

Abstract

Implementation of automated volume-to-volume registration applications for three separate registration steps desired in enhancing neurosurgical navigation is considered. Prototype implementations for MRI-to-MRI registration, MRI-to-US registration and US-to-US registration have been made using registration methods available in the Insight Toolkit, and variants of the Mutual Information similarity metric.

The applications have been tested on clinical data from relevant surgical operations. One of the MRI-to-US registration methods, using a Mutual Information similarity metric, has been very unstable, and has given quite unreliable results. The other MRI-to-US registration method, using Normalized Mutual Information, has proved somewhat more reliable, but with significantly longer running times.

It has been experienced that gradient descent optimizers are inherently difficult to parameterize in order to obtain a stable and reliable optimization. The obtained current results indicate that a rough initialization of the registration transform should be applied, to enable the use of stricter bounds on the optimizers, further alleviating reliability issues.

It has been experienced that to be able to assess the quality of most registration results, visualization of volume slices in a 3D scene is needed. Registration of images of angiographic modalities can be readily evaluated using volume renderings, and results show the obtained results to represent sufficient accuracy to be used in clinical applications, when the stability issues are resolved.

The obtained results indicate that automatic volume-to-volume registration using Normalized Mutual Information should be feasible for the neuronavigational applications considered here, with sufficient accuracy. Further development include both evaluation of different optimization schemes, similarity metrics and preprocessing filters, and development of the necessary tools for performing accurate validation of registration results.

Preface

This report is submitted in partial fulfilment of the requirements for the degree *Sivilingeniør i datateknikk (Master of Science in Computer Science)* at the Norwegian University of Science and Technology (NTNU), Trondheim, Norway.

The overall goal of the project is to propose suitable methods for, and implement a set of fully automated volume-to-volume registration algorithms for use in neurosurgical guidance. The registration methods should enable registration of: different preoperative magnetic resonance imaging data; of an initial ultrasound volume to the preoperative data, and of each of the subsequent ultrasound volumes to the previous. In this thesis I will focus on implementation and evaluation of different methods for reaching this goal.

The work of this thesis has been realized at SINTEF Health Research, department of Medical Technology, under the supervision of Frank Lindseth. I would like to thank him for valuable comments and the necessary pushing, and Professor Richard Blake at the Department of Computer and Information Science (IDI) at NTNU for putting me in contact with the MedTech group at Sintef. I would also like to thank the Sintef MedTech researchers, especially Ole Vegar Solberg, Arild Wolff and Geir Arne Tangen for help and support with CustusX and programming issues. In addition, I would like to thank my parents for seeing things from the outside and sharing their opinions, giving me encouragement and support.

Contents

I	Introduction	1
1	Background and problem description	3
2	Summary of survey	5
2.1	The need for registration	5
2.2	Three-part registration system	6
2.2.1	Considering the three parts	6
2.3	Building blocks of a registration system	10
2.3.1	Transforms	10
2.3.2	Similarity measures — bases and metrics	11
2.3.3	Optimization	11
2.3.4	Interpolation	12
2.4	Experience drawn from related studies	12
2.4.1	Preoperative 3D MRI registration	13
2.4.2	Preoperative 3D MRI to intraoperative 3D US registration	13
2.4.3	Intraoperative 3D US Registration	14
3	Roadmap for implementing volume-to-volume registration	17
II	Detailed discussion of methods	19
4	Phase I — MRI-to-MRI registration	21
4.1	Transformation	21
4.2	Similarity measure	21
4.3	Optimizer	21
4.4	Other issues	22
5	Phase II — MRI-to-3DUS registration	25
5.1	Transformation	25
5.2	Similarity measure	26
5.3	Optimizer	27
5.4	Other issues	27
6	Phase III — 3DUS-to-3DUS registration	29
6.1	Transformation	29
6.2	Similarity measure	30
6.3	Optimizer	31
III	Implementation	33
7	Frameworks and libraries	35
7.1	Insight Toolkit — ITK	35
7.2	Using ITK	35
7.2.1	MRI-to-MRI methods	36
7.2.2	MRI-to-US methods — NMI implementation	37
7.2.3	MRI-to-US methods — MI implementation	39
7.2.4	US-to-US methods	40
8	Platforms and hardware	43
8.1	Mac OS X	43

8.2	GNU/Linux	43
8.3	Platforms and software availability	43
9	Extensions to libraries	45
IV	Results	47
10	Presentation of data sets	49
10.1	Volumes	49
10.2	Tumor1	49
10.3	Tumor2	59
10.4	Aneurism1	66
11	Tests and methods	71
12	Phase 1 registration results	73
12.1	Tumor 1	73
12.1.1	Registration # 1	73
12.1.2	Registration # 2	77
12.2	Tumor 2	80
12.2.1	Registration # 5	80
12.3	Aneurism 1	83
12.3.1	Registration # 7	83
13	MRI-to-3DUS registration results	87
13.1	Tumor 1	87
13.1.1	MR-US Tissue	87
13.1.2	MR-US Angio	96
13.2	Aneurism 1	99
14	3DUS-to-3DUS registration results	103
V	Interpretation and discussion of results	105
15	Stability and reliability discussed	107
15.1	MRI-MRI registration	107
15.2	MRI-US registration — NMI implementation	108
15.3	MRI-US registration — MI implementation	109
15.3.1	Consistency, reliability and determinism	109
15.3.2	Tissue and angio images	113
15.3.3	Varieties of the program	113
15.3.4	Other observations	113
16	Performance discussed	115
16.1	MRI-MRI registration	115
16.2	MRI-US registration	115
16.3	Parallelism	116
17	How to measure success	117
17.1	Use of “gold standard”	117
17.2	Experts’ statements	117
17.3	Other measures of success	119

VI Conclusion	121
18 Future work	123
18.1 Third phase application	123
18.2 Optimizers and second phase applications	123
18.3 Similarity measures and second phase applications	123
18.4 Tools for visualization and validation	124
19 Conclusion	125
VII Appendix	131
A Characteristics of the evolutionary (1+1) optimizer	131
B File names and descriptive names	131
C On calculating Mutual Information — the Viola & Wells way	135
C.1 How are sample points calculated?	135
C.2 How do the samples relate to MI?	136
C.3 Summary	138
D Questionnaire for evaluation of registration results	141
E Notes	147

List of Figures

1	Schematic description of registration steps	7
2	Corresponding slices from two MRI volumes	8
3	Corresponding slices from US and MRI	8
4	Sample 3D US and MR angio images	9
5	MR angio multi-resolution image set	23
6	Slices from gradient magnitude filtered US and MRI	26
7	Example B-spline curve and basis functions	30
8	Dataset Tumor 1 — MRI	50
9	Dataset Tumor 1 — fMRI and MRI T2 in 3D	51
10	Dataset Tumor 1 — MRA	51
11	Dataset Tumor 1 — MRI T2 and MRA in 3D	52
12	Dataset Tumor 1 — fMRI and MRA in 3D	53
13	Dataset Tumor 1 — US before resection	54
14	Dataset Tumor 1 — fMRI and UST#1 in 3D	55
15	Dataset Tumor 1 — MRA and USA#1 in 3D	56
16	Dataset Tumor 1 — US after resection	57
17	Dataset Tumor 1 — USA#1 and #2 in 3D	58
18	Dataset Tumor 2 — MRI	59
19	Dataset Tumor 2 — fMRI and T2 in 3D	60
20	Dataset Tumor 2 — US before resection	60
21	Dataset Tumor 2 — US after resection	61
22	Dataset Tumor 2 — T2 and UST #1in 3D	62
23	Dataset Tumor 2 — UST #1 and USA #1in 3D	63
24	Dataset Tumor 2 — UST #1 and UST #2in 3D	64
25	Dataset Tumor 2 — USA #1 and USA #2in 3D	65
26	Dataset Aneurism 1 — MRI & US	67
27	Dataset Aneurism 1 — MRI & MRA in 3D	68
28	Dataset Aneurism 1 — MRA and USA#1 in 3D	69
29	Dataset Aneurism 1 — USA#1 and USA#2 in 3D	70
30	Tumor 1 — fMRI+T1 & T2 before registration	74
31	Registration of MRI-T2 and fMRI+MRI-T1	75
32	Results Tumor 1 —fMRI+T1 & T2 registered	76
33	Tumor 1 —fMRI+T1 & MRA before registration	77
34	Results Tumor 1 —fMRI+T1 & MRA registered	78
35	Tumor 2 — fMRI+T1 & T2 before registration	80
36	Results Tumor 2 — fMRI+T1 & T2 registered	81
37	Aneurism 1 — T1 & MRA before registration	84
38	Results Aneurism 1 — T1 & MRA registered	85
39	Results Aneurism 1 — T1 & MRA registered	86
40	Tumor 1 — MRI T2 before registration	88
41	Tumor 1 — UST before registration	89
42	Tumor 1 — MRI T2 & UST before registration	90
43	Results Tumor 1 — MRA & USA registered	91
44	Tumor 1 — MRI T2 before registration	93
45	Tumor 1 — MRI T2 & UST before registration	94
46	Results Tumor 1 — MRI T2 & UST registered	95
47	Tumor 1 — MRA & USA before registration	96
48	Tumor 1 — MRA & USA before registration, region of interest	97
49	Results Tumor 1 — MRA & USA registered	98
50	Aneurism 1 — MRA & USA before registration	100
51	Results Aneurism 1 — MRA & USA registered	101
52	Results Aneurism 1 — MRA & USA registered	102

53	Questionnaire used in evaluation of registration results — Page 1. . . .	142
54	Questionnaire used in evaluation of registration results — Page 2. . . .	143
55	Questionnaire used in evaluation of registration results — Page 3. . . .	144
56	Questionnaire used in evaluation of registration results — Page 4. . . .	145

List of Tables

1	Registration operations — summarized	71
2	Results Tumor 1 — Parameters used	73
3	Results Tumor 1 — Resulting transform	74
4	Results Tumor 1 — Resulting transform for op. 2	78
5	Results Tumor 1 — Parameters used for op. 2	79
6	Results Tumor 2 — Parameters used for registration	81
7	Results Tumor 2 — Resulting transform	82
8	Results Aneurism 1 — Parameters used	83
9	Results Aneurism 1 — Resulting transform	83
10	Results MR T2-UST — Tumor 1 — Parameters used — NMI	92
11	Results MR T2-UST — Tumor 1 — Resulting transform — NMI	92
12	Results MR T2-UST — Tumor 1 — Parameters used — MI	94
13	Results MR T2-UST — Tumor 1 — Resulting transform — MI	95
14	Results MRA-USA — Tumor 1 — Parameters used — MI	97
15	Results MRA-USA — Tumor 1 — Resulting transform — MI	98
16	Results MR-US — Aneurism 1 — Parameters used	99
17	Results MR-US — Aneurism 1 — Resulting transform	99
18	RegApp2b — Reliability and consistency of results	110
19	RegApp2b — Parameters used for reliability test	111
20	Registrations included in experts' evaluation	118
21	Result of experts' evaluation	118
22	US mastered Tumor1 file names	132
23	MRI mastered Tumor1 file names	132
24	MRI mastered Tumor2 file names	133
25	US mastered Aneurism1 file names	133

Part I

Introduction

The work of this thesis builds upon my previous in-depth study in the course TDT4725, where I performed a survey of the current state of the art in volume-to-volume registration. Special emphasis was laid on concepts that are useful in multimodal registration of medical images used in neurosurgery.

In the first part of this report, after describing the goal of the overall project, I will summarize the main findings from the survey. The summary includes the concepts and methods that were considered suitable for further investigation. This will lead up to an outline of the work that is planned for this thesis.

1 Background and problem description

Before a typical neurosurgical operation, different modes of MRI data are acquired, containing potentially both structural and functional information. These images have to be combined mentally by the neurosurgeon to find an optimal plan for the operation. The most significant MRI data set is selected as a master, and used as a basis for surgical guidance.

During surgery, before the *dura mater* is opened, a free-hand US scan of the brain is performed, and the scan is reconstructed into a 3D image data set. This procedure is repeated after opening the dura, and may be performed several times during surgery, generating a series of 3D image data sets. These data sets contain updated information on the structures of the brain, in the areas reached by the US scan, which may help the neurosurgeon in monitoring and optimizing the planned procedure.

The overall goal of the project is to propose, and implement, a set of fully automated volume-to-volume registration algorithms that will perform three different registration steps: Registering the different preoperative MRI data to a selected master MRI volume, registering the preoperative master MRI to the initial US volume, and registering each of the subsequent US volumes to the previous. The last step, or the last two, should enable a set of subsequent warpings to the MR master image.

A further elaboration on the need for registration, the threefold division of the registration procedure and an insight into what has been done previously in this field is given in chapter 2. Furthermore, the most promising methods for handling the three steps are introduced in part II.

2 Summary of survey

This section will *not* serve as a complete survey of the current state of medical image registration, but merely as a short introduction to the key issues in this field, and an overview of the methods believed to be suitable for performing the three registration steps introduced in chapter 1. For a more thorough analysis and background on these topics, please see my in-depth study report (available from SINTEF MedTech), or the literature listed in the *References* section.

2.1 The need for registration

The use of medical images both for planning and guiding surgical procedures, has seen a steady growth over the last decades. New methods have been proposed, and new areas of application are included. Especially in neurosurgery is the need for accurate information, both spatially and temporally, quite pressing. Methods to enhance the information presented to the surgeon are of significant importance, since it may give the neurosurgeon the accuracy needed to perform previously inhibiting procedures [33].

In image guided surgery it is common to use a high-quality imaging modality for pre-operative planning, such as X-ray computed tomography (CT) or magnetic resonance imaging (MRI). The same images may also be used during surgery to guide the surgeon, e.g. by displaying the current position and direction of the surgical tools overlaid on the pre-operative images [3].

The pre-operative images have a very high resolution, and should thus be well suited for guidance during surgery. However, during neurosurgical procedures, the brain will normally move (shift) in relation to the skull, and the internal structures in the brain may deform. Both effects are mainly due to cerebrospinal fluid drainage, the use of diuretics, the resection of tumors, or a combination of causes [24] [26]. The pre-operative images will thus be inaccurate, and the need for more updated imagery arises.

The most common method for obtaining updated images is by basing the navigation on an intraoperative imaging modality, i.e. perform image acquisition, either continuously or at specific moments, during surgery [3] [15]. Several different imaging modalities have been proposed and integrated with neuronavigation systems, including CT[21], MRI[22] and ultrasound (US) [9] [18]. While CT and MRI will give high-quality, updated images, both these modalities suffer from drawbacks when used intraoperatively. Use of intraoperative CT imaging will expose the patient and clinical personnel to higher doses of radiation, which may have serious long-term effects. The use of open MRI is more promising for intraoperative imagery, giving good quality, updated images in reasonable real-time, but suffering from the need for extensive investments, high costs of operation and specially equipped operating rooms [9] [17].

Neurosurgical guidance based on intraoperative US imaging has been demonstrated, both by using the ultrasound data directly [9] and indirectly [1] for the guidance. When the US images are used directly, they will give the most up-to-date information available to the neurosurgeon, but this method has historically suffered from quality deficiencies, with the SonoWand® system as a notable exception [9] [13]. Still, this method effectively disregards the high quality preoperative data, and the surgeon is not able to combine the entire range of information available.

Indirect use of intraoperative US images has been reported, but not widely discussed. The proposed method of operation includes using a high-quality preoperative image modality, such as CT or MRI, and use of the comparatively low-quality US modality during surgery. From this, it is possible to determine the quantity and direction of displacement needed for registration of the preoperative with the intraoperative images. The transformation that results from registering the two data sets is used to translate, and possibly deform, the preoperative image data so that they in effect incorporate both their own initial anatomical information and the updated structural data from the US images. In addition, the functional information which is only available from pre-operative fMRI can with this also be brought into, and correctly placed, in the navigation system. The idea of indirectly using intraoperative US imagery in this manner, and in combination with its direct use, is the driving force behind both the undertaken survey and this thesis.

2.2 Three-part registration system

In my survey, I defined three separate steps that would enable the successful integration of images from different intraoperative and preoperative modalities in a single planning and navigation environment. These steps are:

1. Registration of the preoperative MRI volumes to a master MRI volume, for surgical planning.
2. Registration of preoperative *master* MRI volume with initial US volume for surgical guidance.
3. Registration of each subsequent intraoperative US volume to previous US volume, applying the detected change to the master MRI.

The second and third step will include the use of the determined transformation parameters to spatially alter the MRI data, and thus implicitly represent both the information in the US and MRI data. This combined image is dubbed a “pseudo-MRI” image, as it retains the perception of being of an MRI modality, while actually incorporating information also from the 3D US image. The repeated third steps should enable successive transformations of the pseudo-MRI to account for the changes through the operation.

A schematic representation of the three steps of the registration suite can be seen in figure 1.

2.2.1 Considering the three parts

The three registration steps are closely related to each other, but can in many ways be treated separately. Registration of the various preoperative images is a less novel problem to solve, and it should be readily addressed using well-known, and possibly standardized, methods. The different MRI weightings such as T_1 , T_2 and *MR Angio* have notable differences, but they are mostly of similar resolution and visual quality. A suitable choice of frameworks, libraries and manner of implementation should include ready-to-use methods to perform at least this registration step. An illustration of corresponding slices from two different MRI volumes can be seen in figure 2.

Being probably the most difficult step of the three, the registration of MRI and US volumes face several difficulties not present in the first step. First, the two types of images are generally much more different than is the case with various MRI weightings.

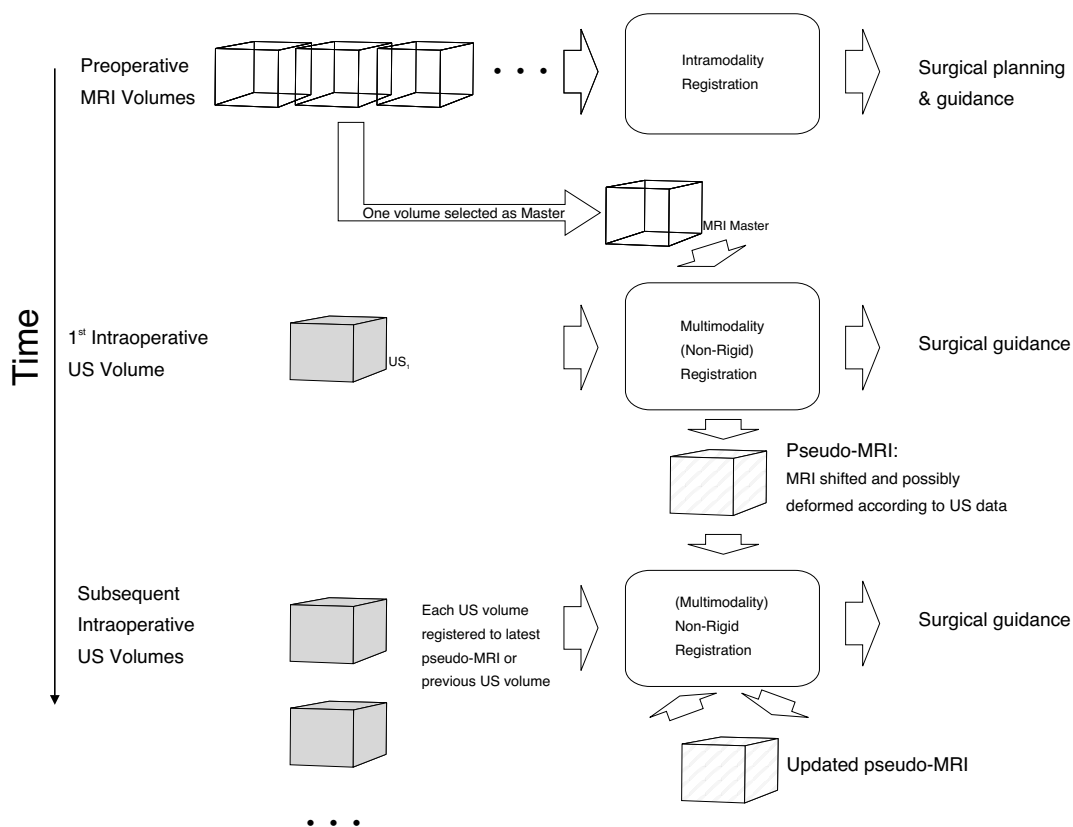


Figure 1: Schematic description of registration steps: On the left, the different image data sets acquired are represented. In the middle are the registration procedures, along with their results, while at the right the application of the particular step is included.

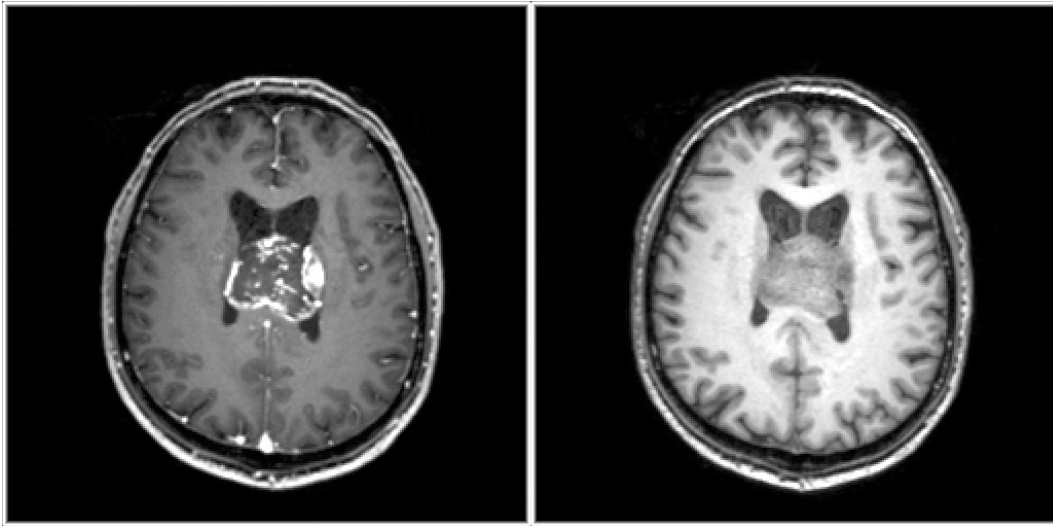


Figure 2: Corresponding slices from two MRI volumes of the same patient. Left, a T1 weighted MRI with contrast fluid used to highlight the tumor, and right, a T1 image acquired without the use of contrast fluid.

