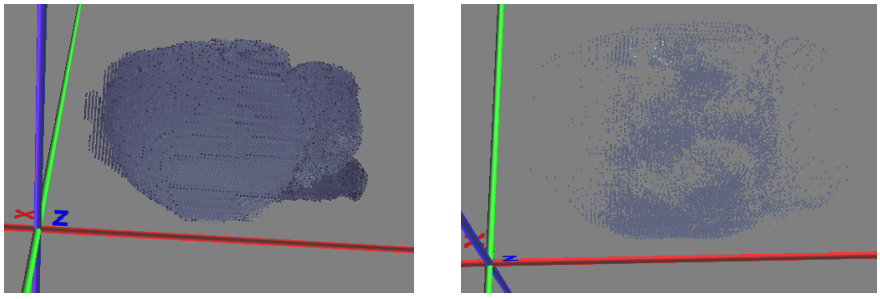

Figure 4.1: Original MR volume (left) and the detected features (right).

4.2 Feature Detection in RGB Volumes

As described earlier, final images of some data acquisition processes are colored-data. For example, when a brain tissue is sliced and stained using luxol fast blue solution, myelinated fibers, which are the important features in these tissues are dark blue. As another example, a lung histological cross section, shown in figure 6.1, is stained such that features are dark pink or red. Using our system, the user can define simple filters on either the brightness

of each pixel or on the magnitude of each color band. Using these filters user can extract features in each colored image. Figure 4.2 shows the result of the feature detection in the histo volume constructed from the four available slices of the rat brain.

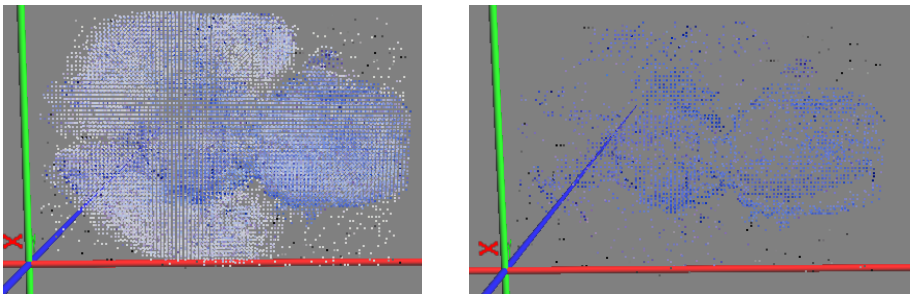

Figure 4.2: This is an image of a histological cross section (left) and the result of using the color-based filtering for detecting the fiber features (right).

4.3 Feature Detection in Vector Volumes

Some acquired data express directional (vector) data. For example, in our brain data set, DTI data are defining the major water diffusion directions at each voxel which is believed to be in the same direction as the myelinated fibers.

Mori [12] has done a thorough survey on fiber tracking algorithms in vector volumes. Some existing algorithms use a line propagation technique. Features are followed until they reach an anisotropy region or a big angle change happens in their direction. There are some energy minimization techniques which can find paths from an arbitrary point to the center by following the gradient of steepest paths. Some algorithm use tensor lines or surface-line techniques which deflect the path using the diffusion ellipsoids at each point instead on the diffusion direction.

Here we propose a fast line clustering algorithm for finding fibers in these

data sets.Main steps of this clustering algorithm are outlined in algorithm 4.3.1. Consider the vector field defined on a set of points $p \in P$ as $d(p)$.

Algorithm 4.3.1: SetLabel()

for each p

do $label(p) \leftarrow -1$

$l \leftarrow 1$

repeat

$p_0 \leftarrow$ *an unlabeled voxel*

$label(p_0) \leftarrow l$

$p \leftarrow p_0$

repeat

$p_1 \leftarrow$ *closest voxel to* $p + d(p)$

if $label(p_1) < 0$ *and* $d(p).d(p_1) > \theta$

then $label(p_1) \leftarrow l$, $p \leftarrow p_1$

until p *is modified*

$p \leftarrow p_0$

repeat

$p_1 \leftarrow$ *closest voxel to* $p - d(p)$

if $label(p_1) < 0$ *and* $d(p).d(p1) > \theta$

then $label(p_1) \leftarrow l$, $p \leftarrow p_1$

until p *is modified*

$l \leftarrow l + 1$

until $\forall p, label(p) > 0$

Figure 4.3 shows one slice of the DTI of the rat brain and the result of the fiber tracking algorithm on this data.

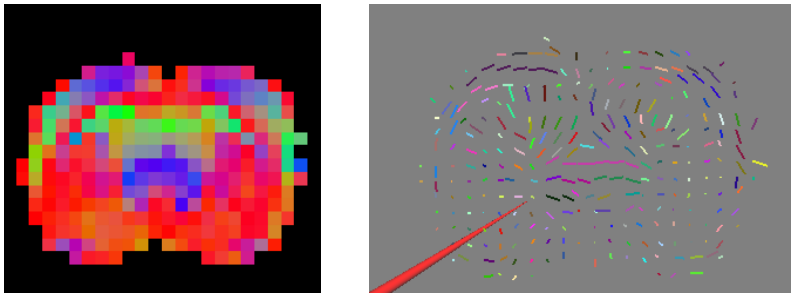

Figure 4.3: This figure shows one DTI slice from the rat brain (left) and result of our fiber tracking algorithm (right).

Chapter 5

3D Volume Registration of Multi-Modal Data

For registering two volumes against each other, at the beginning our system suggests an initial registration. We also provide the user enough controls to modify our initial registration. Basic transformation operations, translation, scaling, and rotation, are allowed on each volume. We define the final transformation matrix based on three vectors. Translation is defined with three parameters, *x_offset*, *y_offset*, *z_offset*. Scale parameters are *x_scale*, *y_scale*, *z_scale*. Rotation is defined with three rotation angles, θ_x, θ_y and θ_z about axes parallel to x, y and z axes passing through the centroid of the object. In this chapter, we explain an automatic pre-alignment algorithm that gives a user an initial registration suggestion. However, since the user might need to make slight modification to the final result, we also provide the user with adequate number of control buttons to modify this initial suggestion.

5.1 Properties of 3D Volumes

5.1.1 Centroid

We define a centroid of an object to be its center of mass. Centroid can be computed by simply averaging the values in each of the three dimensions separately and finding the average for x, y, and z value. Considering n to denote the number of visible voxels, calculation of the centroid $\mathbf{c}$ can be

formulated as below:

$$\mathbf{c} = (\tfrac{1}{n}\Sigma x_i, \tfrac{1}{n}\Sigma y_i, \tfrac{1}{n}\Sigma z_i)$$

5.1.2 Bounding Ellipsoid

Bounding Ellipsoid of a 3D volume can be defined from its covariance matrix. We have used the definition of the covariance matrix described in [15, 10] and extended it to 3D. For three dimensional point sets the covariance matrix is a 3×3 matrix given by:

$$M = [m_{ij}]$$

Where

$$m_{ij} = \tfrac{1}{N-1}\Sigma_{k=1}^{N}(x_{ik} - \bar{x}_i)(x_{jk} - \bar{x}_j)$$

Bounding ellipsoid of each volume is an ellipsoid whose major axes is the eigenvectors of the covariance matrix scaled by their corresponding eigenvalues. We define the major axes and scale of the 3D volume to be the same as the bounding ellipsoid and explain in the following section how we use this information to find the best transformation from one 3D volume to the other.

5.2 Primitive Transformations

5.2.1 Centroid-based Translation

In order to register two volumes, the first primitive transformation is to translate one volume and locate it at the position of the other one. In order to achieve this goal, we compute the centroids of two volumes as mentioned

in the previous section. Then, all the pixels of the source volume is translated by a vector connecting the two centroids. If $\mathbf{c_1}$ and $\mathbf{c_2}$ are centroids of the two point sets P_1 and P_2, the translation matrix T from P_1 to P_2, can be defined as:

$$T = \mathbf{c_2} - \mathbf{c_1}$$

and thus all the points $\mathbf{p} \in P_1$ have to be translated using:

$$\mathbf{p} = \mathbf{p} + T$$

5.2.2 Bounding Ellipsoid Based Scaling

Scaling is another necessary primitive transformation in order to register two volumes with each other. For scaling a volume such that it has almost the same size of a target volume, we compute the bounding ellipsoids of the two volumes. Assuming that the volumes have almost the same shape, we calculate scaling factor being the average scaling factor in all three major directions:

$$s = \tfrac{1}{3}\Sigma_i \sqrt{\lambda_{2i}/\lambda_{1i}}$$

and then P_1 is scaled by applying the scale factor to all its points using:

$$\mathbf{p} = s.\mathbf{p}$$

5.2.3 Bounding Ellipsoid Based Rotation

Once the volumes are scaled and translated to the same location in space, we have to rotate them such that their bounding ellipsoids match. One can match two ellipsoids easily by transforming one such that its first and second major axis align. There might be some cases however, that although one

shape is flipped and has not matched the other one, the bounding ellipsoids are matching. Figure 5.1 shows an example of such a non-accurate rotation.

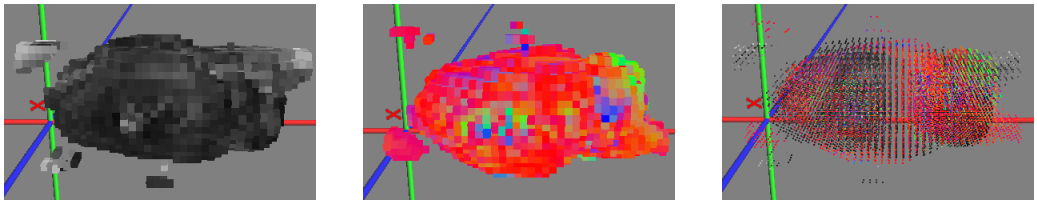

Figure 5.1: This figure shows two modalities of the rat brain (left,center) and the result of volume registration (right) when rotation does not maximize overlapping.