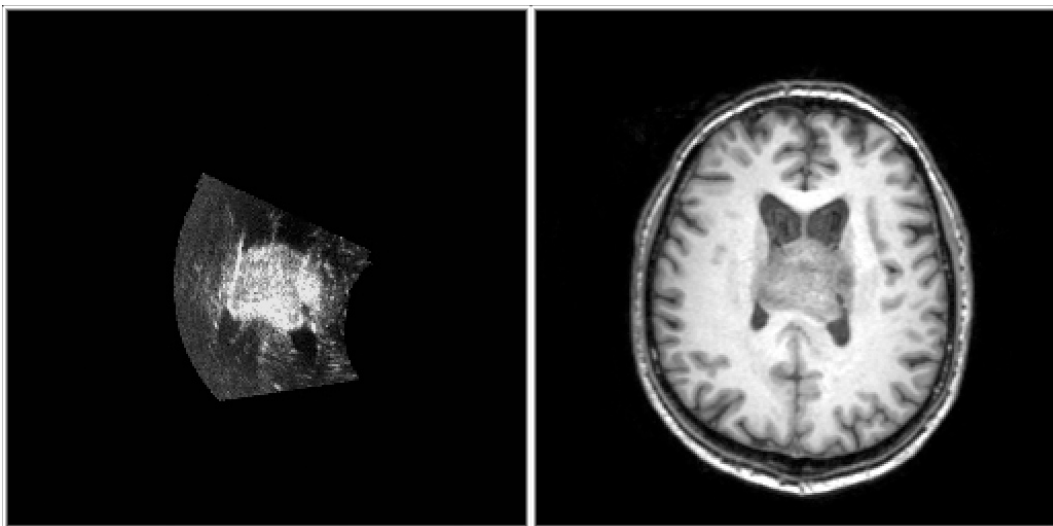


Figure 3: Corresponding slices from US and MRI volumes of the same patient. Left is a slice from the resampled volume of the compounded freehand ultrasound scan. Right, the same T1 weighted MRI as in figure 2.

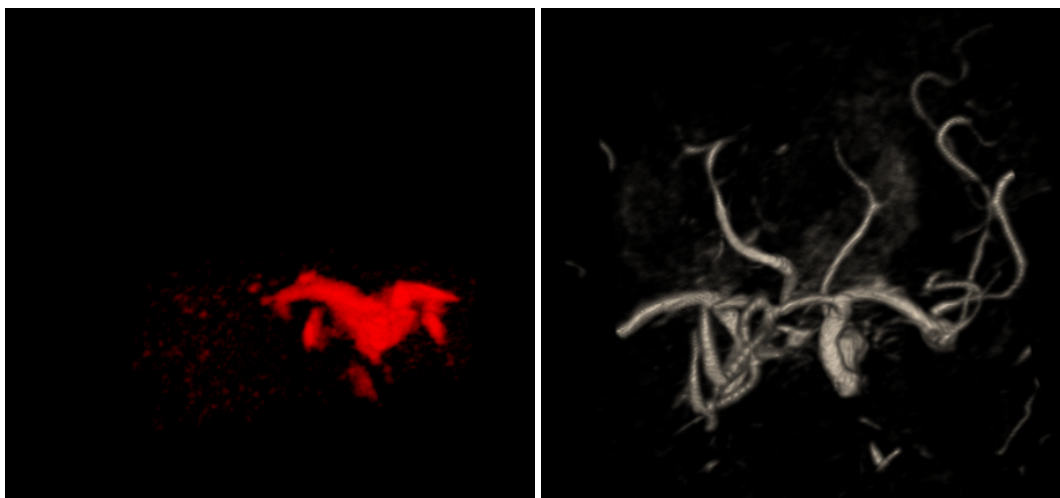


Figure 4: Sample volume renderings of 3D images from *US* *angio*(red) and *MR* *angio* (grey) modalities.

Ultrasound has lower spatial resolution, lower signal-to-noise ratio, and a drastically reduced field-of-view. Additionally the information content is primarily representing tissue *boundaries*, which contrasts with the MRI images that mainly contain information about tissue types. In other words, where an MRI will represent two different tissue types with widely different voxel values, the US may have a high response at the borderline between the two tissue types, while the bulk of *both* regions will have similar voxel intensities, of predominantly low value. This is illustrated in figure 3, where corresponding slices from a preoperative MRI volume and the first intraoperatively ultrasound volume are seen. Here the MRI volume has been brought into general alignment with the patient using artificial fiducial markers on the patient's head and a operating room positioning system, such as the optical *Polaris* from NDI, Canada. Thus, the MRI is also coarsely aligned with the ultrasound, since the latter is acquired relative to the same positioning system. However, brain shift, movement of the fiducials on the skin and calibration errors, among other sources, lead the images to not be perfectly aligned, along either of the three primary axes. Further, in figure 3, the reduced field-of-view, increased amount of noise and general border-enhancing characteristic of the ultrasound, compared to the MRI, is clearly visible. However, it is still apparent that the two modalities share some information, especially regarding the tumor (the quasi-circular shape in the middle) and the ventricles (the two oblonged, slightly curved dark "holes" on either side of the middle of the brain).

In the case of *angio*-graphic images from the two modalities, these differences are less pronounced, as the main information content regards the walls of the blood vessels, which appear in a similar manner in both modalities. In figure 4, thresholded volume renderings of a 3D ultrasound *angio* image (in red) and the corresponding MR *angio* image are presented. The limited field of view of the ultrasound image is still apparent, and both the limited resolution and the lower signal-to-noise ratio of the US is notable. This figure also illustrates how the blood vessels seem "thicker" in ultrasound than in MRI, which is also a characteristic due to the image acquisition process.

These characteristics of the images that should be registered by the second step of the procedure makes this task quite challenging. In my survey, I thus focused mainly on the progress of multimodal registration, and how it would apply to MRI-to-US

registration. A summary of my findings is included in chapter 2.4.

After the second step, we have the master MRI aligned with the first ultrasound volume, and *possibly* a pseudo-MRI constructed from perturbing the MRI according to the transformation matrix from non-rigid registration with the US data. Further US acquisitions then need to be registered to the common information now available. This can be done, either by registering each subsequent US volume to the most current pseudo-MRI, in a fashion parallel to the second step, or by registering just each US volume to the previous one. This second method is beneficial in that the pseudo-MRI may be successively updated with the transformation fields from each US-to-US registration, but will most likely need the implementation of a separate registration algorithm specifically for this step. However, unimodal registration of 3D US images, even when they are resampled freehand scans, is considered to be more likely to succeed and will probably be more robust against noise than successive use of the 2nd step algorithm on the USs and pseudo-MRIs.

2.3 Building blocks of a registration system

The goal of any registration algorithm is to discover how the two input images are related to one another. The relation is some kind of transformation that, when applied to the coordinates of one image, known as the *moving image*, it is “transformed”, as closely as possible, into the other image, called the *fixed image*. How to search for, and find, this transformation, and the very nature of the transformation itself, is what characterizes each registration algorithm. Here, I will give a brief introduction to the four most crucial design choices that will have to be made: Type of *transform*; *similarity basis* and *metric*; *optimization* scheme; and *interpolation* method.

2.3.1 Transforms

Image registration can be seen as the search for the transformation matrix \mathbf{T} . The nature of this matrix itself will therefore have a significant impact on the result, and thus on how we should be searching for it. What type of transformation matrix we want will depend on the application, especially on the complexity of the images we want to register or how different they are, and on how accurate we want the registration to be.

Two broad categories of transforms exist: Rigid and Non-Rigid. Rigid transforms will not consider deformations or scalings of the images, and the resulting transformation matrix will only consist of translations and (possibly) rotations. Rigid transforms are also inherently global, in that the same transformation is applied to the entire moving image. Non-rigid transforms may have an arbitrary number of degrees-of-freedom, which corresponds to the number of elements in the resulting transformation matrix, or deformation field. A higher degree-of-freedom means smaller and more local deformations may be detected, but each degree-of-freedom adds another parameter to be estimated, making the task of determining the correct transformation matrix more demanding and more susceptible to noise. To lessen the noise impact on the deformation field, and ensure a smooth-changing transformation, dampening schemes are usually used in non-rigid registration. A common way to smooth the deformation field is to describe it in terms of parametric cubic curves, such as B-splines or NURBS, or more precisely, their control points. Other useful non-rigid transforms include the use of mechanically or physically based models for the stretching and deformation of the deformation field.

To solve a registration problem, both the global and local transformation models may be beneficial, under different circumstances or for different purposes. Affine and especially rigid transforms will more easily find a good general alignment of two data volumes without exhaustive use of computing power. With the volumes globally aligned, a non-affine registration may be performed to assess the local deformations that have occurred between the images.

2.3.2 Similarity measures — bases and metrics

Finding the right transformation parameters is usually not a trivial task. Searching for the transformation matrix that best represents the physical relation between the moving and the fixed image, it is necessary to consider how to measure the degree of similarity between the fixed image and the moving image transformed according to the currently estimated transformation matrix. The similarity measure is combined with an optimization scheme, see below, to gradually change the estimated transformation parameters towards their optimum. A similarity measure may consist of two parts; the basis defines which information to use from each image, and the metric *how* this information is used in calculating a value of likeness. A basic distinction can be made between methods that use the image voxel values directly in the metric's computation, and those that extract some distinctive information, or features, from the images before calculating the similarity.

The idea behind the feature-based similarity bases is to form a feature space that is hopefully easier to match, using a suitable metric, than the unprocessed images. This feature space is often of dimensionality similar to the original images, which is the case when edge, ridge or other relatively low-level operators are used on the images to form the bases. Other types of feature-based similarity measure may include higher-level operations on the images to extract more abstract features, using segmented images, and possibly statistical or geometrical descriptors of the regions, but these are seldom considered in non-rigid registration of medical images, apart from in angiographic images.

Several different similarity metrics exist, and each may be apt for different purposes. Some common methods, that have proved themselves suitable for a wide range of different applications, include Mutual Information and Normalized Mutual Information, as well as Correlation Ratio. These have also successfully been used in feature-based similarity measures, with the input images enhanced by the use of edge or ridge operators, for instance.

2.3.3 Optimization

All registration methods that do not rely on merely calculating the global transformation from a set of homologous points, will need an optimizing procedure. The similarity measures compute a cost, as a function of the images to be registered and the transformation estimate, indicating how good the current transformation estimate is. At each iteration, the registration algorithm has to make a guess on how to alter the parameters, in order to obtain a better estimate. To perform this conjecture, an optimization scheme suited to the cost function is used [11] [4]. Starting with an initial estimate, supplied interactively by the user or obtained automatically, the optimization algorithm makes subsequent, hopefully better, estimates of the transformation parameters, until the similarity measure converges at an optimal (maximal or minimal) value. Specific algorithms are presented for each of the three registration steps

in chapters 4, 5 and 6, respectively.

2.3.4 Interpolation

For each set of estimated transformation parameters, the similarity metric has to be computed, and to do this, one of the images has to be transformed accordingly. This process implies some sort of interpolation, as the transformation rarely will map the voxel locations of one grid directly onto that of the other. The choice of interpolation scheme can affect the performance of the optimizer, by introducing artifacts in the parameter space [11]; when rigid transformations are considered, it is common that whole-voxel translations have a markedly higher (resp. lower) response from the similarity metric, than nearby off-grid translations. To a certain degree, this problem can be overcome using a multi-resolution approach, by performing the registration on a hierarchy of images with different resolutions, using the solution from a lower resolution level as an initial estimate for registration at the next higher level [30]. This hierarchical, or multi-resolution, approach is currently considered standard in medical image registration [11] [15]. Choosing the best interpolation method remains, however, a balancing act between computational cost and correctness.

Applicable interpolation methods include the fast, but inaccurate nearest neighbor (NN) interpolation, the ordinary linear interpolation, which is usually both accurate and fast enough, and the most accurate, but much slower cubic interpolation. The first of these methods simply uses the intensity value from the voxel that is closest to the new position. The second uses the nearest eight voxels, linearly interpolating between the nearest two voxels in all directions, which gives less aliasing and smoother transitions than using NN interpolation. Tricubic interpolation uses as much as the 64 closest voxels, each contributing to the intensity value in the new point, controlled by a cubic polynomial function of distance.

Using a cubic interpolation scheme can in some cases give a registration method the necessary information to perform accurate sub-voxel size transformations. However, as the interpolation is one of the most frequently used components in a registration application, with one execution per voxel per iteration, this will also increase the running time dramatically. In most cases, it suffices to use a trilinear interpolation in conjunction with the registration algorithm, and perhaps replace it with a tricubic interpolator for the final, real transformation of the moving image.

2.4 Experience drawn from related studies

In the first attempts of using US indirectly during surgery, Bucholz et al. [1] performed only a 2D registration between each US image and a corresponding, recreated slice from the pre-operative images (CT or MRI). The most promising results, however, for indirect use of intraoperative US imagery have been reported by the research group of Prof. T.M. Peters of Univ. of Western Ontario, Canada. They have integrated intraoperative 3D US in a neurosurgical guidance application, although only with a manual registration process [6] [7] [8] [25]. Their application includes real-time compounding and updating of a 3D US volume, which is subsequently registered with preoperative MRI, aided by the manual selection of a number of homologous points in the two volumes.

The use of intraoperative 3D US indirectly for guidance with automatic registration with preoperative images has previously not been reported, and would possibly im-

prove the quality and performance of the neuronavigation system, by enabling the surgeon to see the pre-operative image data of high spatial resolution corrected and possibly perturbed by the 3D US. It may also be possible for the neurosurgeon to choose between this mode of operation and one combining this with direct visualization of the 3D US data. This will effectively enhance the apparent value of the preoperative data in a temporal sense, while also enhancing the spatial value of the intraoperative data.

The following will summarize some of the experiences that may be drawn from reported previous attempts at solving the different registration problems. Each of the three registration phases will be handled independently.

2.4.1 Preoperative 3D MRI registration

Registration of the various preoperative MR images with a selected *master* MRI has previously been successfully performed, and standard methods will probably be adequate. Based on previous reports, using the voxel intensity values directly, together with a modern similarity measure such as correlation ratio (CR) [26] or normalized mutual information (NMI) [31], would be a feasible approach for this step. A multi-resolution registration procedure should probably be used, as this will make the method more robust, and even potentially faster.

As all the MRI data sets will be acquired prior to surgery, it will probably suffice to consider only rigid transforms. Rotational offset is in most cases negligible, so limiting the transform further, to merely the 3 translational parameters is advisable without loss of flexibility. Scaling of the images from voxel-space to physical space can be automatically performed, based on the available meta-information from the acquisition device, which includes physical voxel dimensions. Most all applications and frameworks designed to work with medical images will support this. The MRI volumes will probably be almost aligned, meaning that a manual initialization of the registration would normally not be necessary, as the images are likely to be within the capture range of the similarity metric.

2.4.2 Preoperative 3D MRI to intraoperative 3D US registration

The ultimate goal of this registration step is to align the two volumes, and potentially also detect any local deformations that has occurred between the time of MRI acquisition and surgery. In order to do this, it is necessary to consider a non-rigid transformation; or, rather, both a rigid and a non-rigid one. After first finding the global, rigid transformation that best aligns the two volumes, a non-rigid registration can be applied to compute the deformation field. These two parts of this step are related, but do not have to use the same algorithms or methods. The non-rigid step is also much more susceptible to noise and the difference between the MRI and the US may impose an inhibition on the usefulness of any non-rigid method.

The *rigid* part of the multimodal step has previously been performed by other research groups, using different approaches. Roche & Pennec et al. have employed a bivariate version of the correlation ratio that uses both the gradient and intensity information from MRI, and which has been successful in registering various MR and US images with a registration error of at most 1 millimeter [28] [27]. Lloret & López et al. developed a multilocal ridge- and valley-operator that has been employed to both MRI and US images of the brain, enabling rigid registration of these images using

simple cross-correlation as the similarity metric, although no accuracy or robustness tests have been reported [19]. In a recent PhD-thesis, Letteboer describes an approach for rigid volume-to-volume registration of preoperative MRI with intraoperative US from neurosurgery, by maximizing the normalized mutual information after finding the gradient magnitude of the two image volumes [15], with good results.

These three previous reports of successful rigid multimodal MR-US registration all mention the need for a non-rigid step to complete the registration. Roche & Pennec et al. assume that the local difference between the MR image and the first intraoperatively acquired US image during neurosurgery is sufficiently small to be ignored, and focus rather on tracking local deformations between the subsequent US volumes [24]. Any deformation between the time of MRI acquisition and operation will thus remain unaccounted for throughout the procedure, which can potentially outweigh the deformations found in later US images, during surgery. Thus, their omission of the non-rigid multimodal step should be reconsidered. Lloret & López et al. proposes to use their algorithm also for non-rigid multimodal registration, but they have not pursued this [19]. Letteboer concludes that *“a non-rigid registration algorithm needs to be developed”*, and suggests that the result of the rigid registration method could be used as an initial estimate for the non-rigid step [15]. No matter which method is chosen for the rigid and non-rigid parts of the MR-US registration, they will both most certainly benefit from incorporating a multi-resolution approach.

2.4.3 Intraoperative 3D US Registration

Both Letteboer [16] [15] and Roche & Pennec et al. [23] [24] propose to track local deformations during surgery using a non-rigid registration of the subsequent ultrasound volumes, after registering the preoperative MRI with the first intraoperative US volume.

Non-rigid registration of US volumes is, however, not trivial. Their low signal-to-noise ratio remains a problem, as the noise will differ between the volumes. However, the sound pulse scattering caused by small inhomogeneities in the tissues, which is displayed as *speckle* in the US images, is considered to be quite steady over time. Hence, a similarity measure with a predisposition to match high-valued voxels was proposed by Pennec et al. [23], and after considering various elaborate metrics they conclude that *“we preferred to keep the SSD¹ criterion”*, due to accuracy and computational aspects. Their approach further includes using a free-form deformation field smoothed by the use of a Gaussian weighting at each position, and a Levenberg-Marquardt optimization method, which includes both first and second order gradients of the similarity measure.

Another method for non-rigid registration of a sequence of 3D US images is given by Letteboer et al. [16]. Two distinct approaches to the task are discussed in Letteboer’s PhD thesis [15]; a customized optical flow algorithm, and finding B-spline-smoothened free-form deformation (FFD) field using any of the standard similarity measures such as cross-correlation (CC), SSD or normalized mutual information (NMI). Good results seemed to be possible with either method, but NMI and B-splines FFD was preferred, due to robustness and speed. Using a deformation field governed by B-splines will smoothen the transformation to avoid scattered unnatural warpings, and additionally limits the degrees of freedom, and thereby reducing the computational cost. It will, however, also limit the possible resolution of the transformation field. Increasing the density of the B-spline control point lattice successively,

¹SSD: Sum of Squared Differences

is a way of performing a multi-resolution registration, and at the same time making it possible to use a denser control point lattice, at the highest resolution, without much loss of performance. A recent development in computer graphics, dubbed *T-splines*, with an adaptive control point density [29] should also be evaluated for its potential in enhancing the performance of spline-based free-form deformation registration.

3 Roadmap for implementing volume-to-volume registration

For each of the three steps, the most promising methods from the literature, as previously outlined, should be considered and prototype implementations be made, using standardized libraries wherever available. This is the topic of parts II and III. Each prototype should be evaluated for accuracy, robustness, speed and ease of implementation. This might include testing different optimization schemes, interpolation techniques and transformation description. Speed can be readily measured, while ease of implementation and modularity will have to be subjectively evaluated. The results are presented in part IV, and further discussed in part V.

Measuring robustness and accuracy relies on the ability to decide whether a registration is successful, which may be difficult. Rigid registration methods are often compared to a *gold standard*, which is based on manual selection of a number of homologous points in both data sets. In non-rigid registration it is infeasible due to the number of necessary points. Since SINTEF MedTech currently does not possess the means to perform manual registration of 3D volumes, such as that needed here, other evaluation methods must be found for both cases. This is covered in chapter 17, part V.

Any promising prototypes may then be elaborated further to incorporate necessary error handling and flexibility, in addition to a suitable graphical user interface. *CustusX*², can possibly be used as a suitable basis, as integration of the registration software with CustusX would give it a solid platform for full operational use, while also enhancing the value of CustusX as a navigation system. Integration of the registration methods with CustusX is not covered in this thesis, but some remarks on this topic can be found in the discussion in part V.

²CustusX is a clinically tested interventional navigation system developed by SINTEF Health Research, which supports advanced visualisation features such as stereoscopic vision and facilitates different volume data rendering modes.

Part II

Detailed discussion of methods

In this part, each step of the registration system, and the methods that will be used in their implementation, is discussed. Characteristics of the data at each step are taken into account, together with previous reports of successful registration applications and their methods. Initial, grounded assumptions regarding the suitability of various methods, and likewise their shortcomings, are also presented.

4 Phase I — MRI-to-MRI registration

Registration of the different preoperative MRI data sets should be feasible using the voxel intensity values directly, together with a modern similarity measure such as correlation ratio[26] or (normalized) mutual information[31]. A multi-resolution registration procedure should probably be used, as this will make the method more robust, and potentially faster. As all the MRI data sets will be acquired prior to the surgery, with little deformation, it will probably suffice to consider only rigid transforms.

4.1 Transformation

Choosing a suitable type of transformation is crucial for the construction of a sound registration method. For this first registration step, where only MR images of different weightings are considered, there will be negligible deformations, and we only need to consider rigid transformations. This will also limit the number of degrees of freedom of the system drastically, implying a reasonably fast and accurate registration. There are two distinct types of rigid transforms that may be used — with or without rotations. Ordinarily, all the MRI sets are acquired while the patient remains lying within the MRI scanner, and in the same coordinate system, which will give image sets with little rotational differences. Based on these characteristics of the input data to this registration step, we will assume that only translational transformations need to be considered. This is the easiest form of registration, with merely 3 degrees-of-freedom.

4.2 Similarity measure

Given the fact that the input images are likely to be of comparable spatial resolution and extension, as well as in content, it should not be necessary to use very elaborate similarity measures. Ordinary maximization of mutual information (MI) should be sufficient for this task. As we assume the inputs to be different MR images, the images' intensity values should be a good enough basis. The only preprocessing needed to compute the MI efficiently and reliably is to normalize both input images (i.e. re-map the intensities so that each image will have $\mu = 0, \sigma = 1.0$), and ensure that they share a common, reasonable resolution.

4.3 Optimizer

Searching for a simple translational transform with a mutual information similarity metric can be performed using a choice of different optimizers. An optimizer that considers the gradient of the metric at each point should not be too demanding for this application, since the dimensionality of the search space is rather limited. The exact choice of algorithm is dependant on the availability of methods in the libraries used.

With any such optimizer a number of parameters must be set, indicating acceptable limits on the searching. This will likely include maximum and/or minimum step length, and the maximum number of iterations before ending the registration. The registration will be ended when either of two criteria are fulfilled. Either if the new transformation values are closer to the current values than the minimum step

length, the program will terminate. Also if the maximum number of iterations has been reached, the registration will end.

4.4 Other issues

As with all registration methods, quite a few parameters have to be set for the method to perform optimally. To compute the MI metric, a number of random sample points throughout the two images are used — how many are needed will depend on the size of the images and the wanted accuracy of the registration, but too many will slow down the procedure heavily. Too few will give a poor registration, or no result at all.

The registration method may, as described in chapter 2.3, also be performed in a multi-resolution fashion; where a stack of images with varying resolution are used successively to find the optimal transform. The transformation coefficients that result from the registration procedure on one level is used as an initial estimate at the next, finer level. Building up a suitable stack, or pyramid of images, such as the one in figure 5, can be somewhat timeconsuming when a high resolution 3D image is used, but this may be regained by the increased speed of the entire registration process. The images at the different resolution levels are constructed by smoothing and interpolating the image repeatedly, and it is often a good idea to change the optimizer's parameters at each level to ensure that the step lengths used are sound, considering the size of the features one wants to align at each level.

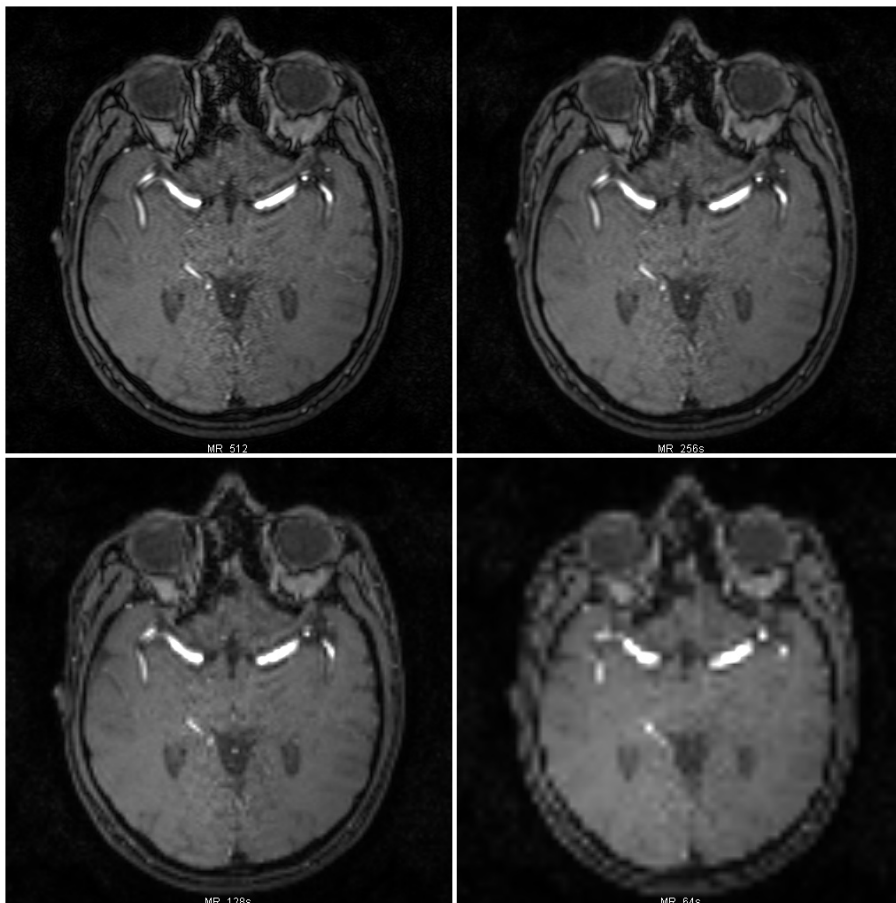


Figure 5: Sample set of corresponding image slices from a set of successively downsampled MRI angiograms. The original (512x512 px.) at top left. Approx. 1:2 scale

5 Phase II — MRI-to-3DUS registration

The true multimodal registration step will be the registration of one of the preoperative MRI volumes, with an intraoperative US volume. This will also be the most challenging part. In order to detect any local deformations that has occurred between the time of MRI acquisition and surgery, it is necessary to consider both a rigid and a non-rigid transformation. The *rigid* part of the multimodal step has previously been reported, using different approaches that should be tested:

- Roche & Pennec et al. have employed a bivariate correlation ratio, using gradient and intensity information from MRI, which has been successful in registering various MR and US images with a registration error of at most 1 millimeter[28].
- Lloret & López et al. developed a multilocal ridge- and valley-operator, enabling rigid registration of MRI and US using simple cross-correlation [19].
- In a recent PhD-thesis, Letteboer described using maximization of the normalized mutual information after finding the gradient magnitude of the two image volumes [15], with good results.

Neither of these reports, nor others, include automatic non-rigid registration of 3D MR and US images. Further investigation and testing of the three methods has to be performed to assess whether either of them would be useful for the non-rigid registration.

5.1 Transformation

Considering first the rigid registration of MRI and 3D US, the transformation we seek in this part is naturally a rigid one. Rotations should most probably be taken into account since the coordinate systems and orientation of the images are presumably not as similar as in the MRI-to-MRI-registration in chapter 4. A 6 degrees-of-freedom transform is thus sought for the rigid part.

Any extension to perform a non-rigid registration of the master MRI and the first US volume, should use the result of the rigid registration as a starting point for its search for an optimal deformation field. The deformation field itself should probably be defined in terms of cubic splines, or rather their control points. The exact choice of spline-type to be used will primarily depend on the availability of corresponding representation in the employed image processing library. An important aspect of the choice of spline-type is the varying ease of subdivision, i.e. how computationally expensive it is to re-compute a spline control point lattice at a different resolution. Some of the most appealing parametric curve types in this respect are Bèzier curves, nonuniform B-splines (including Non-Uniform Rational B-Splines — NURBS) [5], and a rather new generalization of non-uniform B-splines used to define surfaces, called T-splines [29]. Using T-splines, control points can be added and removed where they are needed, while keeping the number of control points low where there is little detail to describe. If it is possible to extend the T-spline approach from surfaces to an entire deformation field, this may enhance the usefulness of spline-based, non-rigid transformations. This would enable the transform to describe a smoothly varying 3-dimensional deformation field, without an inhibiting number of control points to be determined, and with an ease of subdivision unlike traditional spline-based deformation fields.

However, available libraries supporting non-rigid registration procedures are not likely to have implemented a form of T-splines, while deformation fields based on Bèzier curves or some type of B-splines are most likely available. The desirable ease of subdivision of the T-splines will have to be weighed against the costs of implementing such a method in full, contrasted with the ease of use of any readily available methods.

5.2 Similarity measure

Mainly, the three similarity measures used in the previous attempts at registration of 3D US and MRI can be regarded as the best current practice in this matter. However, third-party comparisons of the methods has not been reported, and it is therefore difficult to say which is most promising. All three use a differential similarity basis, i.e. use a gradient or ridge/valley operator to extract image information prior to registration. The methods vary in the specific operator used, and the different metrics applied for similarity measurement.

The reason for using a differential similarity basis, such as a gradient magnitude filter, when one wishes to compare an MRI and a 3D US image, lies in the nature of the modalities. As mentioned in chapter 2.2.1, the ultrasound gives high response (image voxel intensity) in areas of rapid change in the acoustical impedance, such as at tissue boundaries. High voxel intensities in an MRI, on the other hand, represent areas of high proton density and high freedom of rotation of the molecules containing hydrogen [14]. As an effect, the intensities in an MRI volume are more or less uniform in areas of uniform tissue, while the same areas in the US volume have mostly very low values with scattered noise — speckle. However, using an edge-enhancing operator on the MRI volume will result in high values where the intensity changes abruptly, such as at tissue boundaries. Thus, submitting the images to an edge-enhancing operator of some sort, will lessen the difference that is due to the inherent properties of their modalities. In figure 6 we can see corresponding slices from both a US and an MRI volume after gradient magnitude filtering. The originals are those pictured in figure 3.

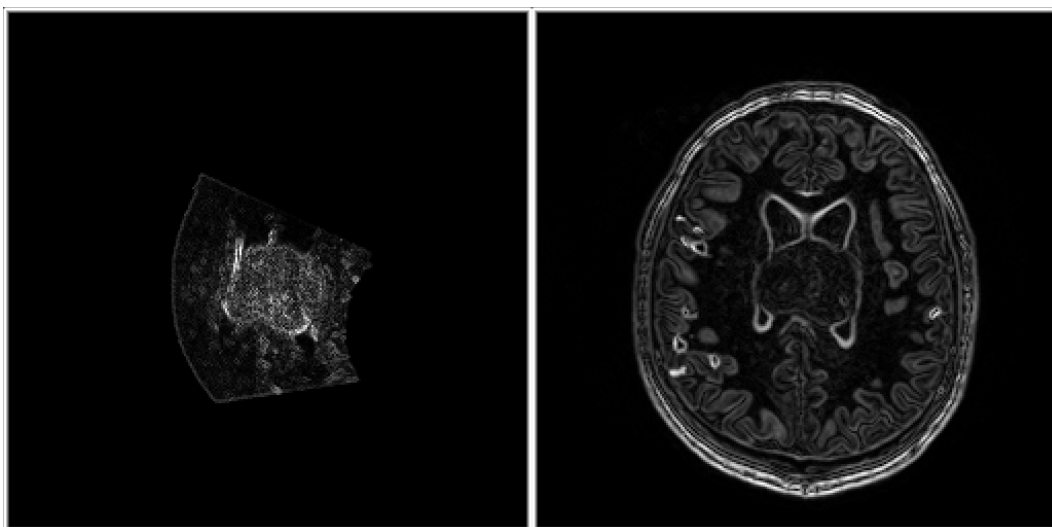


Figure 6: Corresponding slices from gradient magnitude filtered US (left) and MRI (right) volumes of the same patient.

The three approaches that should be considered, as similarity measures both for the rigid and for the potential non-rigid registration, are: Normalized Mutual Information with gradient magnitude filtering of the input images (NMI+GM), a bivariate version of the Correlation Ratio metric, using both gradient magnitude and unfiltered intensity values as data (BCR+GMI), and a simple Correlation Coefficient with multilocal creaseness (a ridge operator) filtering (CC+MLC). Of these, the first is the most likely to be implemented with existing library software, while the two others depend on highly specific methods; i.e. the bivariate Correlation Ratio metric and the multilocal creaseness operator, respectively.

5.3 Optimizer

Following the suggestions in the papers presenting the three methods, and those of the Insight Toolkit Software Guide [12] (see chapter 7.1 for more on the Insight Toolkit), an evolutionary optimizer should be used for the NMI+GM [15], while an ordinary gradient descent optimizer with a controlled step length is advisable for the CC+MLC [19]. The BCR+GMI has been successfully implemented using a Newtonian gradient descent optimization scheme. The exact choice of optimizer, and its parameters will depend on the implementation, and the specific libraries used.

Evolutionary optimizers try, as suggested by their name, to mimic the mechanisms of natural evolution, applying evolution-like rules to the iterative process of searching for the optimum. The method of operation of this optimizer differs substantially from the more common gradient based optimizers. Thus, a more thorough description of its characteristics is included in appendix A.

5.4 Other issues

As in the first step, a multi-resolution approach should be desired, both for the rigid registration, and for the possible, subsequent nonrigid registration. This will ensure that many small-scale differences are only taken into account when the images are well aligned on a larger scale. While this increases the complexity of the registration method, the ability to make significant steps towards an optimal registration at each iteration on the lower resolution levels, makes it worthwhile and possibly even time-saving in total running time.

However, the choice of optimizer may affect the usefulness of the multi-resolution procedure. This concerns perhaps primarily the evolutionary optimizer, which to a certain extent incorporate a similar smoothing effect to the parameter space as the levels of the multiscale approach, because of its multinormally distributed parameter change vector, and this multidimensional Gaussian's covariance matrix based on previous success and failure. In short, the (1+1) evolutionary optimizer may not need several resolution levels, and may even become too slow for practical purposes with many levels.

6 Phase III — 3DUS-to-3DUS registration

Tracking of local deformations during surgery should be possible using non-rigid registration of the subsequent ultrasound volumes, after registering the preoperative MRI with the first intraoperative US volume. Matching the US volumes with previous US volumes will make the registration easier, as the images are of the same modality, maintain a similar field of view and mainly picture the same anatomical structures. Proposed methods for doing this includes:

- Pennec et al. [23] suggests using Sum of Squared Differences as similarity measure to find a free-form deformation field smoothed by the use of a Gaussian weighting at each position.
- Letteboer proposes to optimize a B-spline-smoothed free-form deformation field using normalized mutual information,

Both the methods proposed by Pennec et al.[23] and by Letteboer [15] should be considered for this implementation. Using the correlation ratio similarity metric of Roche et al. [26] should also be considered.

6.1 Transformation

As described in chapter 5.1, there are several ways to describe a non-rigid transformation. Such a transformation can no longer be described as a single transformation matrix, since the transformations will differ from point to point within the volume. Thus, a deformation field (DF) is needed. In its crudest form, the DF has the same dimensions as the moving image volume, with the exception that each point in the DF is associated with a vector, describing the displacement in space of that point. In this form, there is nothing to keep the resulting, transformed moving image from wrapping around itself or warping completely out of realistic boundaries. To discourage these physically implausible deformations, and to ensure that adjacent points are not displaced completely independently, some form of dampening or regulation of the DF must be introduced. Additionally, it is a goal to keep the transformation's ability to adapt to small, local deformations that are present in the images, while reducing the number of degrees-of-freedom (DOF) of the system. With the most basic formulation of the DF, the DOF will be as high as the number of voxels in the volume. This is typically in the multi-millions, and much too high for any optimizer to practically estimate (in anything near reasonable time for the purposes discussed here).

Two methods that will reduce the DOF drastically, while maintaining the ability to describe smaller deformations, and at the same time smooth out unnatural displacements, are cubic spline DFs and Gaussian weighted, sparse DFs, as mentioned above. These techniques are indeed quite similar, and differ mostly in their formulation and theoretical background. They both use a number of grid points that are lesser than the total number of voxels, and a smoothing function defined over each point. The smoothing function distributes each point's displacement vector over the surrounding volume. Any voxel will thus be most influenced by the nearest grid point's displacement vector, and to a lesser extent by the displacement at grid points further away. To avoid having to sum up a combination of displacement vectors, factoring in all their distances, it is desirable for the DF to have a locality property — that only a limited number of grid points will affect each voxel's displacement and, correspondingly, that each grid point will only affect a limited number of voxels.

When Gaussians are used to distribute the displacement vectors over the grid, this locality property is not theoretically fulfilled, as the normal distribution does not reach absolute zero on either side of the mean. A Gaussian-smoothed displacement vector will, in theory, contribute to the entire volume's displacement. However, as the distance between the voxels and the grid point increases, the influence fades quickly (by the square of the distance), and most implementations of this method would use a cut-off radius — outside of which the grid point is assumed to have no influence.

Using cubic splines, and especially B-splines, the locality property is fulfilled also in theory. B-splines are normally used to describe curves (in the plane or in space) and surfaces, but can also be expanded further to describe a DF. For each grid point — or control point in the terminology of splines — a cubic basis function describes how the influence of that point's displacement vector is distributed over the local area. The basis function is only defined for a limited range of 0 to 1 of the relative parameter t , and for B-spline curves this ensures that each control point will only affect the four nearest segments. This is illustrated in figure 7, where a cubic b-spline curve segment is seen, together with its defining control points, and the basis functions for each of those control points. The position of the curve in space is a linear combination of the positions of the control points, weighted by the basis functions.

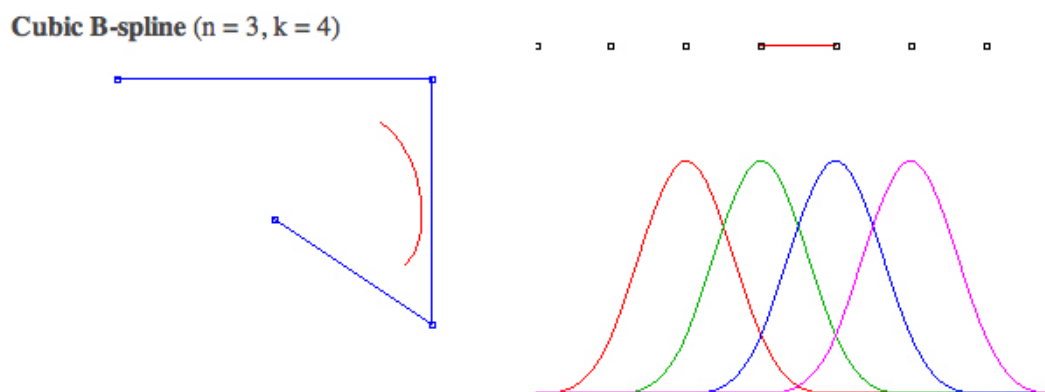


Figure 7: A cubic B-spline curve segment (red), on the right, with the four defining control points (blue), and the four basis functions corresponding to the weight of the four control points. Figure courtesy of [2].

In a similar fashion, the displacement of any position in the volume will be calculated by linearly combining the displacement vectors from each of the nearby control points, multiplying it by the corresponding basis function value for each position.

6.2 Similarity measure

As described in chapter 2.4 and above, two proposed methods for registering ultrasound volumes stand out. The first, by Pennec et al. [23], is based on the experience that while the US volumes may be of poor visual quality compared to MRI or CT, the high-intensity speckle introduced in US images by inhomogeneities in the tissue is well suited for matching two subsequent US volumes. To do this, one only needs a similarity metric that will give priority to matching voxels of high intensity, such as sum of squared differences (SSD). This is much less sophisticated method than those discussed so far, as it merely calculates the square of the intensity value differences

between the fixed and the transformed moving image, voxel by voxel, summing the result.

Another approach, as suggested by Letteboer [15], includes using a more elaborate similarity metric — normalized mutual information (NMI) — in the belief that the characteristics of this metric makes it suitable for matching most images, regardless of their special features, such as speckle. NMI would potentially also use the information content of the speckles to match the images, but would do so implicitly, while the SSD metric has this as its explicit goal.

Neither of these approaches will require any preprocessing of the images to extract features, or otherwise compose a similarity basis other than the volumes themselves.

6.3 Optimizer

The SSD-based method has been suggested to use a Levenberg-Marquardt-like optimization scheme (LMO), while the approach based on NMI has not referred to any preferred optimizer. The LMO is basically an optimization method that uses both the first-order gradient and a simplified second-order gradient to estimate the next parameters, and will vary the weight put on the two gradient types according to the progress of the optimization. When the estimated parameters give a better metric value, more weight is put on the second-order derivative to facilitate optimal small-scale matching, while as long as the new parameters give no improvement in the similarity metric, more weight is put on following the gradient, which is assumed to give better larger-scale matches.

For the NMI-based method, any of the methods used in the previous phases with this metric will probably be sufficient. This includes the (1+1)-evolutionary optimizer, and possibly the simpler gradient descent scheme.

With either of these methods, a multi-resolution approach should probably be used. When combining multi-resolution registration with a non-rigid deformation field controlled by B-splines (or similar), the grid points should be subdivided along with the resampling of the image from coarsest to finest.

Part III

Implementation

This part describes how the designs from the previous chapters is implemented. The programming frameworks and software libraries to be used are presented, along with the most important algorithms and methods they provide, and do not provide. A brief outline of the computer hardware and software platform used for the implementation is also given. Finally, the need for extensions to the chosen software libraries, to support the needed methods, is discussed.

7 Frameworks and libraries

This chapter will focus on the software libraries that are used to build the registration system incorporating the methods discussed previously. Chapter 7.1 includes a description of the *Insight Toolkit (ITK)*, a widely recognized software library in medical image processing and analysis. Comments on specifics of implementing the registration system using ITK, and especially which particular methods are available to perform the needed work, are found in chapter 7.2.

7.1 Insight Toolkit — ITK

The Insight Toolkit (ITK)³ is a software library implementing a multitude of methods for analyzing images, primarily concerning segmentation and registration of medical images. The toolkit has extensive support for the different data and file formats used in the various fields of medical imaging, as well as a very modular and consistent programming interface for including ITK functionality in medical image analysis software.

ITK is released under an open-source license, where anyone, free of charge, can download the source code and instructions for building the library on their own computer. The toolkit is implemented in standards compliant C++, using a special build environment called CMake⁴ to generate system-specific compilation procedures, making the entire suite completely cross-platform. Interfaces to other programming languages are also offered, and these currently include Java®, Python® and Tcl. Most application development using the ITK libraries, however, is performed using the same templated C++ programming language constructs as the ITK itself. This *generic programming* approach ensures that more potential errors can be discerned at compile-time, and that classes and functions of the library can be used in different settings — i.e. with different templated parameters — without any rewrite of the source code.

The ITK is under constant development, and being an open-source initiative, anyone can suggest changes or additions to the library. The development process, indicated by the core developer team to be following the *extreme programming* paradigm, is monitored and managed by the Insight Software Consortium (ISC)⁵, a non-profit organization set up for this purpose. Sponsors include the U.S. National Institutes of Health, especially the U.S. National Library of Medicine, which initiated the work, and the six principal members of the ISC: Kitware, GE Corporate R&D, and Insightful, all commercial enterprises, along with the three academic institutions UNC Chapel Hill, University of Utah, and University of Pennsylvania. People from all these institutions continue to participate in the development of ITK, increasing the speed, stability and functionality of the toolkit with each release.

7.2 Using ITK

Below follows a short summary of the methods described in chapters 4, 5 and 6, and the algorithms, classes and functions in ITK that are available to perform the various tasks.

³See: <http://www.itk.org/>

⁴See: <http://www.cmake.org>

⁵See: <http://caddlab.rad.unc.edu/ISC/>

7.2.1 MRI-to-MRI methods

The first registration phase application, for registering different MRI volumes, will be implemented using these methods:

Translation Transform

The translation transform will give the registration problem 3 degrees-of-freedom. It is provided by:

```
itk::TranslationTransform<TScalarType=double, NDimensions=3>,
templated over the data type for the transformation calculations and the number of
dimensions.
```

Mutual Information Metric

A basic formulation of the Mutual Information Metric, as devised by Viola and Wells will be used, and this is provided by:

```
itk::MutualInformationImageToImageMetric<TFixedImage, TMovingImage>,
templated over the Image types for the input images. The metric also has three im-
portant parameters that will need to be set, in order to make the registration method
perform well. They are: the number of spatial samples to be used in computing the
mutual information value, and the standard deviation of the Gaussian kernels used
in the probability distribution density estimation, for the fixed and moving images
respectively, upon which the MI value is calculated (see Appendix C for a detailed
description of how the MI metric value is actually calculated). Finding optimal val-
ues for these, and the other parameters, is a crucial part of developing a working
registration application.
```

Gradient Descent Optimizer

Using a gradient descent optimizer with a bounded step length — that is, a step length kept between a maximum and minimum value, is a good choice in combination with the Mutual Information Metric. The ITK class `RegularStepGradientDescentOptimizer` does this, by starting with a maximal step length and halving it each time a step gives a gradient of the metric value that point in the general opposite direction of the last. This is provided by:

```
itk::RegularStepGradientDescentOptimizer.
```

This optimizer also has a number of parameters, the most common and important ones include: the maximal number of iterations it is allowed to use, the maximal and minimal allowed step length per iteration, and whether it should try to minimize or maximize the metric value. When using the MI metric, the last parameter should always be set to *maximize* the metric value. The upper step length boundary should be set so that the method accomodates reasonably large translations, but not much larger, to avoid the optimizer searching too wide an area in parameter space. The lower step length limit, n , also acts as a stop criterion; when the optimizer cannot find a new set of tranformation parameters more than n millimeters in any direction, from the previous, the registration process ends. The lower limit should thus be set as low as the needed registration accuracy requires, but setting it too low will likely make the procedure run much longer (until it runs out of iterations) without achieving more than hopping between a few parameter values slightly more apart than the lower limit distance.