In order to overcome this problem we consider all possible rotations that map the major axes of one volume to the ones of the other volume. Our bounding ellipsoid based algorithm rotates the source volume with each of the possible rotations and calculates the overlapping of the volumes. The rotation which results in the maximum overlapping is then selected as the best rotation. Figure 5.2 is the result of the well-rotated volume for the same data sets shown in figure 5.1. Once the correct rotation angles $\Theta = (\theta_x, \theta_y, \theta_z)$ are detected, we form the rotation matrix to be:

$$R = R_z(\theta_z)R_y(\theta_y)R_x(\theta_x)$$

and then we can rotate P_1 by rotating each single pixel in it using:

$$\mathbf{p} = R.\mathbf{p}$$

5.3 Automatic Pre-alignment

In this section we explain our pre-alignment algorithm which can be used for computing an automatic initial transformation to register two 3D volumes. We have implemented a fast 3-D to 3-D volume registration which maps the

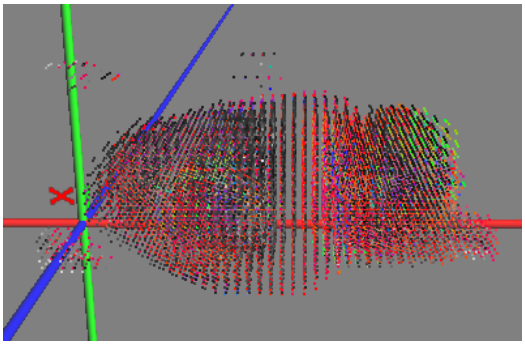

Figure 5.2: This figure shows the rotation of two volumes based on matching their bounding volumes while the overlapping metric is maximized.

bounding ellipsoid of volumes such that the overlapping of one volume with the other one is maximized. The evaluation of the registration is done by computing the overlapping of the volumes. Details of this metric can be found in the following section.

Our pre-alignment algorithm uses the centroids and bounding ellipsoids of two point sets P_1 and P_2 to calculate the volume transformation from the first one to the other. First, the scaling factor is calculated as described above. Then, P_1 is translated to the centroid of the other volume. Finally, the best rotation of the major axes is chosen and applied to P_1 such that the overlapping of the volumes with respect to each other is maximized. If S, T, and R are scaling, translation and rotation matrices defined as above, one can extract the final transformation matrix using the following equation:

$$M = RTS$$

We have used the following algorithm to apply the same transformation on the point set, since this would consume less operations in order to complete the transformation compared to the matrix multiplication.

Algorithm 5.3.1: REGISTER(P_1, P_2)

$C_1 \leftarrow CovarianceMatrix(P_1)$

$C_2 \leftarrow CovariancetionMatrix(P_2)$

$\mathbf{c_1} \leftarrow Centroid(P_1)$

$\mathbf{c_2} \leftarrow Centroid(P_2)$

$s \leftarrow AverageScale(C_1, C_2)$

$Scale(P_1, s)$

$Translate(P_1, \mathbf{c_2} - \mathbf{c_1})$

$overlapping \leftarrow -1$

for each *possible rotation*

do
- $Rotate(P_1, \Theta)$
- $new\_overlapping \leftarrow Overlap(P_1, P_2)$
- **if** $new\_overlapping > overlapping$
  - **then**
    - $overlapping \leftarrow new\_overlapping$
    - $\Theta_{final} \leftarrow \Theta$

$Rotate(P_1, \Theta_{final})$

5.4 Registration Quality Evaluation

Once we have registered volumes or after the initial volume registration, there is a need to measure the quality of the registration. Here we propose an overlapping measurement algorithm that encounters the overall overlapping of the volumes. The result would be two quantities in percent each of which represent an overlapped percentage of each volume with the other. Our proposed algorithm, outlined bellow, is linear in the sum of visible voxel count of both volumes.

5.4.1 Overlapping Metric

Consider two 3D volumes P_1 and P_2 and their corresponding transformation matrices M_1 and M_2. The volume percentage of P_1 overlapped with P_2 can be computed using a simple algorithm below.

Algorithm 5.4.1: OVERLAPPED(P_1, P_2)

$overlapped\_counter \leftarrow 0$

for each $\mathbf{p} \in P_1$

do
- **comment:** transform $\mathbf{p}$, then move it to the coordinate system of P_2
- $\mathbf{p_1} \leftarrow M_2^{-1} M_1 \mathbf{p}$
- $\mathbf{p_2} \leftarrow$ *a voxel in* P_2 *such that* $\|\mathbf{p_2} - \mathbf{p_1}\|$ *is minimized*
- **if** $visible(\mathbf{p_2})$
  - **then** $overlapped\_counter++$

return $(overlapped\_counter / visible\_count(P_1))$

Chapter 6

Results

In this chapter we show the results of using our framework in order to integrate multi-dimensional multi-modal data. We illustrate figures that represent different steps of our different integration algorithms. We have used different data sets such as the human brain, the rat brain and some lung data. Data modalities are obtained either from a brain or lung tissue. Data modalities varies in each data set. We have MRIs, microCT, and neutron beam images which are all grayscale, histological cross section which are RGB images, and DTIs which are directional data.