Other toolkit classes used

Other classes in the ITK used for the implementation of the registration method include:

```
itk::ImageRegistrationMethod<TFixedImage, TMovingImage>,
setting up the interaction between the transform, optimizer and metric, and
itk::LinearInterpolateImageFunction<TInputImage, TCoordRep=double>,
supplied to the ImageRegistrationMethod for defining how to interpolate voxel values when calculating the metric value. Additionally, both the fixed and moving images are subjected to a normalization scheme as mentioned in chapter 4, to better fit the Mutual Information characteristics. This is performed by:
itk::NormalizeImageFilter<TInputImage, TOutputImage>,
and is performed on image intensity values converted to floating point precision.
```

7.2.2 MRI-to-US methods — NMI implementation

The application that will perform the second phase registration, will be implemented by combining the specific algorithms described below. For the current implementation, only rigid transforms will be considered, as initial studies of typical data sets show little apparent deformation between the MRI volumes and the first US volume. If there, after rigid registration has been accomplished, is evidence that deformations are occurring, a non-rigid extension should be made to this program.

Additionally, an application based on (non-normalized) mutual information is also attempted, and this is covered in chapter 7.2.3.

Rigid Transform — or Translational Transform

The second phase application should be able to discover both translational and rotational differences between the MRI and US volumes, and while an affine transform could do this, only a rigid transform is needed. The translational transform from the first application must be augmented by a rotational component. In ITK this can be provided by:

```
itk::VersorRigid3DTransform<TScalarType=double>,
but this (and the other available rigid transforms) expect specialized gradient descent optimizers, which in turn are not well suited when using the Normalized Mutual Information metric. Thus, the NMI-based registration will first be implemented using only a translational transform, provided as before by:
itk::TranslationTransform<TScalarType=double, NDimensions=3>.
```

Gradient Magnitude Filter

To be able to better match the MRI and US images, features will be extracted from both, using a gradient magnitude filter. This is performed by:

```
itk::GradientMagnitudeRecursiveGaussianImageFilter<TInputImage, TOutputImage>,
which computes the gradient magnitude by convolving the image with a recursive approximation of the Gaussian. Objects of this class takes as parameter the standard deviation (in millimeters) to be used for the approximated Gaussian, which results in a varying degree of edge smearing.
```

Normalized Mutual Information Metric

Normalized Mutual Information (NMI) is believed to have some advantages over the basic Mutual Information metric, especially in their sensitivity to image overlap. On the other hand, the NMI does reportedly not work as well with traditional optimization schemes like the gradient descent method, as mentioned above. The NMI metric is in ITK provided by:

```
itk::NormalizedMutualInformationHistogramImageToImageMetric<TFixedImage, TMovingImage>
```

which computes the NMI based on histograms for each of the two input images and their joint histogram. The metric is customizable by tuning of its parameters, which include the number of bins in the histograms and scaling of the derivative of the metric in terms of the parameter components. The number of bins should be set high enough to enable the metric to implicitly distinguish between e.g. tissue types, but should be held low enough to ensure that most bins receive an ample number of samples, to ensure the statistical significance of the calculations.

Evolutionary Optimizer

The Normalized Mutual Information metric in combination with a gradient magnitude filtering of the input images would, as commented in chapter 5, benefit from an evolutionary optimizer. In ITK, this is provided by the class

```
itk::OnePlusOneEvolutionaryOptimizer.
```

As mentioned above, however, this optimizer is not likely to work very well with the `VersorRigid3DTransform`, so the application will first employ only translational transforms, for which the evolutionary optimizer will probably work fine. This optimizer, as others, take a number of parameters, including the option to minimize or maximize the metric value. As before, this is set to maximization, since this is one of the basic properties of the mutual information metric. Of other important parameters, this specific optimizer needs an initial search radius, a maximal number of iterations and a minimal value of ϵ , the Frobenius norm of the covariance matrix, as described in further detail in chapter 5.3. The initial search radius should be set high enough for the optimizer to find the path toward the wanted translational offset, and the epsilon limit set so high that it has a realistic chance of interrupting the registration process on the grounds of too little progress per iteration. This will typically put the initial search radius in the range of 1-2 millimeters, and the epsilon at around $1e - 3$ to $1e - 4$.

Other toolkit classes used

In addition to the methods and classes already mentioned, this application will use the interpolator as mentioned for the first application. Using the gradient magnitude filter, along with the NMI metric, it is not necessary to normalize the image intensity values in this application.

While the NMI combined with the (1+1)-evolutionary optimizer probably does not benefit very much from a multi-resolution approach, the application will be set up for this possibility, using:

```
itk::MultiResolutionImageRegistrationMethod<TFixedImage, TMovingImage>
```

Performing a mono-resolution registration will then still be possible, by specifying to the registration method to only use one level in the image pyramid.

7.2.3 MRI-to-US methods — MI implementation

In order to overcome the problems with the NMI metric in combination with the rigid transform, another Mutual Information based application for the second phase will be implemented. This will use these components:

Rigid Transform

The MI application will use the rigid transform discarded for use with the NMI metric;

```
itk::VersorRigid3DTransform<TScalarType=double>.
```

This has six degrees-of-freedom; three translations, and a three-component versor, or unit quaternion. The versor is the most optimal form of describing a 3D rotation, corresponding to the first three components of the quaternion. Internally it consists of the four quaternion components, but the last is computed so that the quaternion is kept at unit length.

Gradient Magnitude Filter

The gradient information shall again be used as basis for the registration, and the gradient magnitude filter is the same as in the previous application:

```
itk::GradientMagnitudeRecursiveGaussianImageFilter<TInputImage, TOutputImage>.
```

Mattes' Mutual Information Metric

A different implementation of the Mutual Information metric, due to D. Mattes [20], is available in ITK, and will be used. This calculates the MI based on estimates of the images' marginal and joint histograms (or PDFs). The histograms and the MI is computed using only a single sample point set, and this implementation is considered to be faster than the original MI metric. The method is provided by:

```
itk::MattesMutualInformationImageToImageMetric<TFixedImage, TMovingImage>.
```

The metric only has two parameters that need to be set, the number of bins to be used for the histograms, and the number of spatial sample points. Further, it is important to note that this particular class returns the negative of the actual MI value, so an optimal value is as large a negative number as possible — that is, the optimizer will need to *minimize* the metric value.

Specialized Gradient Descent Optimizer

The VersorRigid3DTransform works reportedly best with a gradient descent optimization algorithm. However, the straight-forward RegularStepGradientDescent makes the assumption that the parameters form a vector space, and while that is true for the three translational parameters, it is *not* true for the three versor components. Thus, a specialized version of the gradient descent optimizer has been implemented in ITK to ensure that the versor is updated in a sane way. This is provided by:

```
itk::VersorRigid3DTransformOptimizer.
```

The versor rigid optimizer, as its regular step gradient descent sibling, has a number of parameters. In this context, it first has to be set to minimize, rather than maximize the metric value, due to the special characteristics of the Mattes' Mutual Information metric class. Second, the maximum and minimum step lengths will have to be set,

as for the `RegularStepGradientDescent`. Further, the optimizer will be working with both translational and rotational (versor) values, which have widely differing scaling — e.g. a change of 1.0 in translation will affect the result much less than a change of 1.0 in one of the versor components. This will be compensated for, by specifying scaling parameters to the optimizer. The scales are used in gradient descent optimizers to adapt the gradient vector, which is used to calculate the new transformation parameters, by dividing each of its components with the corresponding scale component. A rule of thumb given by the ITK authors for setting the scales is to set the three rotational scales to 1.0, and the translational scales to $1/(10 * \sqrt{X^2 + Y^2 + Z^2})$, where X , Y and Z are the dimensions of the (fixed) image in physical units. For typical neurological images of 100-300 mm in each dimension, this will give translational scales at around 1/1000 to 1/5000.

Other toolkit classes used

This registration application will use the same normalization and interpolation classes as the first application. In addition, the multi-resolution approach from the NMI-based program will also be employed for this program, possibly with even more effect. The multi-resolution registration program takes the number of levels in the image pyramid as a parameter, along with specifications on how to calculate the image size at each level. The default subdivision scheme in ITK should work reasonably well, which uses the following equations to calculate the shrink factors, s_i for each level $i = 1 \dots N$, N being the number of indicated levels:

$$s_1 = 2^{(N-1)} \quad (1)$$

$$s_i = s_{i-1}/2, i = 2 \dots N \quad (2)$$

For four levels, this gives the shrink factors: 8; 4; 2; 1, indicating that the coarsest image has 1/8 the resolution of the original (which is used at the last and finest level of the pyramid).

7.2.4 US-to-US methods

These methods will be used in the third registration program, to register US volumes:

Rigid Transform, and Non-Rigid Deformation

The third registration phase will make use of both a rigid and a non-rigid transformation. The rigid part will be identical to the previous, based on the `VersorRigid3DTransform` and the Mattes' Mutual Information Metric, while the non-rigid part will be an addition to this, using a B-spline based deformable transform, provided in ITK by:

```
itk::BSplineDeformableTransform< TScalarType, NDimensions, VSplineOrder >.
```

Mutual Information Metric

The same mutual information metric used by the previous application will be used for this program, provided by:

```
itk::MattesMutualInformationImageToImageMetric<TFixedImage, TMovingImage>.
```

This will be used for both the rigid and non-rigid part of this registration phase.

Gradient Descent Optimizers

The rigid part of the registration will use the same specialized gradient descent optimizer as the previous application, while the non-rigid part will use a Limited Memory Broyden, Fletcher, Goldfarb and Shannon Optimization scheme, provided by:

```
itk::LBFGSOptimizer,
```

which is a quasi-newton method especially apt for minimization of locally smooth functions with computable 1st derivatives — as we believe the metric value is, as a function of the B-spline deformable transform parameters. It may be more successful with a larger number of parameters than the gradient descent algorithms, and in the Limited Memory version, with a reduced number of recorded updates, a reasonable memory footprint should be possible, along with an unprohibitive running time. A disadvantage of the LBFGSOptimizer is its lack of support for interactive observation of the progress of the optimization. Most optimizers in ITK support the Event framework, where for each iteration of the optimization, the Observers associated with the particular IterationEvent are invoked, and information can be printed, or adjustments to the registration process submitted. In the case of the LBFGS, only timing information — which parts of the optimization process used the most time — is available, and only after the registration has finished. However, the LBFGS is reckoned to be the most suitable optimizer for this problem.

Other toolkit classes used

The multiresolution approach used in the rigid part is identical to the one described for the phase 1 registration application, above. For the non-rigid part, the image resolution will not be changed; the resolution levels will be implemented entirely by changing the resolution of the B-spline control point grid. To go from one resolution level to the next, the actual deformation field will have to be computed from the low-resolution B-spline grid, and decomposed back into the the high-resolution B-spline grid. This will be performed by using the ITK class:

```
itk::BSplineDecompositionImageFilter,
```

among others.

8 Platforms and hardware

In this chapter, a brief description of the platforms used for developing and testing the registration software, is given.

8.1 Mac OS X

Mac OS X[®] is an operating system from Apple Computer based on *BSD*[®] *UNIX*[®], for Apple's computers which are based on the PowerPC[®] CPU architecture. Mac OS X, which is currently at version 10.4, uses the free GNU[®]⁶ Compiler Collection (*gcc*), for compilation. Since the operating system vendor is also the supplier of the computer hardware, the Mac platform has earned a reputation for highly optimized performance. This is especially true in image and video processing, and other applications where floating point operations are abundant, as this is one of the strong points of the RISC⁷ PowerPC architecture, compared to the x86 platform.

8.2 GNU/Linux

Linux[®] is a free, open-source operating system available for various platforms, but primarily intended for Intel x86 and its derivatives. Linux is closely attached to various tools from GNU, and the suite's proper name is thus *GNU/Linux*. GNU/Linux has become a popular operating system, both for personal computers and enterprise servers, as it combines the power and flexibility of a UNIX-like operating system, with the cheap and standardized hardware of the x86 PC platform.

8.3 Platforms and software availability

Both these platforms are readily supported by the Insight Toolkit, using CMake for configuration, and the vendor-supplied GNU utilities *make*, *gcc* and *c++* for compilation of C++ sources. Adhering to ISO standard⁸ C++ code in the registration applications, the programs should compile on both these platforms, and others, without any modifications. While CMake and ITK are both available also for Microsoft Windows[®], to make the registration applications compatible with this environment would potentially have sacrificed some functionality. The registration programs are therefore developed for use on any platform having ITK, CMake, a POSIX-compliant C Library (such as GNU's *glibc*) and an ISO-compliant C++ library (such as GNU's *libstdc++*, released with the *gcc*).

⁶GNU — an acronym for GNU's Not Unix

⁷RISC — Reduced Instruction Set Computer

⁸ISO Standard 14882

9 Extensions to libraries

So far, the focus of our attention has been on the registration methods currently supported by the classes in ITK. However, in order to test the other algorithms — primarily similarity metrics, feature extraction filters and/or optimizers — that are *not* available in ITK at the moment, these methods will have to be implemented. To ensure consistency, interoperability and interchangeability of the modules of the registration applications, any such implementations should be done within the same framework, which in this case is ITK, deriving the necessary classes from the most similar ones in the toolkit. The ITK is still a work in development, and new classes supporting different registration methods emerge as they are implemented and thoroughly tested — this can be a very elaborous and time-consuming process, since the accuracy and accountability requirements for equipment in the medical industry are very high.

The registration methods currently available in ITK, as discussed in the previous chapters, represent the most standard and widely used algorithms, which are accordingly also those most likely to be available in future versions of ITK, or any other medical imaging library with image registration capabilities. Building registration applications mainly with these standard building blocks will ensure that revising the applications later, or porting them to use another, competing library, is likely to be easier. In addition, investigating first how well these methods can perform, with detailed analysis of the various combinations of methods, and the many parameters each method carries, will unveil both whether the standard methods may be sufficiently accurate, and whether there is a need to test some of the more exotic methods.

Thus, since this work represent the first steps in the use of automatic volume-to-volume registration methods at SINTEF MedTech, I will primarily use the available registration methods in ITK. I will concentrate on assessing how accurate they may be, and whether they are suited for the tasks at hand. Further additions to the set of ITK registration methods, either as released by the Insight Software Consortium, or developed independently, may then be compared to these standard methods working as close to their optimum as possible.

Part IV

Results

Three of the four registration applications, implemented using the methods presented in the previous part, are used to register 3D images of several modalities for different patients. First, each data set consisting of a varying number of MRI and US volumes, is presented using illustrative 2D and 3D representations. The various volumes are then submitted to the corresponding registration application(s), with a choice of parameter settings. The registered volumes are presented in 3D renderings along with their pre-registration counterparts, and the corresponding reference volume used in the particular registration. Interactive 3D renderings will be the principal source for interpretation and discussion of the results, in the subsequent part V. Problems and difficulties regarding stability, reliability and robustness of the methods will also be covered in part V. Thus, in this part, the presented results will represent typical results that will give an indication on how well the methods may perform, with the specified parameters.

The phase 3 (US-to-US) registration application was not finished in time for typical results to be included in this part.

10 Presentation of data sets

Below follows a presentation of the data sets used to test and assess the implemented registration methods. All data sets have images that need registration for all three phases.

10.1 Volumes

The datasets are all from real surgical operations, and made available in Meta Header/Raw format, exported from the SonoWand ultrasound navigation system. The pre-operative volumes are imported into that system prior to the surgical procedures, from the DICOM⁹ series acquired by the MRI scanner(s). Subsequently, the SonoWand system has been used to reconstruct the acquired ultrasound scan into a 3D representation. The entire collection of volumes for a single surgical procedure has then been exported with a common representation, i.e. using the same file format, bit-depth, voxel spacing and physical extent for all the volumes. Here, all datasets are exported using 8 bits per voxel and a voxel size of 1 mm or less (see below).

Some of the datasets are available in two versions. One version where the MRI volumes have been used as master for 3D composing and resampling the ultrasound scans; that is, the extent of the MRI defines the size of the ultrasound volumes. In the other version, the ultrasound volumes have been used as master; and only those parts of the MRI that overlap with the ultrasound is included in the exported MRI volumes. This is remarked upon in each case, below.

The files are named using a time-stamp notation, representing when they were first imported/acquired in the navigation system, using a format like this:

{YYYY}{MM}{DD}T{HH}{MM}{SS}.mha/raw, e.g.: 20050308T080204.mha.

The volumes will in the following be referred to using a descriptive name, with tables showing the relationship between file names and descriptive names included in Appendix B.

10.2 Tumor1

The data set consists of seven (7) MRI volumes, and seven (7) US volumes of a patient with a brain tumor. The first three volumes contain fMRI¹⁰ data overlaid onto a T1-weighted MRI (included separately as the seventh volume). The responding fMRI areas are given maximum intensity (255), and the three fMRI volumes represent the following stimulations, respectively: Finger movement, language and tongue movement. Of these, only the last (fMRI Tongue) was considered relevant to the tumor resection. A slice from this volume can be seen in figure 8.

The fourth volume is a T2 weighted MRI, and a slice of this can also be seen in figure 8. A 3D rendering of the fMRI and the T2 MRI is included in figure 9. The fifth is a T1 weighted MRI acquired without the use of contrast fluids, and is only of modest interest. The sixth is an MR Angiogram, which is potentially of great interest

⁹The Digital Imaging and Communications in Medicine standard — a standardized file format for storing medical images and wide variety of meta information, including patient data, acquisition information, physical and medical properties of the images etc.

¹⁰functional MRI — an MRI modality where contrast fluid is used to determine the areas of the brain that are most active during specific stimulations

in tumor resection, and a slice is seen in figure 10. This volume is seen in 3D together with the MRI T2 in figure 11, and with the fMRI in figure 12. The seventh, and last MRI volume, is the underlay for the fMRI, i.e. a T1 weighted MRI with contrast fluid. This is mostly interesting in conjunction with the fMRI data.

The US volumes are: one US tissue (UST #1) and two US angio (USA #1 and USA #2) volumes acquired before opening the dura mater. In addition, there is one US angio (USA #3) and three US tissue (UST #2 - #4) volumes from after tumor resection. A corresponding slice, before registration, from the first three volumes can be seen in figure 13, and from the last four in figure 16.

The Tumor1 data set is available both with US- and MRI-mastered volumes. All figures mentioned above are from the MRI-mastered volumes, unless explicitly stated.

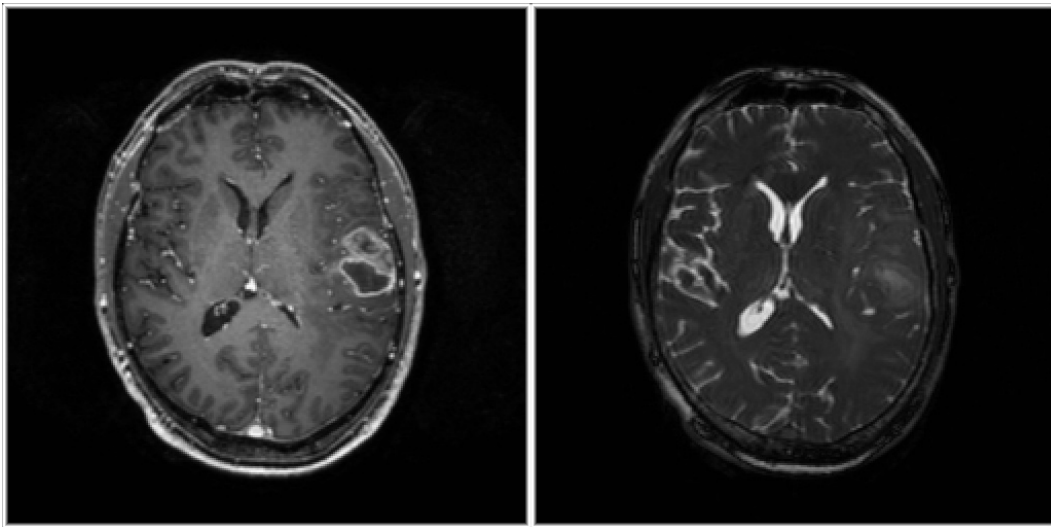


Figure 8: Dataset Tumor1. Left, fMRI with highlighted tongue response (visible in the back of the brain and along the circumference). Right, the corresponding slice from the MRI T2.

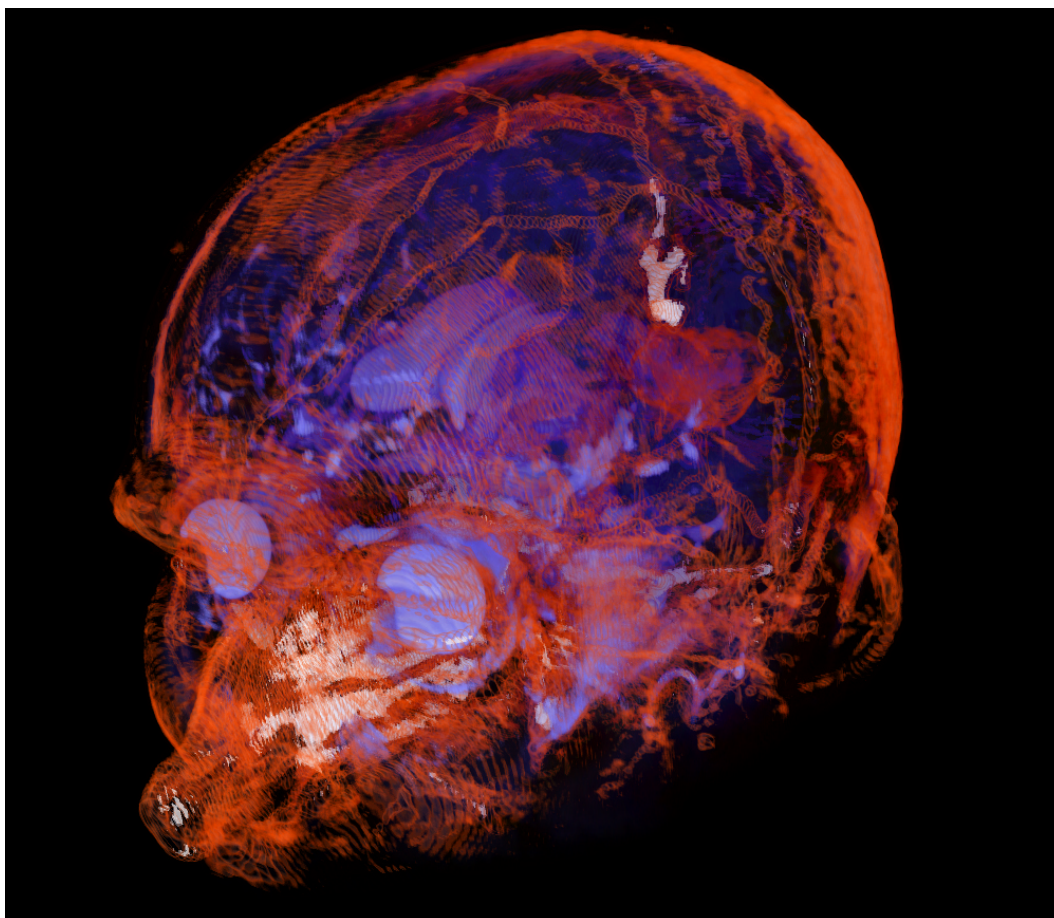


Figure 9: Dataset Tumor1. Volume renderings of fMRI (*orange* MRI T1 with *white* fMRI tongue response overlay) and MRI T2 (*blue*).

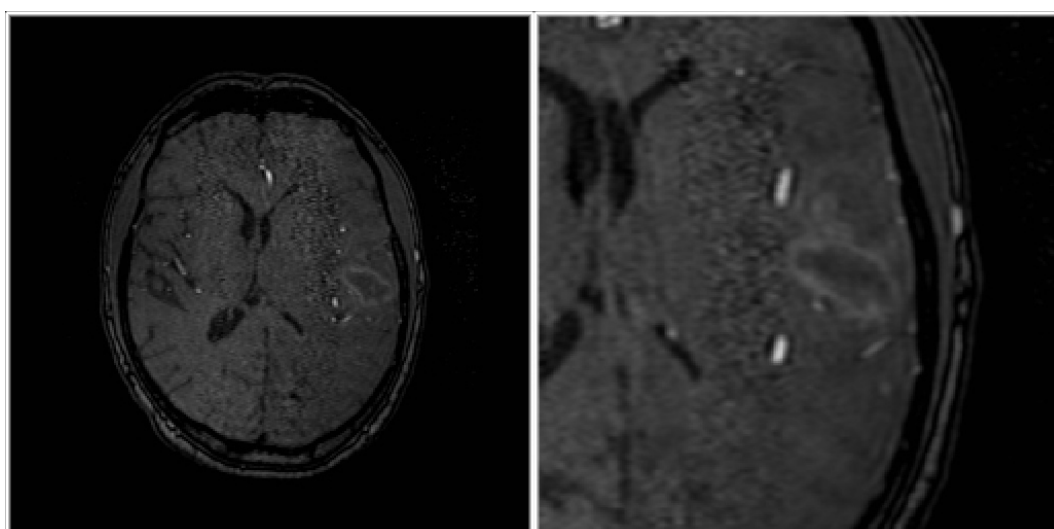


Figure 10: Dataset Tumor1. Left, slice corresponding to figure 8, from MRA. Right, same slice from US-mastered MRA.

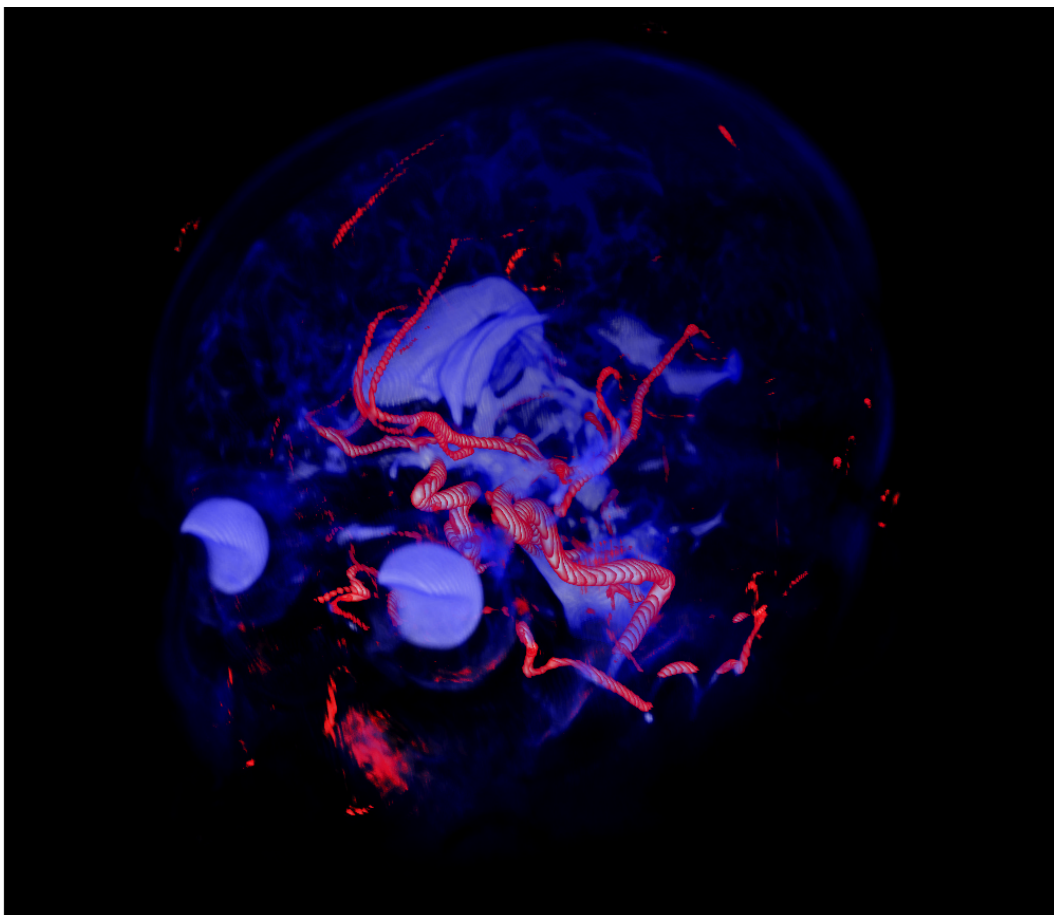


Figure 11: Dataset Tumor1. Volume renderings of MRI T2 (*blue*) and MRA (*red*).

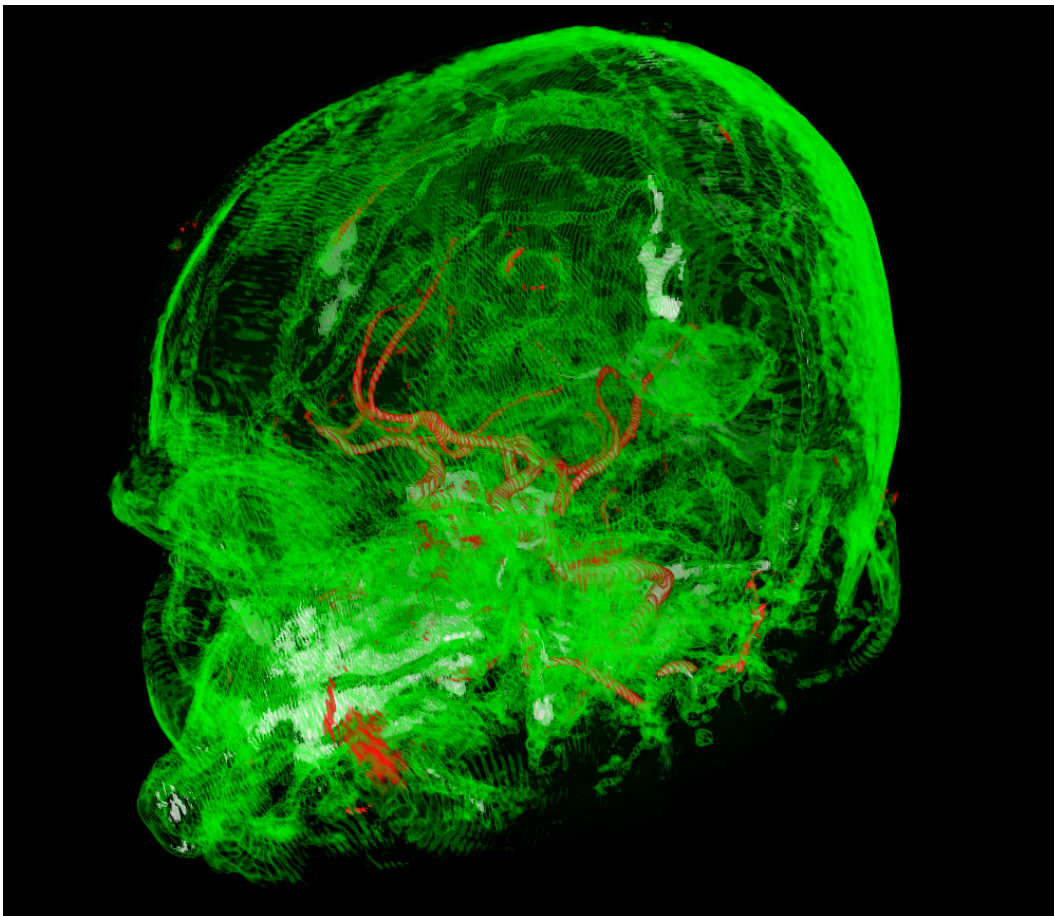


Figure 12: Dataset Tumor1. Volume renderings of fMRI (*green* MRI T1 with *white* fMRI tongue response overlay) and MRA (*red*).

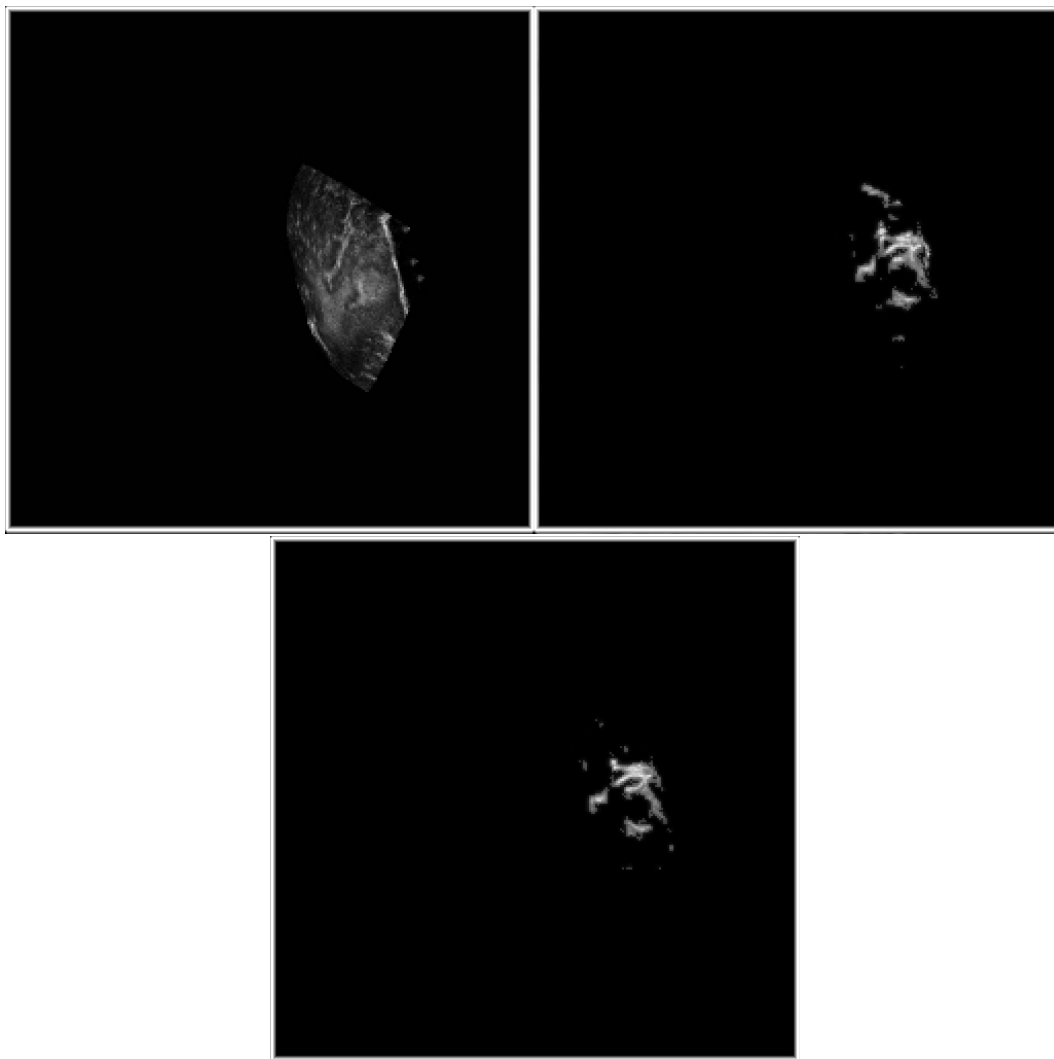


Figure 13: Dataset Tumor1. All slices corresponding to figure 8, before registration. Top left, UST #1. Top right, USA #1. Bottom, USA #2.

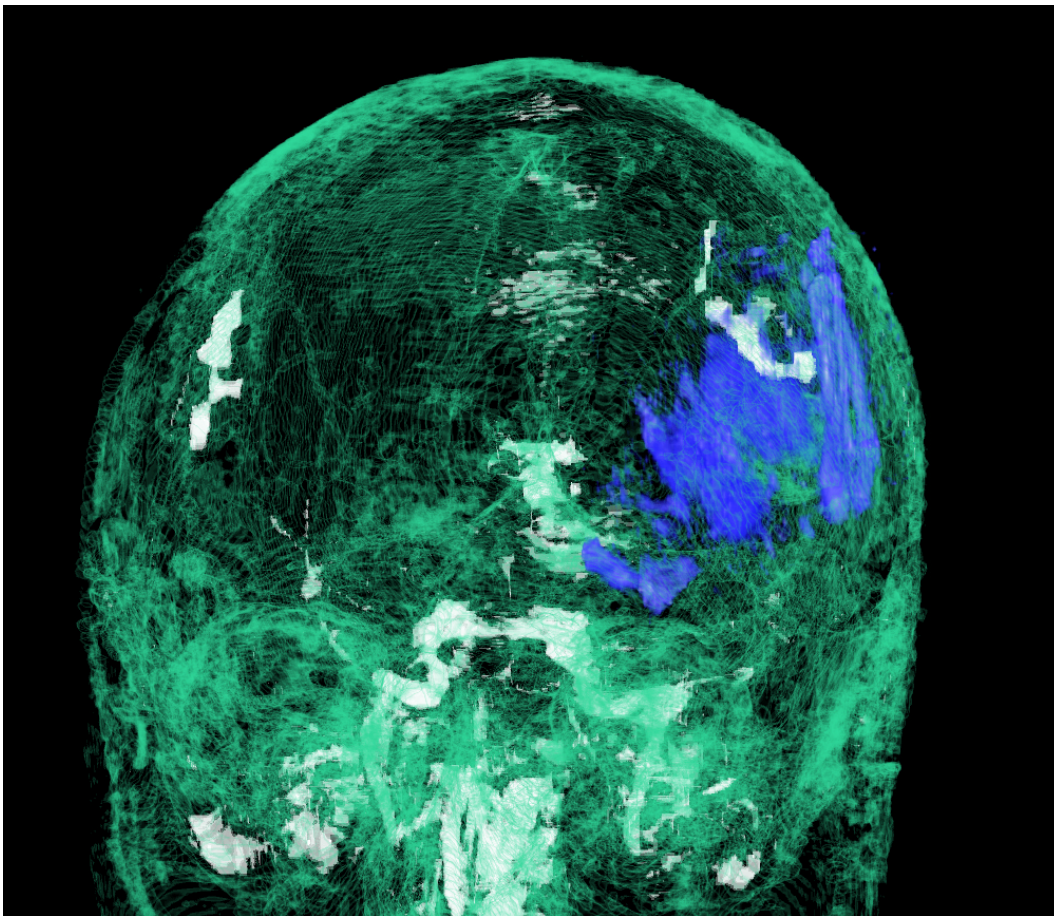


Figure 14: Dataset Tumor1. Volume renderings of fMRI (*green* MRI T1 with *white* fMRI tongue response overlay) and UST #1 (*blue*).

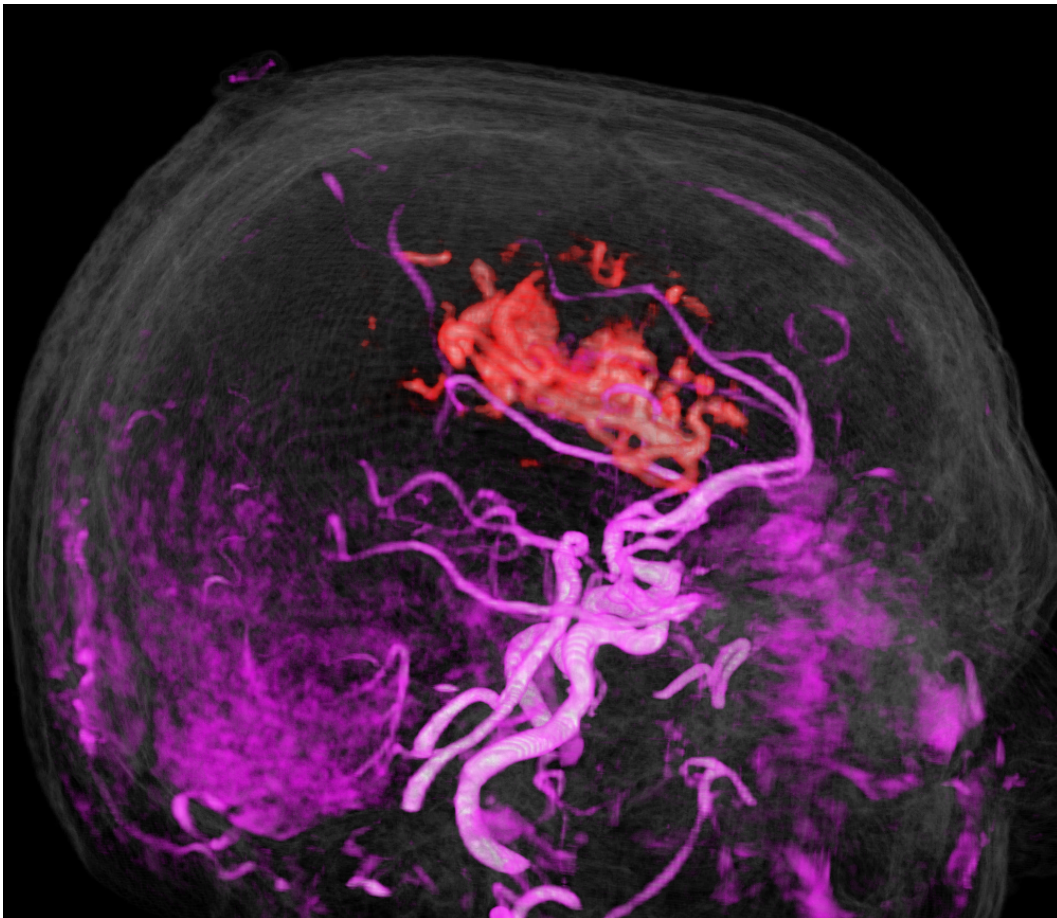


Figure 15: Dataset Tumor1. Volume renderings of MRA (*pink*) and USA #1 (*red*).

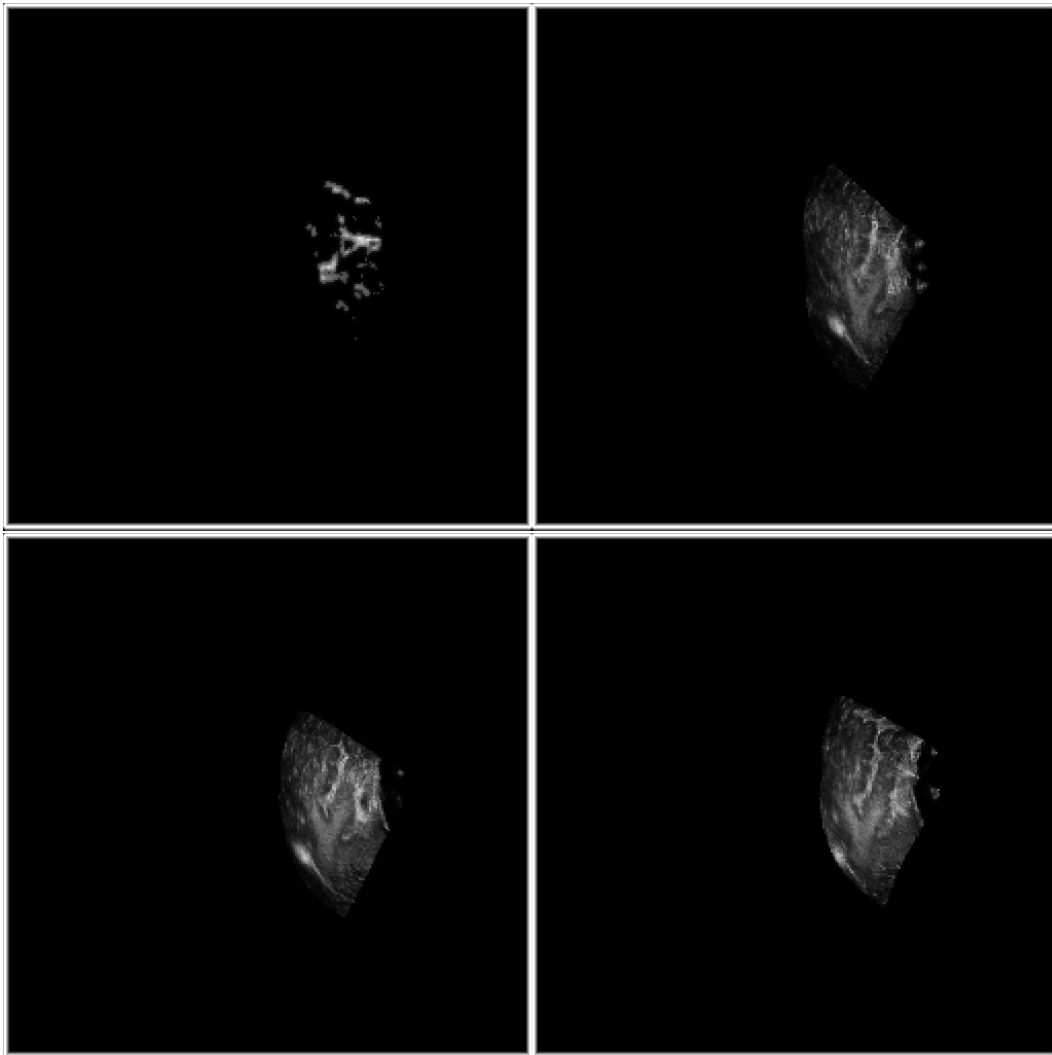


Figure 16: Dataset Tumor1. All slices corresponding to figure 8, before registration. Top left, USA #3. Top right, UST #2. Bottom left, UST #3. Bottom right, UST #4.