6.1 Integration of 2D Images

One can use our system to integrate and analyze the correspondence between a set of multi-modal corresponding 2D images. As an example, we have run the system for co-registering different image modalities of a lung tissue. Corresponding slices of microCT, a histological cross section and an image acquired by neutron beam imaging from the lung tissue are shown in figure 6.1. Figure 6.2 shows all three modalities after background subtraction. Images are not registered to each other. Three images, shown in a single 3D canvas, do match neither in their resolution nor orientations.

Figures 6.3 and 6.4 shows all the integration results. First row illustrates the pair-wise registrations of the images and the second row shows the vi-

sualization of all three data modalities in a single canvas. Quality of the registrations can be evaluation by overlapping shown in table 6.1. Each entry of the table, t_{ij} shows what percentage of the image of row i is overlapped with image of column j.

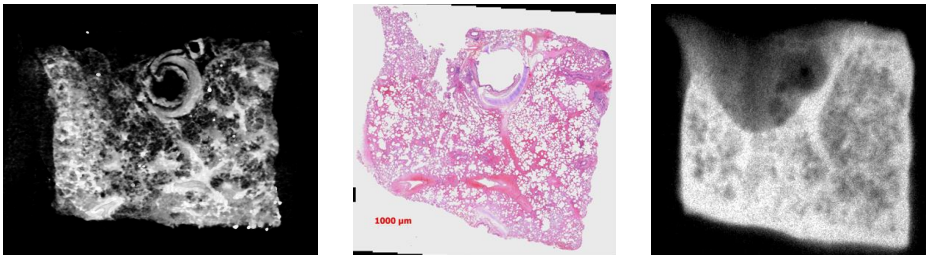

Figure 6.1: Images in this figure are microCT (left), histology (center), and the neutron beam image (right) of a lung tissue.

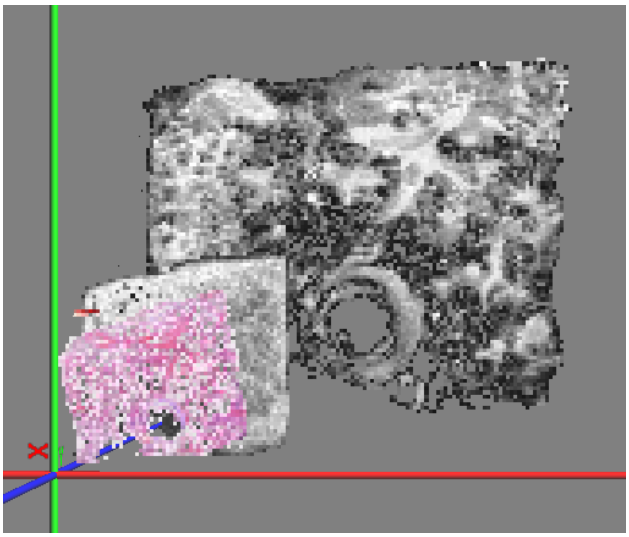

Figure 6.2: Unregistered images acquired from a lung tissue after background subtraction.

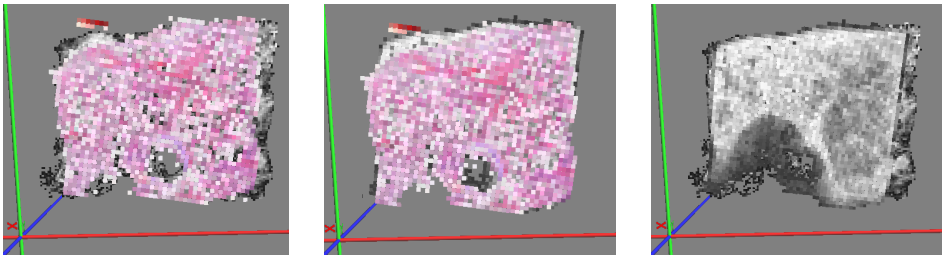

Figure 6.3: This figure shows pair-wise registration results for lung images. Registration results show registered histology and microCT (left), histology and neutron beam image (center), and microCT and neutron beam image (right)

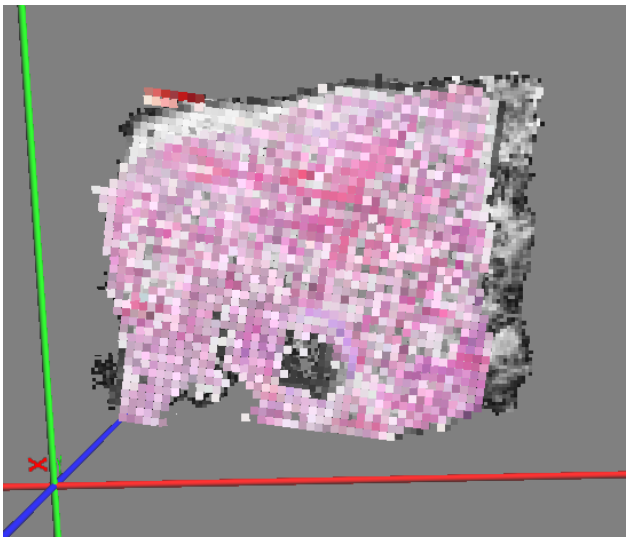

Figure 6.4: This figure shows the final integration result of all different lung modalities.