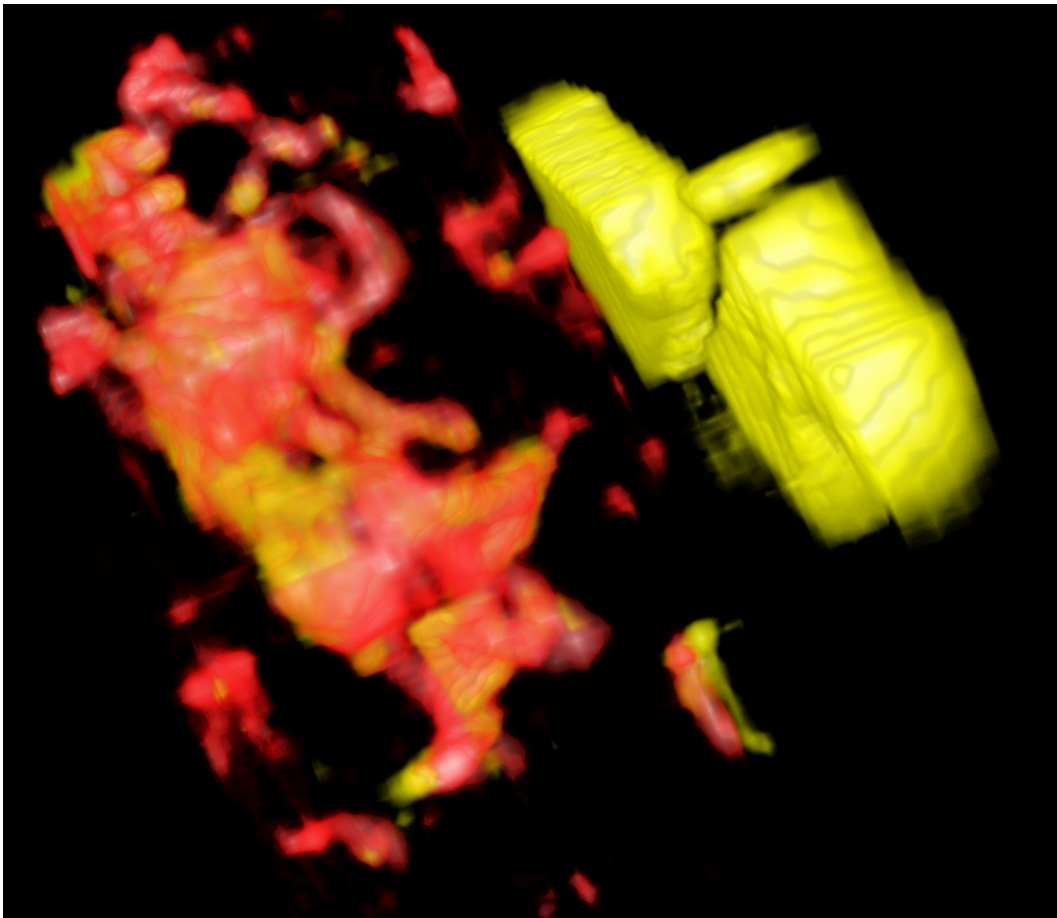


Figure 17: Dataset Tumor1. Volume renderings of USA #1 (*red*) and USA #2 (*yellow*).

10.3 Tumor2

Like data set Tumor1, this is of a patient with a brain tumor, which was due to be resected. Data set Tumor2 consists of 12 volumes, with one fMRI, two MRI T1 (with and without contrast fluid) and one MRI T2. The remaining eight volumes are five tissue and three angio ultrasound volumes. The first UST and USA are both acquired at the dura, while the others are recorded after tumor resection

The fMRI image consists, as for the previous data set, of an fMRI response (tongue stimulation) overlaid on the T1 weighted MRI with contrast fluid. Slices from the fMRI and MRI-T2 are seen in figure 18, and a 3D representation is included in figure 19. Figure 20 shows slices from the first two ultrasound volumes, and 21 the remaining six US volumes. The first UST can also be seen in a 3D scene, together with the MRI T2, in figure 22, and with the first USA in figure 23. The first ultrasounds can be compared to the second ones of each modality in figure 24 and figure 25, for UST and USA, respectively.

This data set is only available with MRI-mastered volumes.

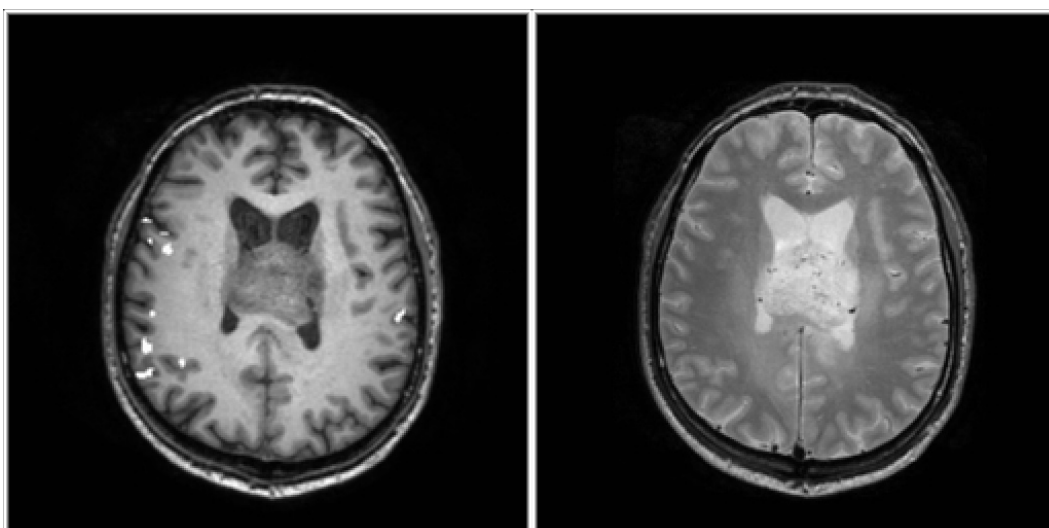


Figure 18: Dataset Tumor2. Left, fMRI with highlighted tongue response. Right, corresponding slice from MRI T2.

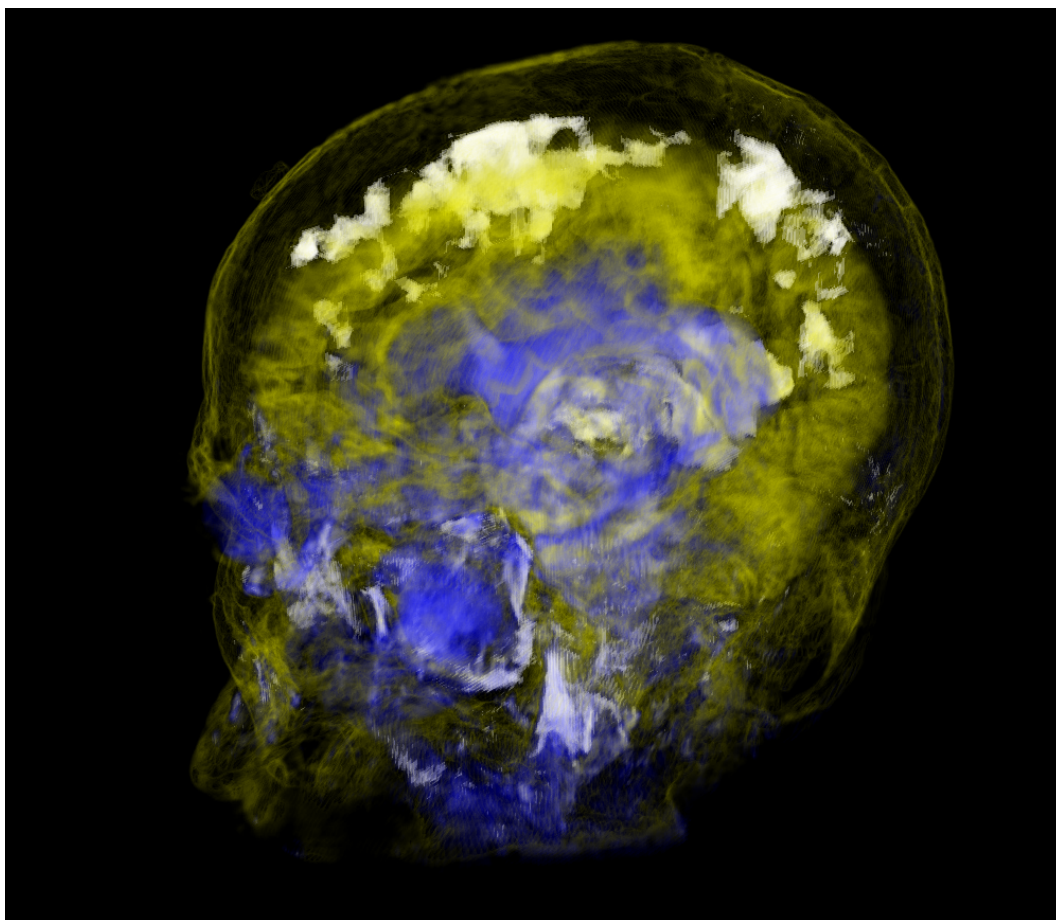


Figure 19: Dataset Tumor2. Volume renderings of fMRI (*yellow* MRI T1 with *white* fMRI tongue response overlay) and MRI T2 (*blue*).

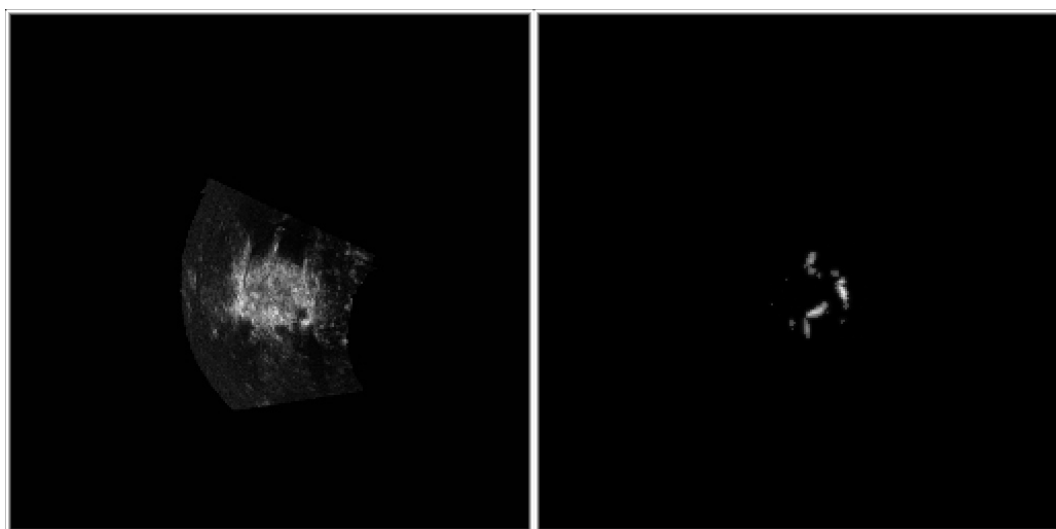


Figure 20: Dataset Tumor2. All slices corresponding to figure 18, before registration. Left, UST #1. Right, USA #1.

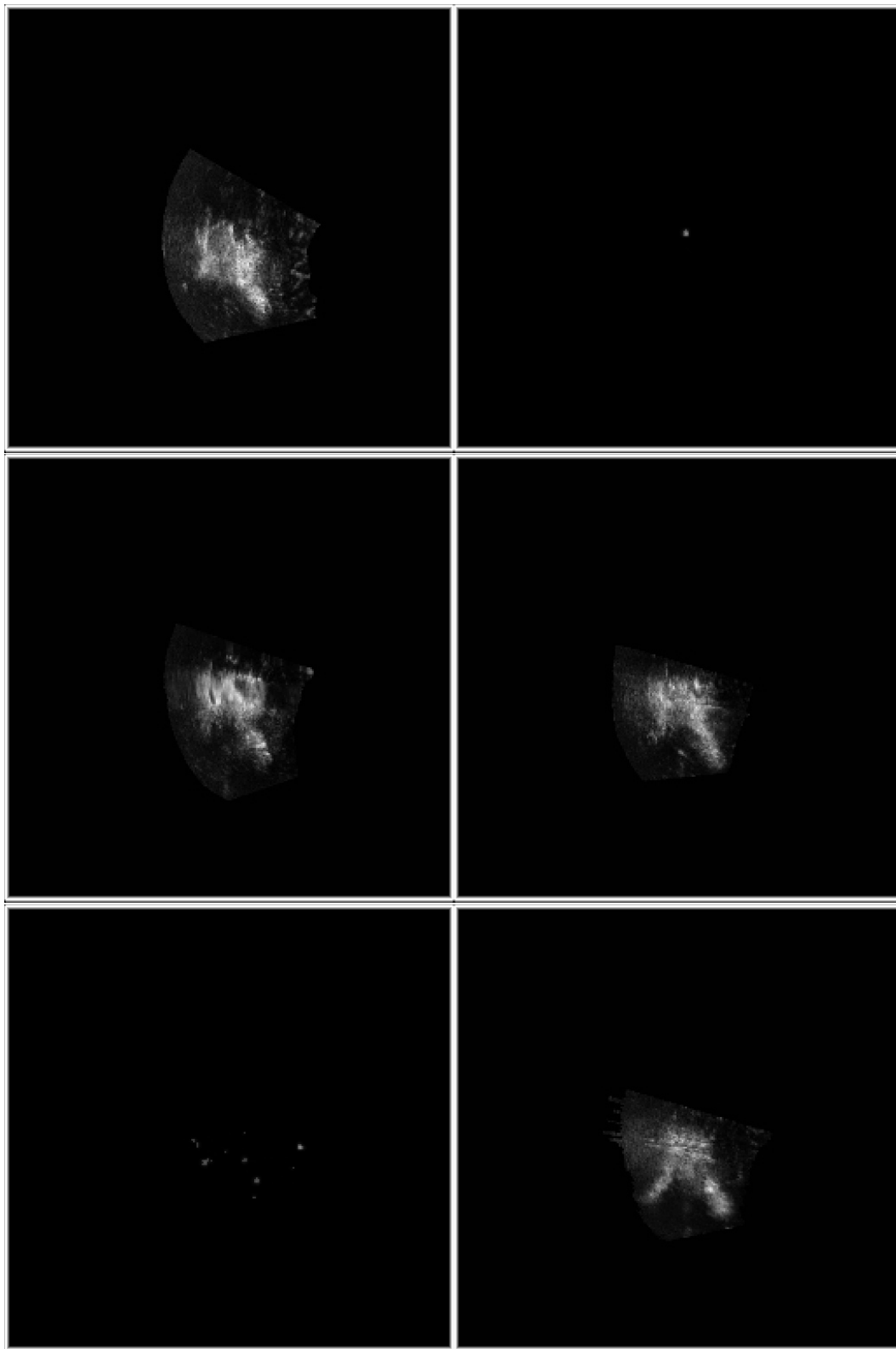


Figure 21: Dataset Tumor2. All slices corresponding to figure 18, before registration. Top left, UST #2. Top right, USA #2. Middle left, UST #3. Middle right, UST #4. Bottom left, USA #3. Bottom right, UST #5.

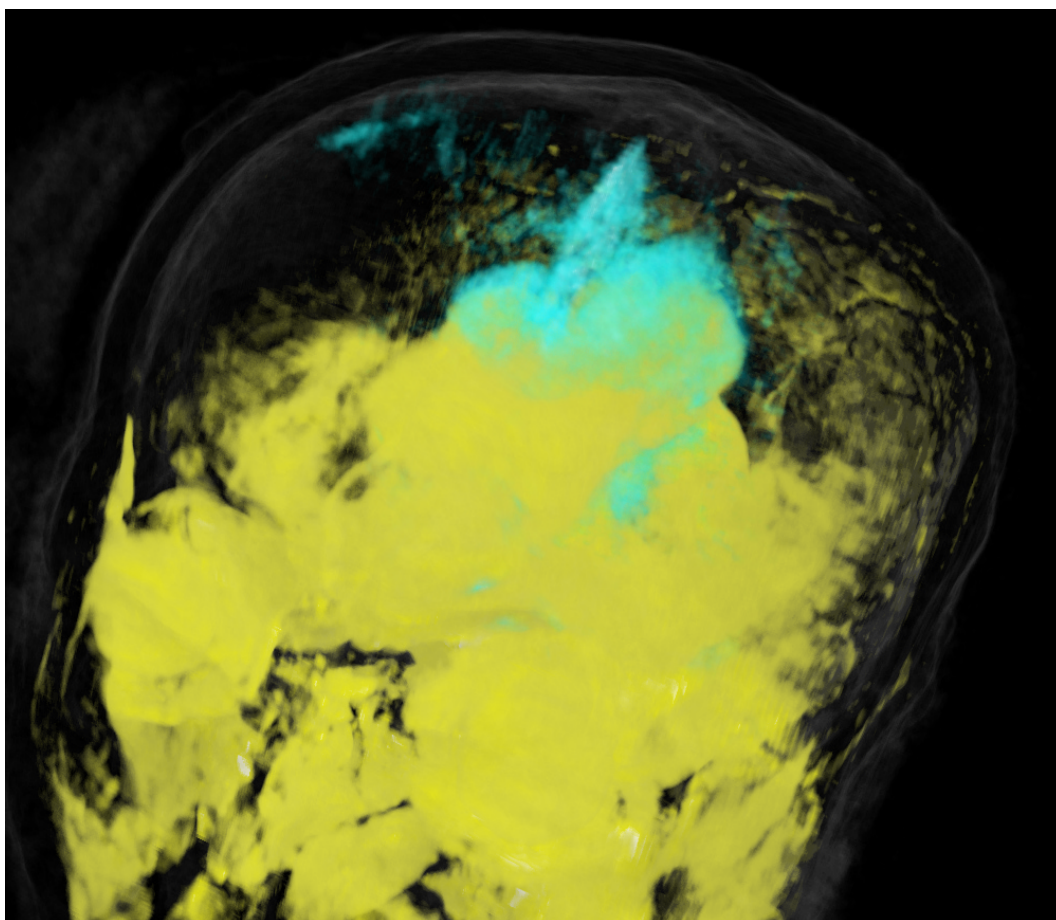


Figure 22: Dataset Tumor2. Volume renderings of MRI T2 (*yellow*) and UST #1 (*cyan*).

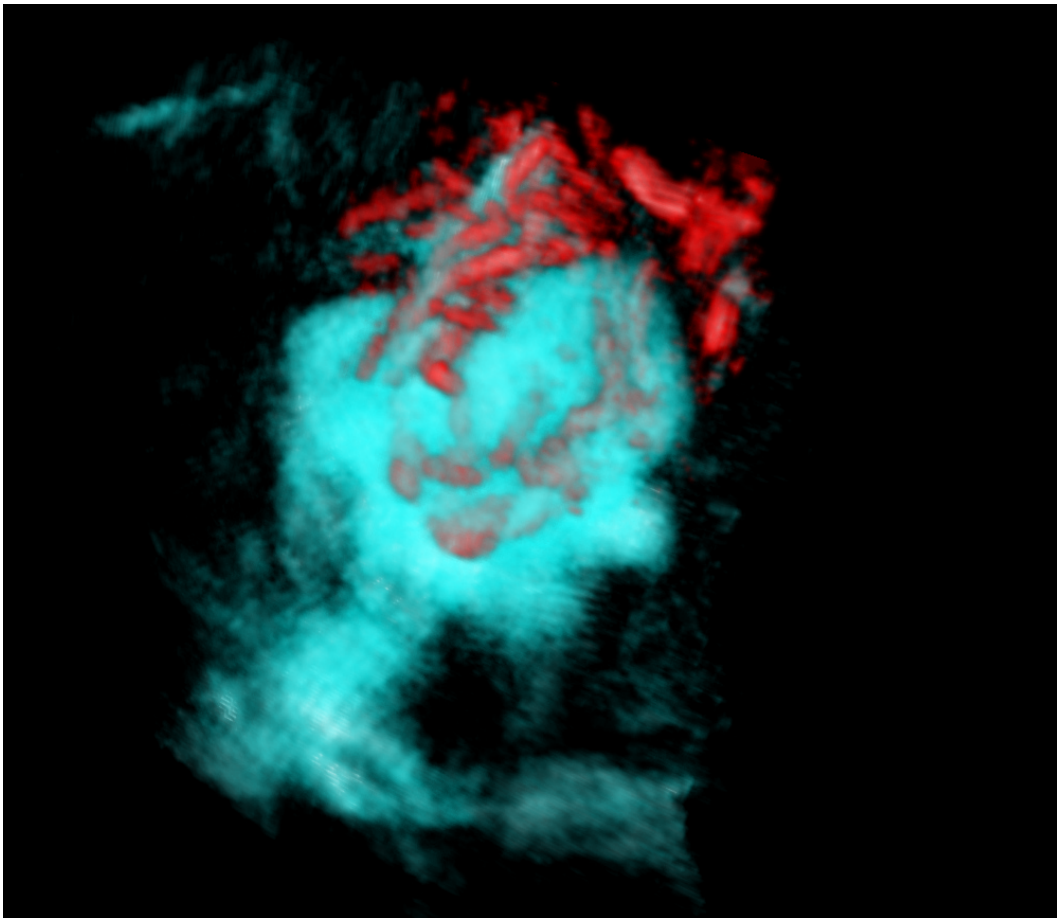


Figure 23: Dataset Tumor2. Volume renderings of UST #1 (*cyan*) and USA #1 (*red*).

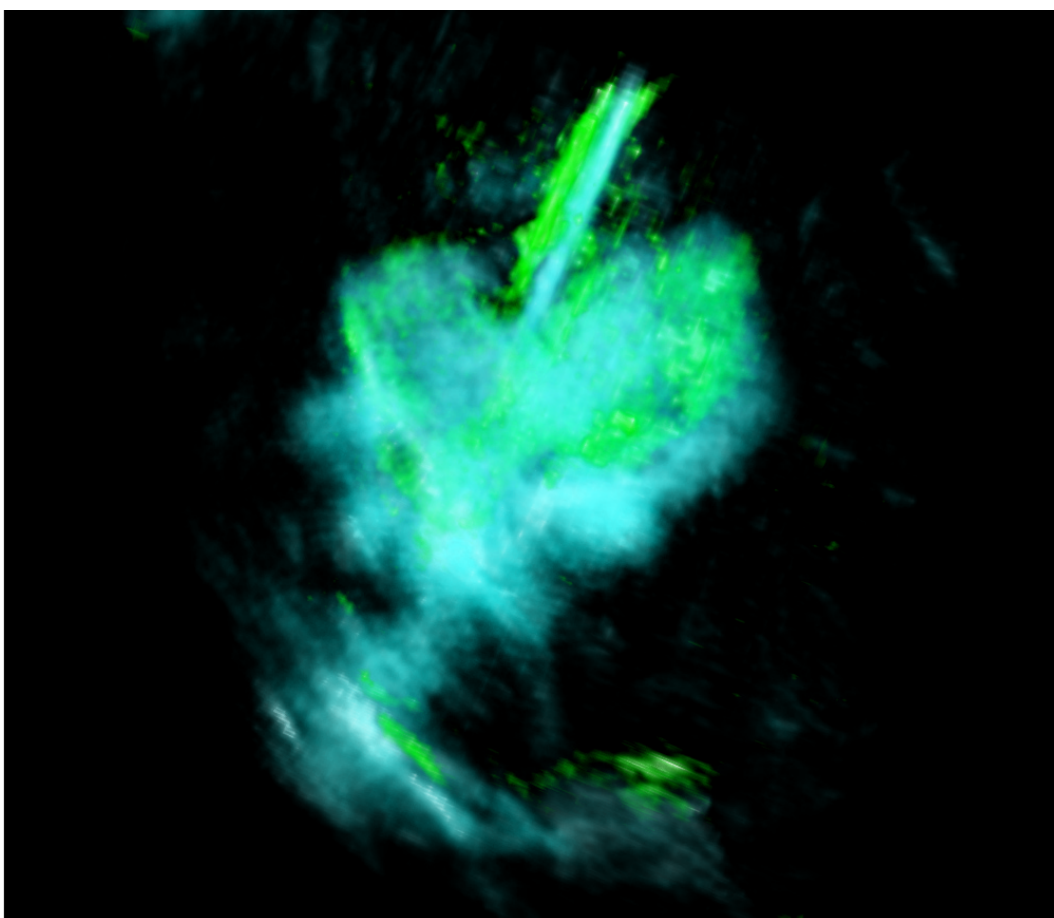


Figure 24: Dataset Tumor2. Volume renderings of UST #1 (*cyan*) and UST #2 (*green*).

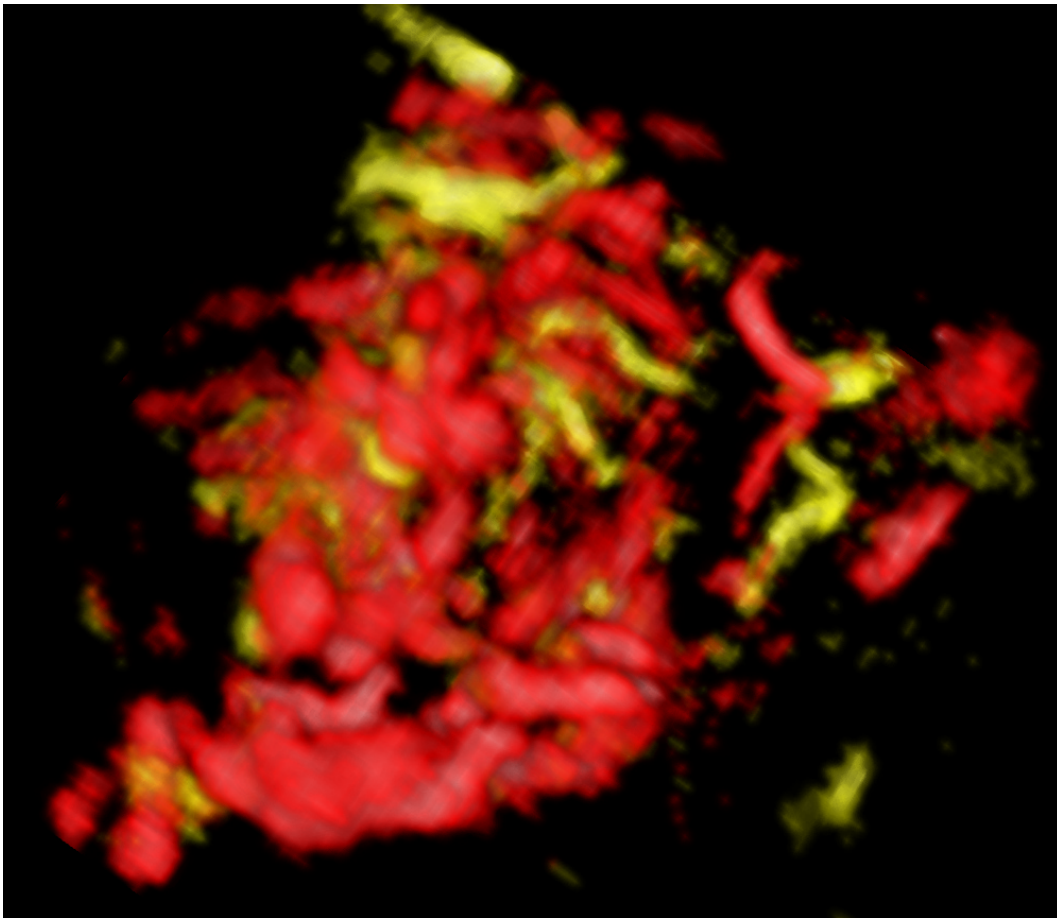


Figure 25: Dataset Tumor2. Volume renderings of USA #1 (*red*) and USA #2 (*yellow*).

10.4 Aneurism1

The Aneurism1 data set contains merely four volumes, of a patient with an aneurism¹¹ in one of the principal cerebral arteries. The surgical procedure attempts to clamp off the aneurism to avoid rupture of the blood vessel.

The first volume is a T1 weighted MRI, and the second is an MR Angio volume. The other two volumes are US Angios recorded before opening the dura, and after clamping the aneurism, respectively. Slices from all four images can be seen in figure 26, and are visibly mastered using the US volume. That the slice from the last US volume is seemingly empty is probably a result of the surgical procedure, where the aneurism was clamped, thereby reducing blood flow in the particular area. However, in large part this is also due to movement of the clamped blood vessel in respect to the surrounding tissue; the vessel seen in the first three slices is likely to be visible, but deformed, in another slice in the last volume. This illustrates the need for both rigid and non-rigid registration of the ultrasound volumes.

¹¹Swelling or ballooning of a blood vessel.

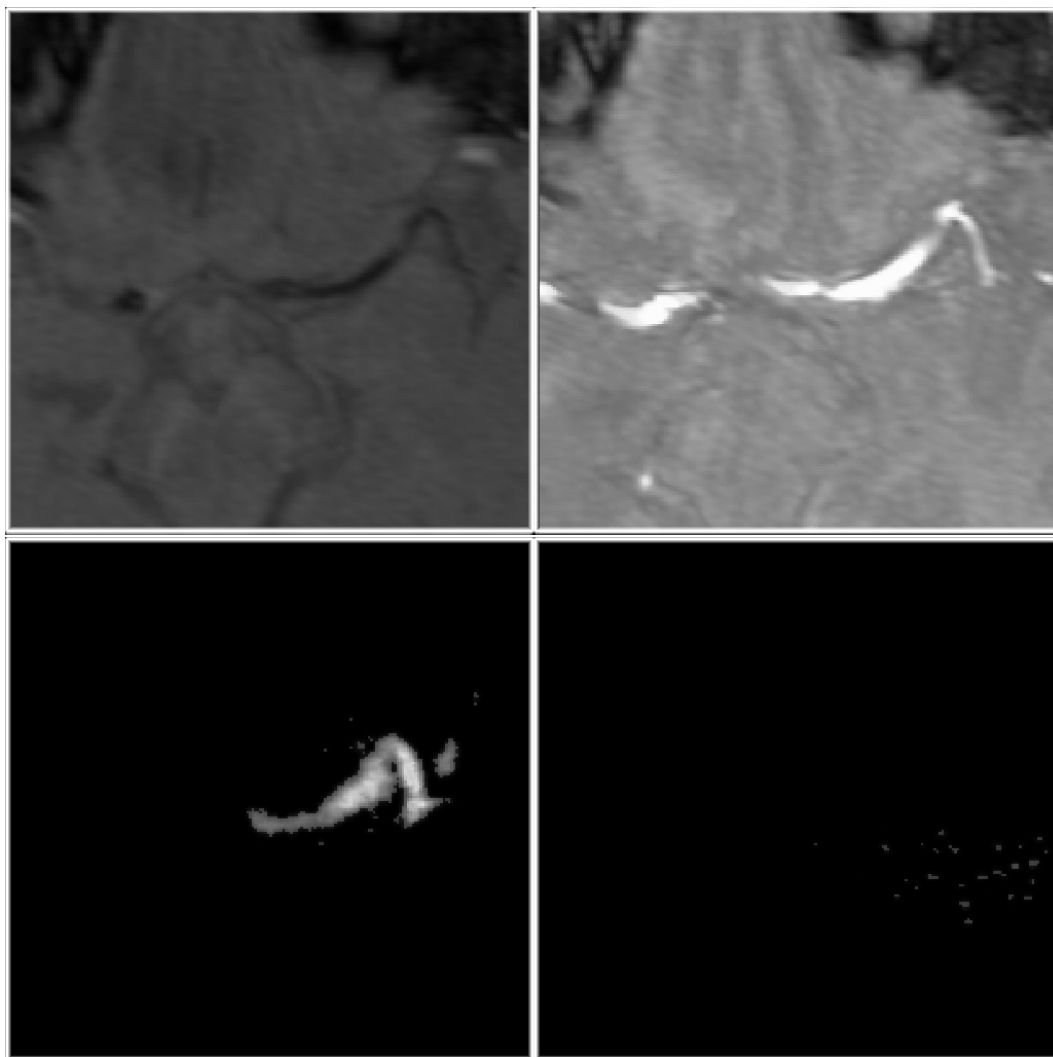


Figure 26: Dataset Aneurism1. Top left, MRI T1, and at top right, corresponding slice from MRA. Bottom left, USA #1, from before clamping. Bottom right, USA #2, from after clamping. Unregistered volumes.

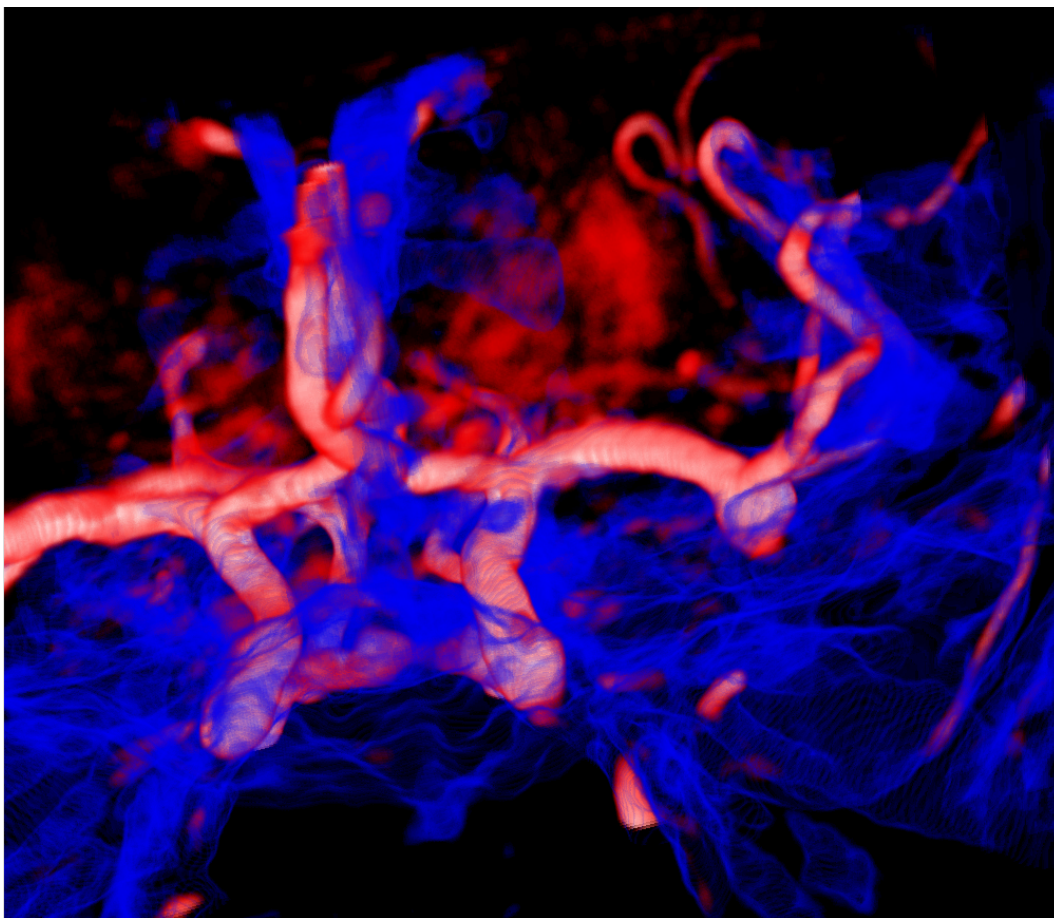


Figure 27: Dataset Aneurism 1. Volume renderings of MRI T1 (*blue*) and MRA (*red*).

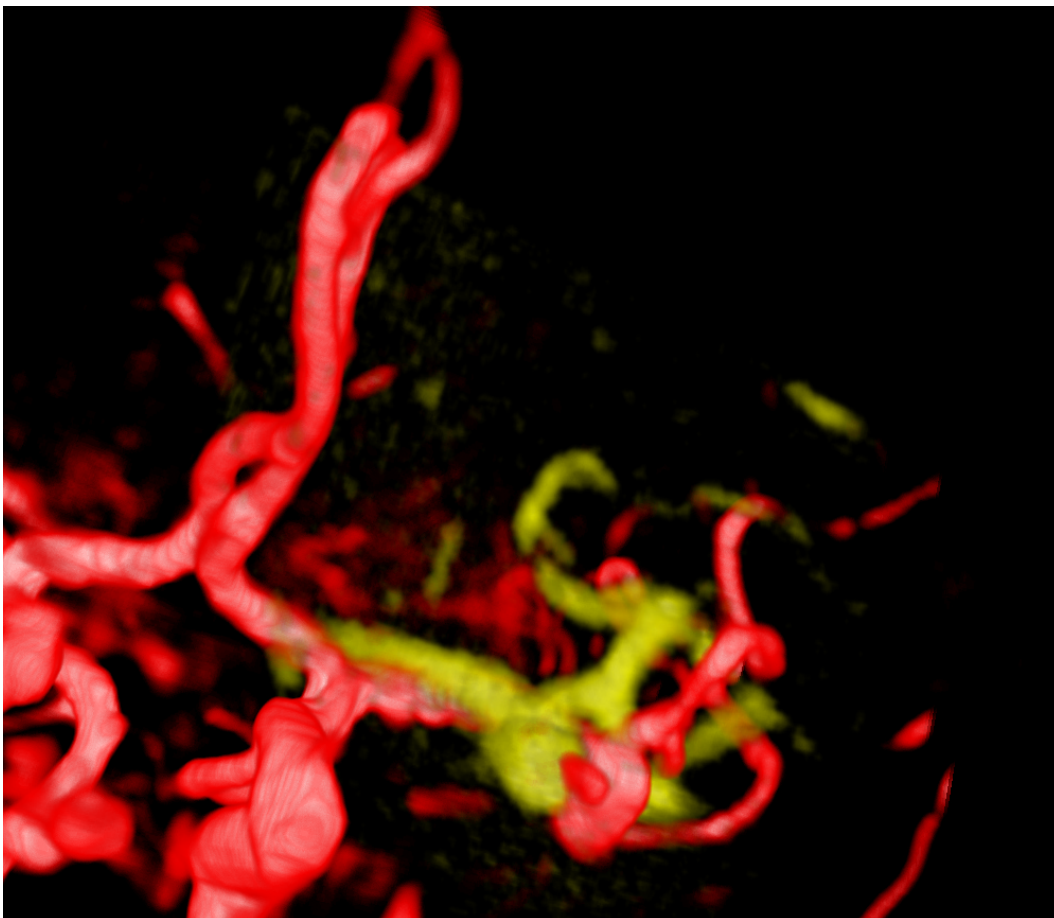


Figure 28: Dataset Aneurism 1. Volume renderings of MRA (*red*) and USA #1 (*yellow*).

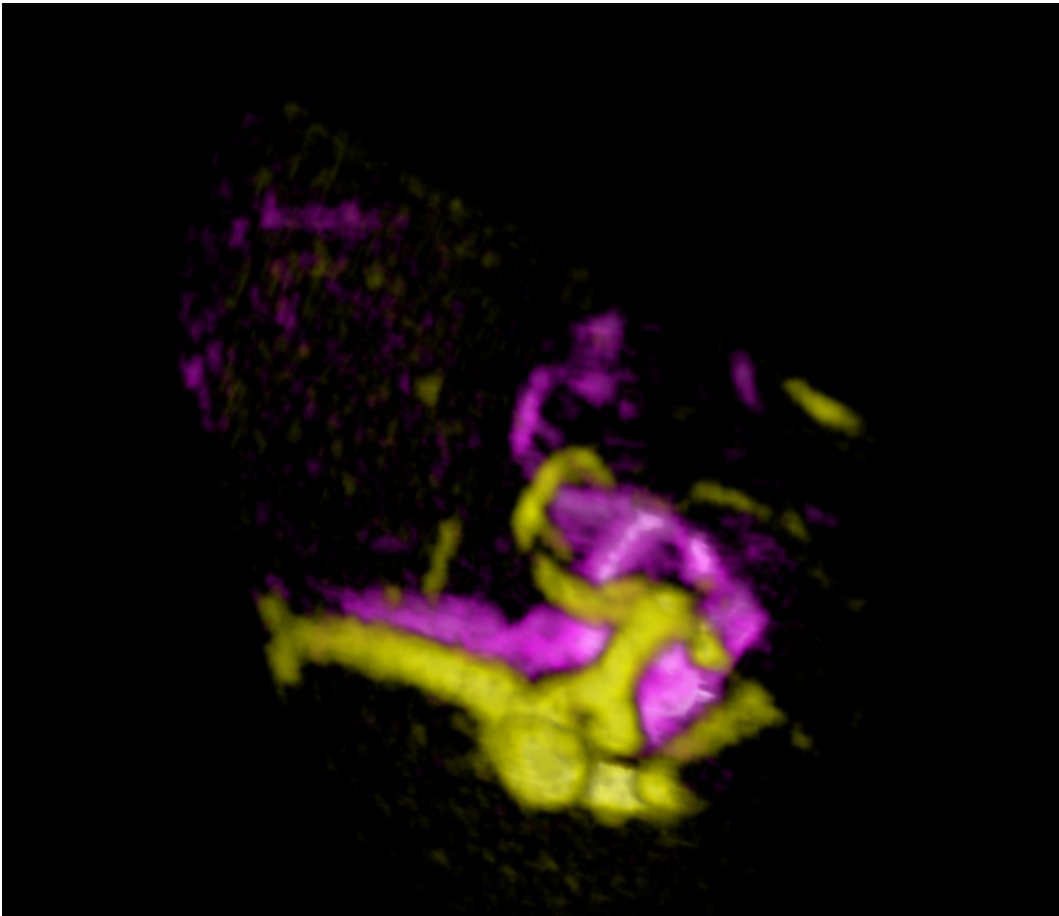


Figure 29: Dataset Aneurism 1. Volume renderings of USA #1 (*yellow*) and USA #2 (*pink*).

11 Tests and methods

A table including all the registration attempts that are conducted, for each of the three phases and for volumes from all three datasets, is seen in table 1. Note that, while the US volumes are always referred to as “moving” in phase 2, it is the MRI volumes that will be transformed, to fit the updated positional information in the US.

Phase 1		
Reg.op. no.	Fixed	Moving
• 1	Tumor 1 fMRI tongue	Tumor 1 MRI T2
• 2	Tumor 1 fMRI tongue	Tumor 1 MRA
3	Tumor 1 MRI T2	Tumor 1 fMRI tongue
4	Tumor 1 MRA	Tumor 1 fMRI tongue
• 5	Tumor 2 fMRI Tongue	Tumor 2 MRI T2
6	Tumor 2 MRI T2	Tumor 2 fMRI Tongue
• 7	Aneurism 1 MRI T1	Aneurism 1 MRA
Phase 2		
Reg.op. no.	Fixed	Moving
1	Tumor 1 fMRI tongue	Tumor 1 UST #1
• 2	Tumor 1 MRI T2	Tumor 1 UST #1
• 3	Tumor 1 MRA	Tumor 1 USA #1
4	Tumor 2 fMRI Tongue	Tumor 2 UST #1
5	Tumor 2 MRI T2	Tumor 2 UST #1
• 6	Aneurism 1 MRA	Aneurism 1 USA #1
Phase 3		
Reg.op. no.	Fixed	Moving
1	Tumor 1 UST #1	Tumor 1 UST #2
2	Tumor 1 USA #1	Tumor 1 USA #2
3	Tumor 1 USA #2	Tumor 1 USA #3
4	Tumor 1 UST #2	Tumor 1 UST #3
5	Tumor 1 UST #3	Tumor 1 UST #4
6	Tumor 2 UST #1	Tumor 2 UST #2
7	Tumor 2 USA #1	Tumor 2 USA #2
8	Tumor 2 UST #2	Tumor 2 UST #3
9	Tumor 2 USA #2	Tumor 2 USA #3
10	Tumor 2 UST #3	Tumor 2 UST #4
11	Tumor 2 UST #4	Tumor 2 UST #5
12	Aneurism 1 USA #1	Aneurism 1 USA #2

Table 1: All registrations should be performed, and those that will be presented (marked with a •)

12 Phase 1 registration results

The first registration application, called `RegApp1`, is implemented using the ITK classes described in chapter 7.2.1. This includes using a Mutual Information metric, a translational transform and a gradient descent optimizer. These are quite standard building blocks in a multi-modal registration suite, and should be able to capture the differences between the preoperative MRI images, and bring them into alignment.

All the parameters discussed in chapter 7.2.1 are available for the user of `RegApp1` to set, including the number of spatial samples, the maximal number of iterations and maximum and minimum step length of the optimizer, but all have reasonable default values. The standard deviation of the Gaussian used in the kernel density estimation is seldom set explicitly, as the default value of 0.4 is reckoned as a generally valid setting, for normalized input images.

12.1 Tumor 1

Results from registration operation number 1 and 2 are presented below.

12.1.1 Registration # 1

The first registration operation, using the T1 + Tongue fMRI from dataset Tumor1 as the fixed image, and the MRI T2 from the same dataset as the moving, is illustrated in figure 30. Here the situation before registration is seen, and the registration error is visible in the relative offset of the fiducials in the two volumes. The same situation can be seen in the top two images of figure 31, where the internal structures of the brain are emphasized. In the bottom part of 31, the result after successful registration (execution # 3-N338US) is seen. Note particularly the difference around the ventricle. Further, figure 32 displays the scene of figure 30, after registration (execution # 17).