	MicroCT	Neutron Beam	Histology
MicroCT	%100	%83	%79
Neutron Beam	%89	%100	%85
Histology	%88	%89	%100

Table 6.1: Computer generated assessment of 3D-to-3D volume registration quality by computing the pair-wise volume overlapping according to algorithm 5.4.1

6.2 Integration of 3D Volumes

We had the most complete data set for the rat brain. Available data sets for this tissue were: fine spatial resolution MR images, coarse spatial resolution DTI, fractional anisotropy of the water diffusion, as well as four histological cross sections. Figure 6.5 shows one slice of histology, a single slice of the fine spatial resolution MR, and their corresponding directional diffusion slice. Background subtraction is applied to all slices in all three series of images. Histological cross sections are aligned as described in Chapter 5 and the slices are stacked up to form the corresponding 3D volumes. Result of the volume reconstruction for all three modalities are shown on right in that figure. We have also used our implemented algorithms to detect features in each of the above modalities. Figure 6.6 illustrates the feature detection results in Histoology, MR and DTI volumes of the rat brain.

Our integration framework can be used to open several modalities and analyze their correlation. Figure 6.7 shows the results of our volume registration to integrate different modalities. Integration of the MR and histology volumes are shown on left and the image on the right integrates one slice of the DTI with the integrated result of the two other volumes. We can see that a set of fibers colored as blue in the DTI is matching with a dark blue region of the histology volume.

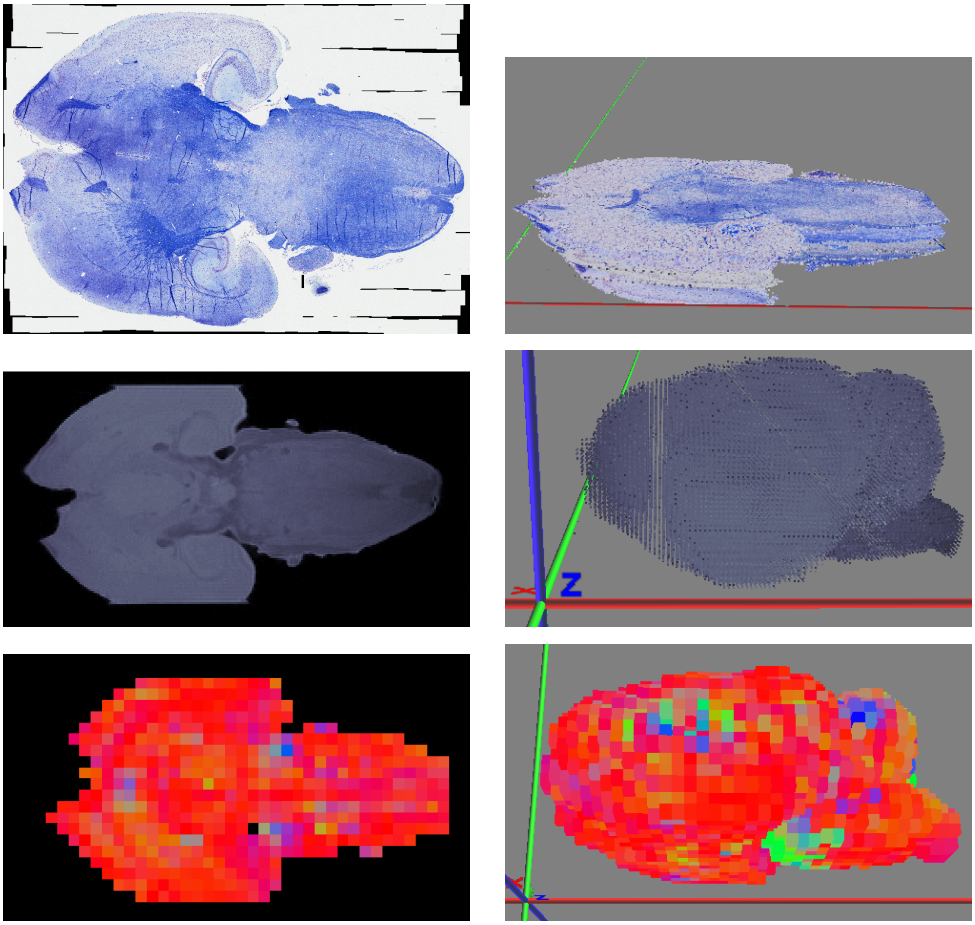

Figure 6.5: Original slices of the rat brain are shown on left: Histo (top), MRI (center), DTI (bottom). Corresponding reconstructed volumes for each data modality is shown on right.

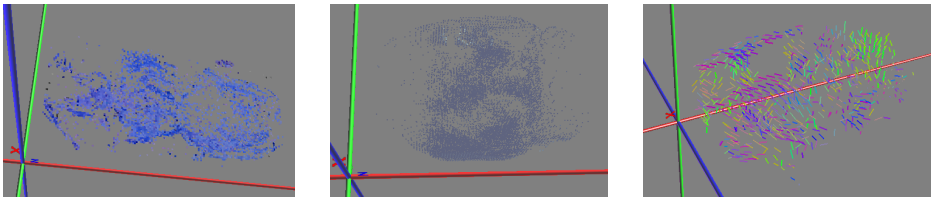

Figure 6.6: Results for detecting features in the rat brain: Histology features (left), high resolution MR (center), directional water diffusion (right).

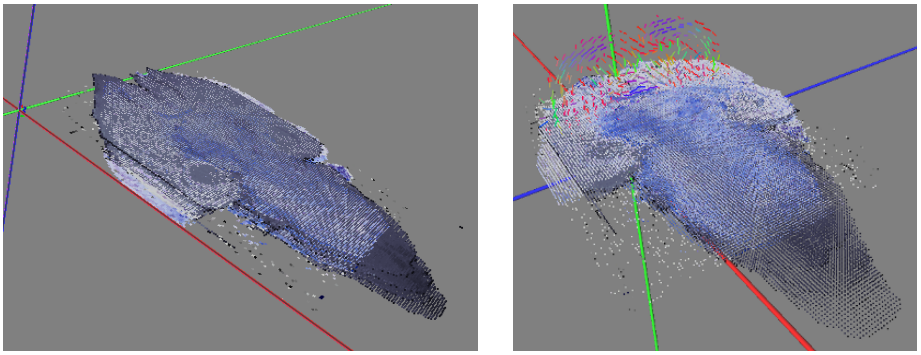

Figure 6.7: Integration results of rat brain data. Four slices of histological cross sections and the corresponding region from the MR volume are first integrated (left). DTI volume is registered to the MR volume as well and the final integration is shown by visualizing one slice of the DTI (right). Notice the blue feature detected both in histology and DTI.

Chapter 7

Conclusion

In this thesis we have designed and implemented a framework for integrating multi-modal 3D medical volumes. Different data modalities differ from each other in their spatial resolutions in every dimension and also in the physical meanings of the measurement at each voxel. In order to complete an integration, volumes were reconstructed from 2D cross sections. Moreover, we prototyped detection of important features in each data modality that was used for 3D volume registration and later for volume correlation analysis. The 3D registration step was supported by a computer assessment of the registration quality that guides the end user with a better understanding of the 3D volume registration uncertainties.

All of our proposed algorithms could be used on either 2D or 3D data sets. They were designed to be scalable with the data size and could be executed on a PC. The developed framework has been viewed by medical users as an enabling technology for better understanding the correlation of multi-modal data.

Appendix A

Graphical User Interface

Along with the development of our volume reconstruction and registration, and visualization tool, we have also implemented an easy-to-use graphical user interface which enable the medical user to access functionalities of the integration framework. This section mainly lists the features we have provided to a user via our GUI. Figure A.1 shows the implemented GUI with two open volumes.

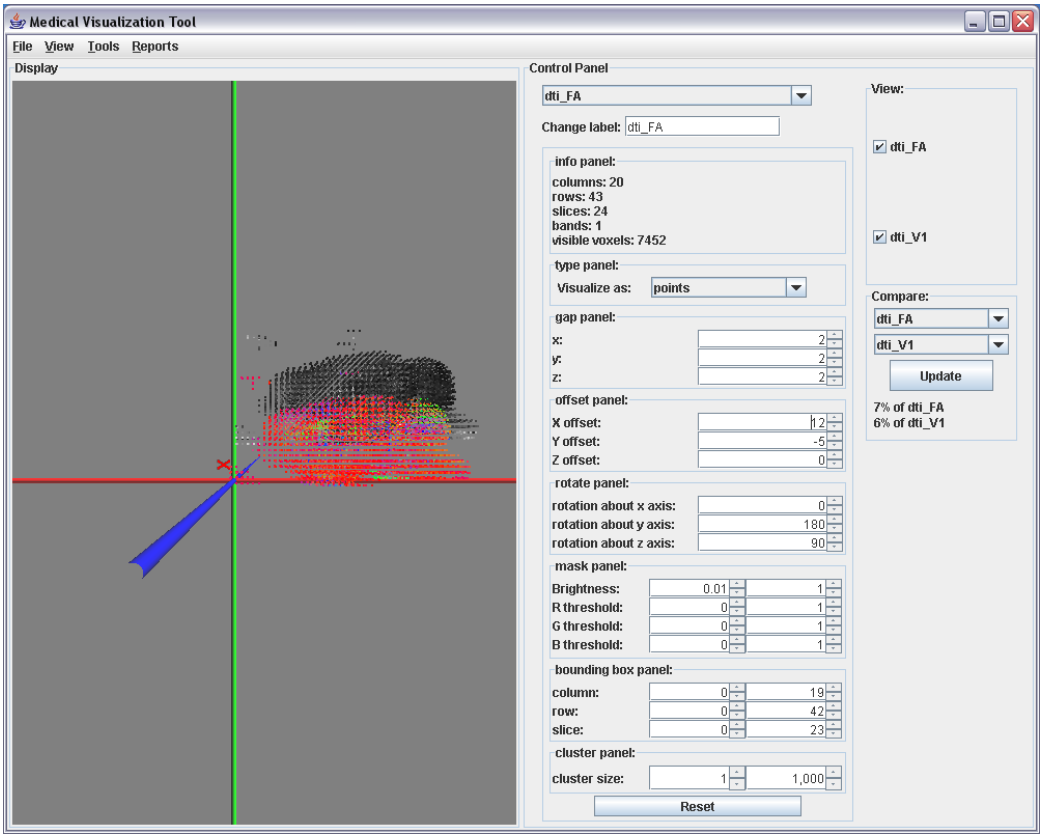

Figure A.1: An overview of the implemented Graphical User Interface

A.1 Open Multiple Volumes

Our tool support opening volumes defined by a series of parallel images, as well as some 3D volume data formats. The user is able to open one or multiple volumes. Each volume might contain one or several slices of the tissue. The user would then have to perform a desired filtering on each data set separately to obtain a good foreground volume, good features, or the region of interest. Once the filtered volumes are available, one can run a registration or comparison algorithm on pairs of volumes to align them on top of each other and detect the correlation between them. The following sections describe the user interface provided to the user both to control the visibility and the type of a single volume as well as the to run the pair-wise algorithms.

A.2 Volume Control Panel

The GUI contains a control panel for each volume. Volume-specified properties can be modified via this panel. This section contains a listing and some details of what modifications a user can perform on each single volume.

A.2.1 Information Panel

As shown in Figure A.1, we have an information panel on top of the control panel for the selected volume that displays information such as number of column, rows, and slices of the volume and number of bands per each voxel as well as the number of visible voxels.

A.2.2 Change Label

In some of the existing medical programs, volumes are named by the name of the first file in the series. However, it is always useful for the user to rename the data sets with a name that represents the volume to him/her. Therefore, the control panel for a single volume starts with a text box for modifying the name of the selected volume.

A.2.3 Select Visualization Mode

There are several types of visualizing a 3D voxel. We have provided a pixel view, a 2D patch view parallel to any of the XY, XZ, or YZ planes, as well as the cube view. Since some data might represent a directional information, a user can also choose to see a vector field view of the data or the fiber tracks which can be formed based on an following the flow in a vector field data.

A.2.4 3D Transformations

We have provided the user a control panel that allows him/her to transform the volume in 3D space. Controls for all three basic transformations are available to the user. The final transformation is then completed by applying the scaling and the rotations first, and then by translating the volume to the desired location in the space.

Translation: In order to support translation, we have defined an *offset* to the centroid of the volume. At the beginning, the centroid is placed on the origin and can be moved later in the 3D space.

Rotation: Volume rotation is defined by three rotation angles, one about each major axes. The rotation is then completed by applying the following three rotations: $R = R_z(\theta_z)R_y(\theta_y)R_x(\theta_x)$

Scaling: We scale the volumes by modifying the side lengths of each voxel. Voxels are not necessarily cubes because the acquisition resolution is not always the same for all data modalities. However, we usually know the size of each voxel in microns and can adjust the scale to represent the tissue with its correct scalings.

A.2.5 Filtering Visible Voxels

There are several reasons why a user might want to filter some of the visible voxels to see the more important voxels more clearly. Below are the filters we have found useful to the medical society and thus have implemented.

Color-based Filtering

As mentioned in chapters 3 and 4 color-based filters can be used to separate foreground and backgrounds, as well as for finding some features in the volumes. Therefore we have provided the required control for filtering with the brightness or with the value of each color channel.

Bounding Box-based Filtering

We included a bounding box filter that filters layers from the rows, columns or slices of each volume. This will be used more frequently if one data set is a subset of another or if one wants to find a correspondence between a small region of one volume and another volume.

A.2.6 Filtering Fiber Clusters

As explained before, we can track the vectors defined on each voxel and detect a fiber structure. This will result in a clustering of the vector field

where each cluster is representing a single fiber. Usually, there are a lot of small clusters passing through only a few voxels. A simple filter on the minimum (and a maximum, in some cases) would be used to visualize the more important regions and fibers.

A.3 Volume Visibility Selection

Once we have several data sets open, one might want to select to visualize some but not all of the open volumes. The GUI supports this feature by easily checking/unchecking a check button and visualize the volumes based on this flag.

A.4 Volume Registration

One can choose to perform an automatic primitive transformation which can be either scaling, translation, or rotation. Moreover, our automatic registration algorithm, described in chapter 5, can be run on two open volumes. It first scales the source volume to match the scaling of the target volume. Next, the volume is translated such that the two volumes are co-centered. Finally, a rotation occurs that maximizes the overlapping of the volumes is achieved.

A.5 Volume Comparison

A comparison tool is frequently used to compare the quality of the registration and the correlation between the different volumes. Therefore, we have provided a panel on the main panel for volume comparison. Source and target volumes are selected from a combo-box list and the selected volumes are

compared against each other. We do not update the result for comparison lively since it makes the interaction for transformations slower. Moreover, one of the selected volumes might not be visible any more and thus the live comparison would be of no use to the user. So, it is updated as user demands.

References

[1] David Akers, Anthony Sherbondy, Rachel Mackenzie, Robert Dougherty, and Brian Wandell. Exploration of the brain's white matter pathways with dynamic queries. In *VIS '04: Proceedings of the conference on Visualization '04*, pages 377–384, Washington, DC, USA, 2004. IEEE Computer Society.

[2] Paul J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[3] M. Capek, R. Wegenkittl, A. Jaschke, R. Sweeney, and R. Bale. Multimodal medical volume registration based on spherical markers. *9-th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, 2001.

[4] D. Chetverikov, D. Svirko, D. Stepanov, and Pavel Krsek. The trimmed iterative closest point algorithm. In *ICPR '02: Proceedings of the 16 th International Conference on Pattern Recognition (ICPR'02) Volume 3*, page 30545, Washington, DC, USA, 2002. IEEE Computer Society.

[5] L. W. Clements, D. M. Cash, W. C. Chapman, R. L. Galloway, Jr., and M. I. Miga. Robust surface registration using salient anatomical features in image-guided liver surgery. In K. R. Cleary and R. L. Galloway, Jr., editors, *Medical Imaging 2006: Visualization, Image-Guided Procedures, and Display. Proceedings of the SPIE, Volume 6141, pp. 105-115 (2006).*, volume 6141, pages 105–115, March 2006.

[6] Ishita Garg, Logan W. Clements, and Jr. Robert L. Galloway. A computational approach to pre-align point cloud data for surface registration in image-guided liver surgery. volume 6509, page 650932. SPIE, 2007.

[7] D. Karampinos L. Ciobanu B. P. Sutton Z.-P. Liang J. G. Georgiadis L. G. Raguin, D. Hernando. Quantitative analysis of q-space mri data. In *Proceedings of the 3rd European Medical & Biological Engineering Conference (EMBEC'05)*, 2005.

[8] Sang-Chul Lee and Peter Bajcsy. Feature based registration of fluorescent lscm imagery using region centroids. In J. M. Fitzpatrick and J. M. Reinhardt, editors, *Medical Imaging 2005: Image Processing.*, volume 5747 of *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference*, pages 170–181, April 2005.

[9] Michael E. Leventon and W. Eric L. Grimson. Multi-modal volume registration using joint intensity distributions. In *MICCAI '98: Proceedings of the First International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 1057–1066, London, UK, 1998. Springer-Verlag.

[10] Shan Lu, Gabriel Tsechpenakis, Dimitris N. Metaxas, Matthew L. Jensen, and John Kruse. Blob analysis of the head and hands: A method for deception detection. In *HICSS '05: Proceedings of the Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05) - Track 1*, page 20.3, Washington, DC, USA, 2005. IEEE Computer Society.

[11] Niloy J. Mitra, Natasha Gelfand, Helmut Pottmann, and Leonidas Guibas. Registration of point cloud data from a geometric optimization perspective. In *SGP '04: Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 22–31, New York, NY, USA, 2004. ACM.

[12] S. Mori and P.C.M. van Zijl. Fiber tracking: principles and strategies: a technical review. *NMR in Biomedicine*, 15(7-8):468–480, 2002.

[13] Susumu Mori, B. J. Crain, V. P. Chacko, and P. C. M. van Zijl. Three dimensional tracking of axonal projections in the brain by magnetic resonance imaging. *Ann Neurol*, 45:265–269, 1999.

[14] Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the ICP algorithm. In *Proceedings of the Third Intl. Conf. on 3D Digital Imaging and Modeling*, pages 145–152, 2001.

[15] Harini Veeraraghavan, Osama Masoud, and Nikolaos P. Papanikolopoulos. Managing suburban intersections through sensing. Technical Report CTS 02-07, Department of Computer Science and Engineering, University of Minnesota, 2002.

[16] W. Wells, P. Viola, H. Atsumi, S. Nakajima, and R. Kikinis. Multimodal volume registration by maximization of mutual information. *Medical Image Analysis*, 1(1):35–51, 1996.