Registration # 3-N338US has been done using the US-mastered version of the Tumor1 dataset, while the other registrations have been performed on the MRI-mastered volumes, as discussed in chapter 10. The parameters used for the registrations giving the results seen in figures 31 and 32, are included in table 2, and the resulting transformations can be seen in table 3.

Parameter	Value	
	# 17	# 3-N338US
Max no. of iterations	5000	5000
Samples	5000	8000
Max. step length	1.0	5.0
Min. step length	0.001	0.001

Table 2: Parameters used for results seen in figure 32 and 31.

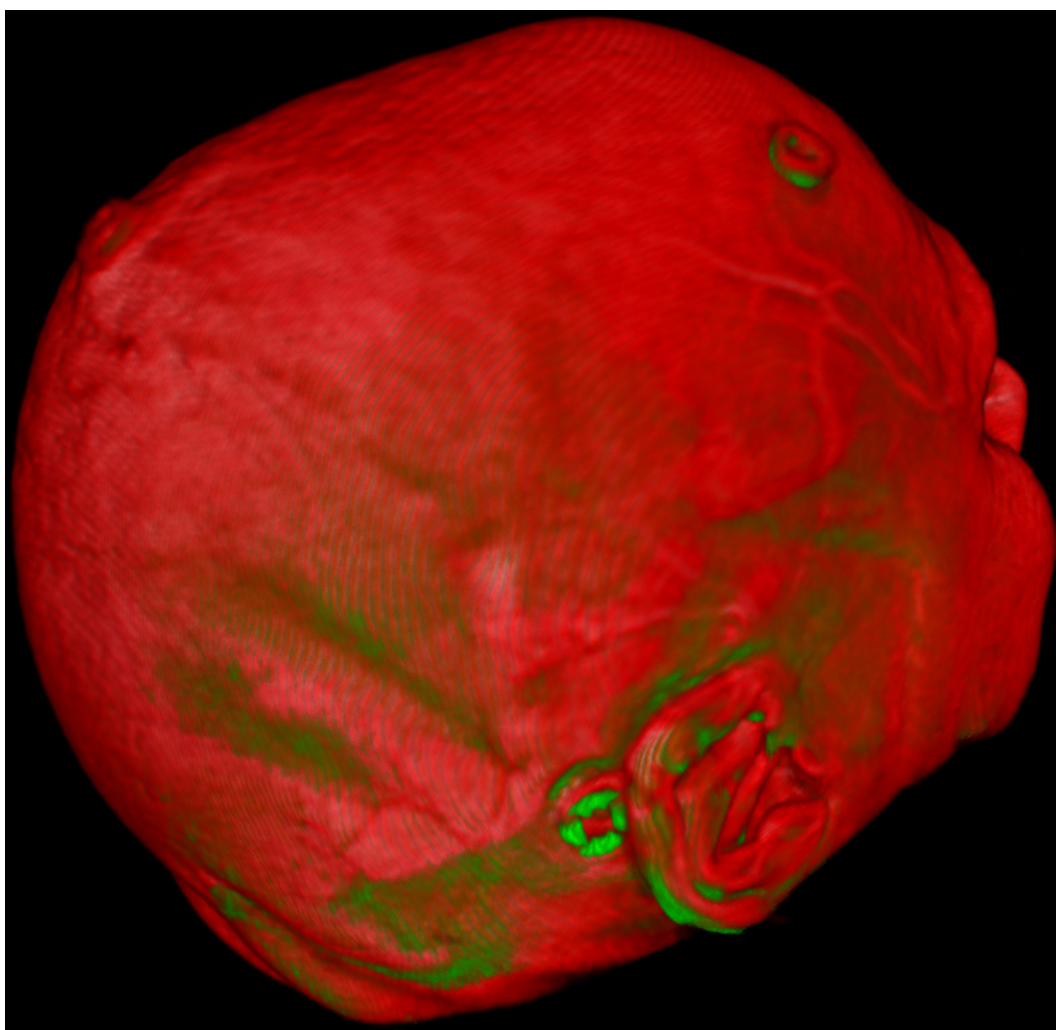


Figure 30: Results Tumor 1. Volume renderings of fMRI+T1 (red) and T2 (green) before registration.

Parameter	Value	
	# 17	# 3-N338US
<i>X</i> translation	0.0831206	-0.33883
<i>Y</i> translation	-0.0117184	0.700214
<i>Z</i> translation	-1.89721	-1.9349
<i>MI</i> metric value	0.410462	0.414696
No. of iterations used	24.	23.

Table 3: Transform resulting from registrations, as seen in figures 32 and 31.

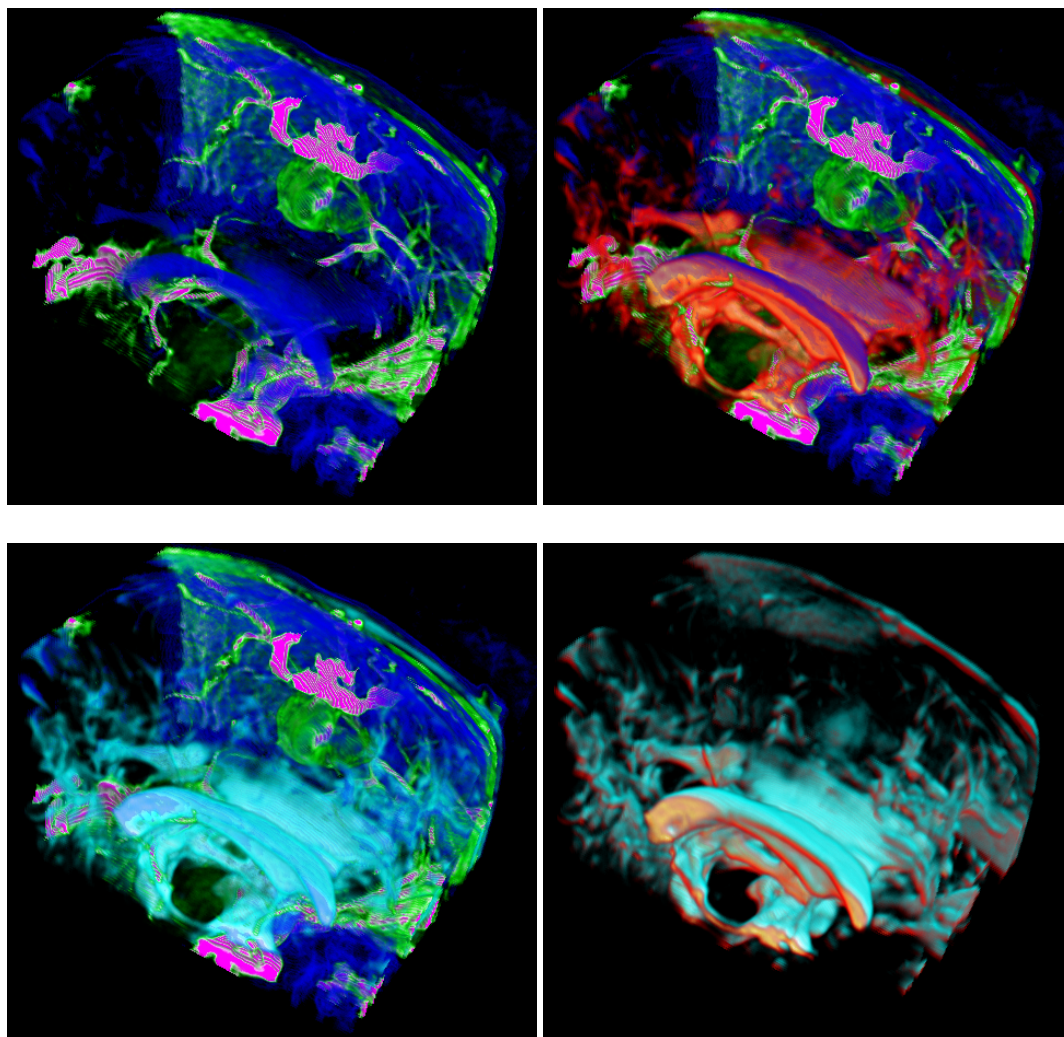


Figure 31: Illustration of result of registration of MRI-T2 and fMRI+MRI-T1 using the phase 1 registration application.

Volumes are thresholded for visualization. Blue-green is T1, with magenta (pink) parts from fMRI overlay. Original T2 is red, and T2 after registration is cyan (turquoise).

Top left: fMRI/T1. **Top right:** fMRI/T1 + T2.

Bottom right: fMRI/T1 + registered T2. **Bottom left:** Original and registered T2.

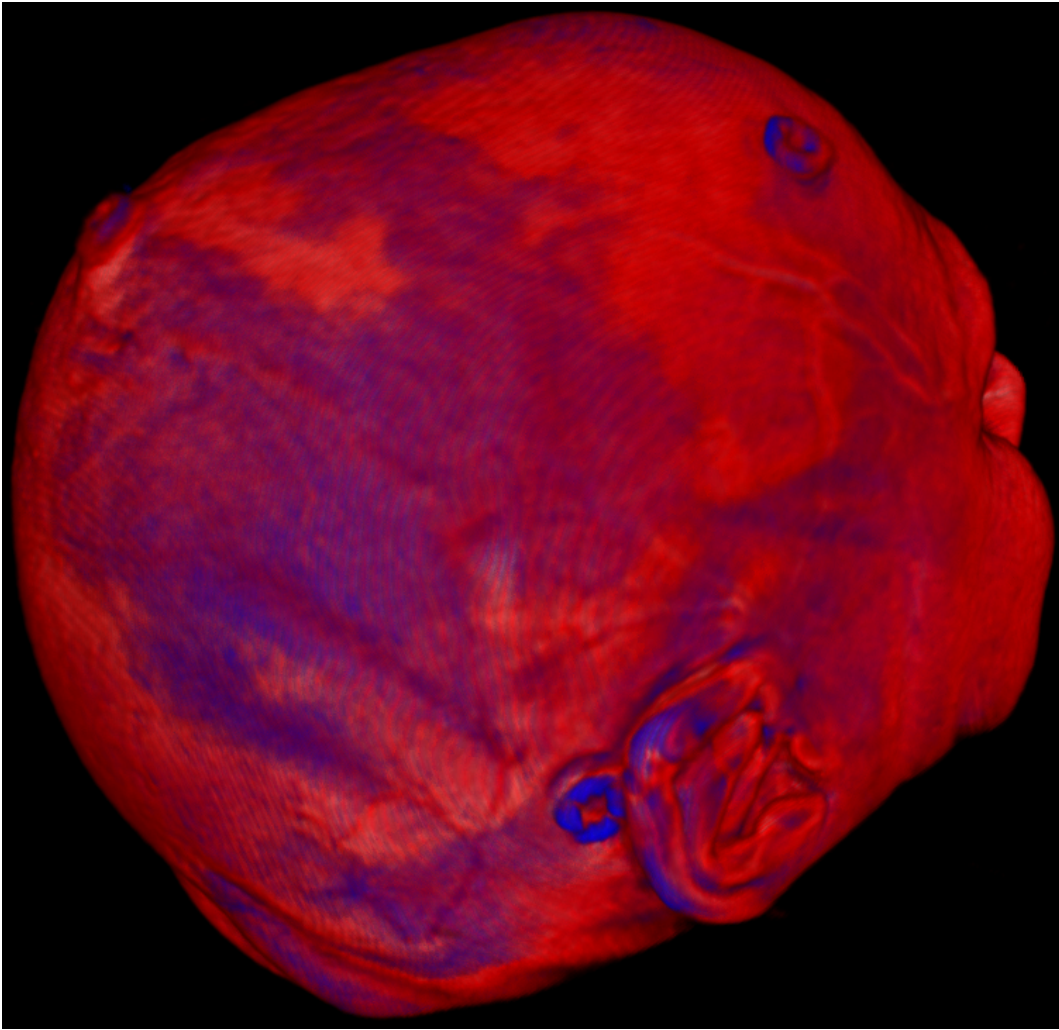


Figure 32: Results Tumor 1. Volume renderings of fMRI+T1 (*red*) and MRI T2 (*purple*) after registration.

12.1.2 Registration # 2

The second registration operation is between the same T1 + fMRI, and an MR angio volume of the patient. The two volumes prior to registration can be seen in a 3D rendering in figure 33, with the misalignment apparent at the fiducials both at the top of the patient's head, and behind the ear; most prominently in the vertical direction. After registration run # 29, where the resulting translations were calculated as indicated in table 4, the volumes are brought into alignment, see figure 34. The parameters for this registration are included in table 5.

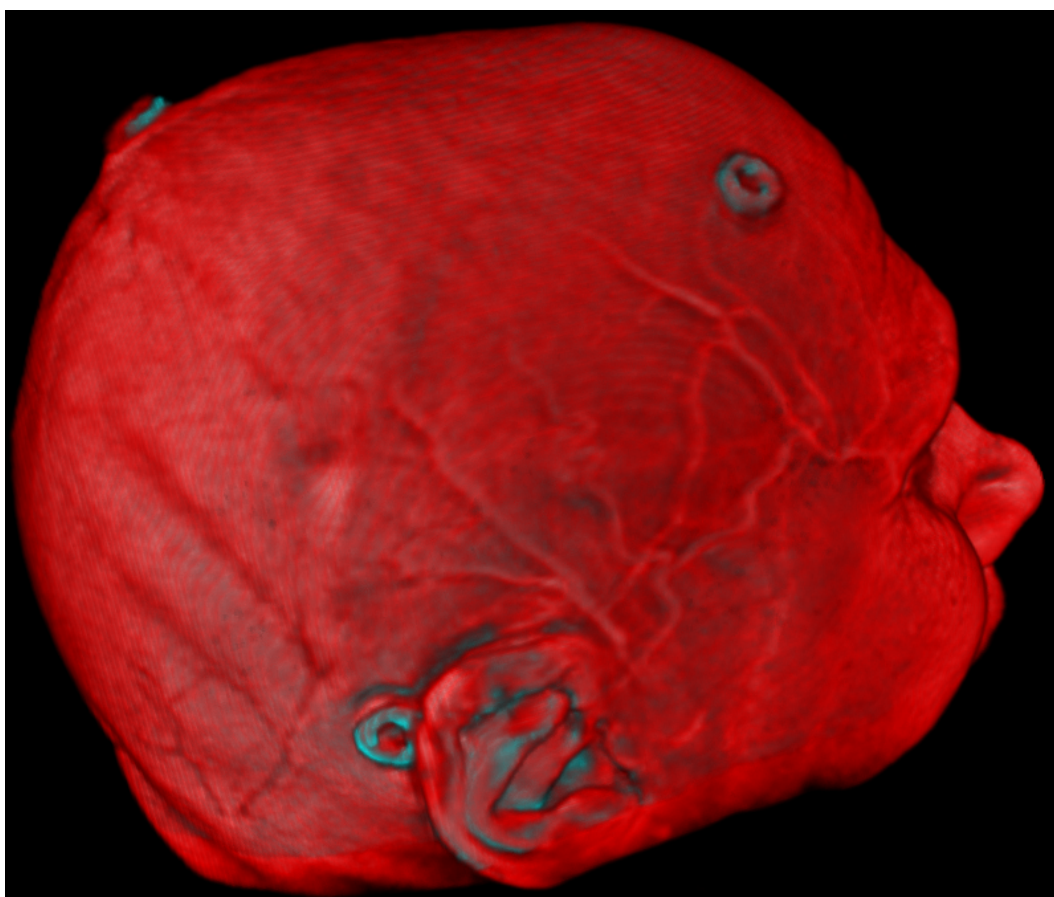


Figure 33: Results Tumor 1. Volume renderings of fMRI+T1 (*red*) and MRA (*green*) before registration.

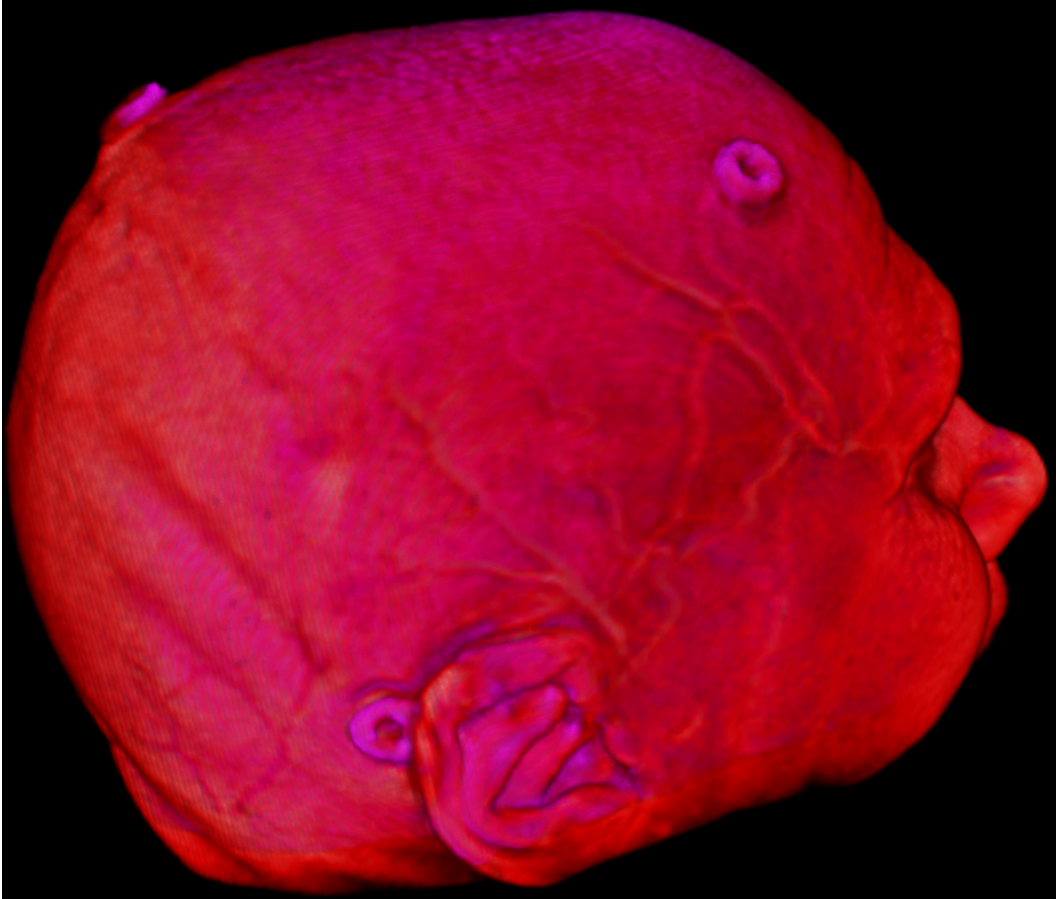


Figure 34: Results Tumor 1. Volume renderings of fMRI+T1 (*red*) and MRA (*magenta*) after registration.

Parameter	Value
	# 29
<i>X</i> translation	0.220904
<i>Y</i> translation	-0.0706977
<i>Z</i> translation	-0.826528
<i>MI</i> metric value	0.357882
No. of iterations used	22.

Table 4: Transform resulting from registration of fMRI and MRA, as seen in figure 34.

Parameter	Value
	# 29
Max no. of iterations	2000
Samples	5000
Max. step length	1.0
Min. step length	0.001

Table 5: Parameters used for results seen in figure 34.

12.2 Tumor 2

Results for fifth registration operation is presented in the following

12.2.1 Registration # 5

The fifth registration operation, pertaining to the second dataset, *Tumor2*, is performed on the volumes visualized in figure 35. The volumes are one T1+fMRI and one MRI T2. The misalignment is most clearly visible in the fiducial on the upper frontal part of the head. After `RegApp1` was used on the volumes, with the parameters indicated in table 6, the volumes ended up as shown in figure 36. The transformation found by the registration application is included in table 7.

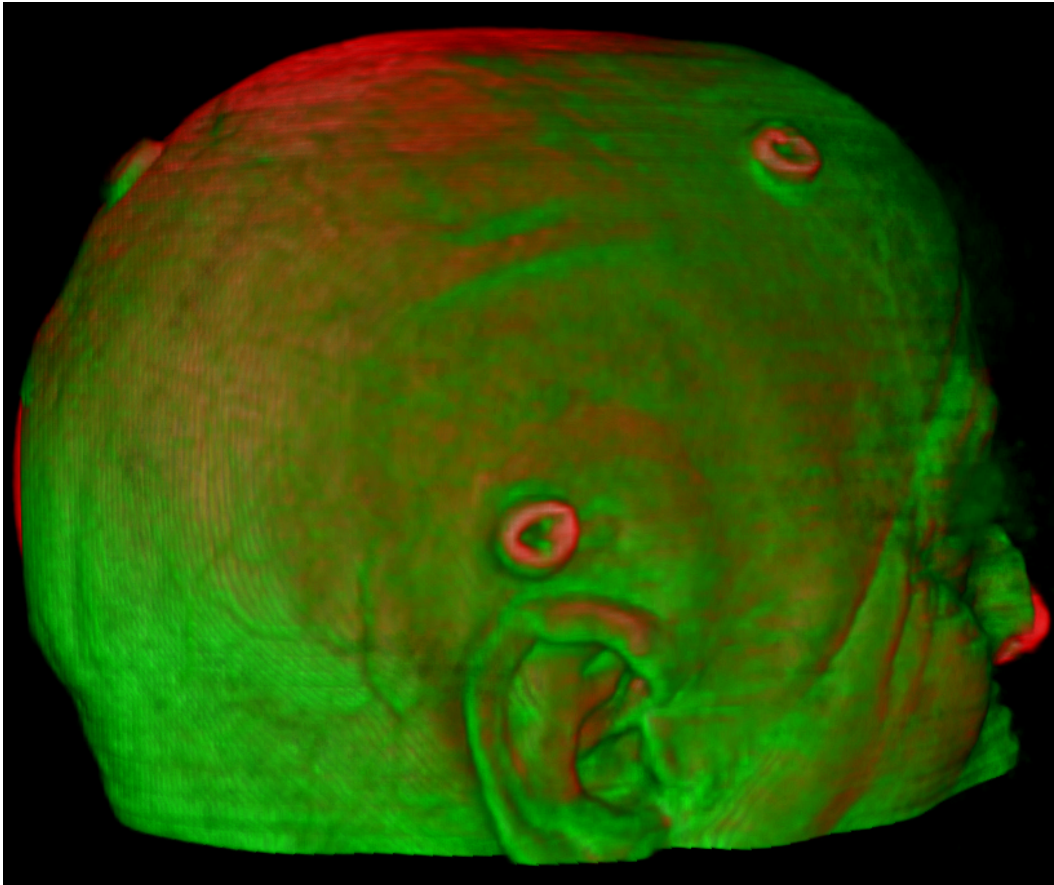


Figure 35: Results Tumor 2. Volume renderings of fMRI+T1 (red) and MRI T2 (green) before registration.

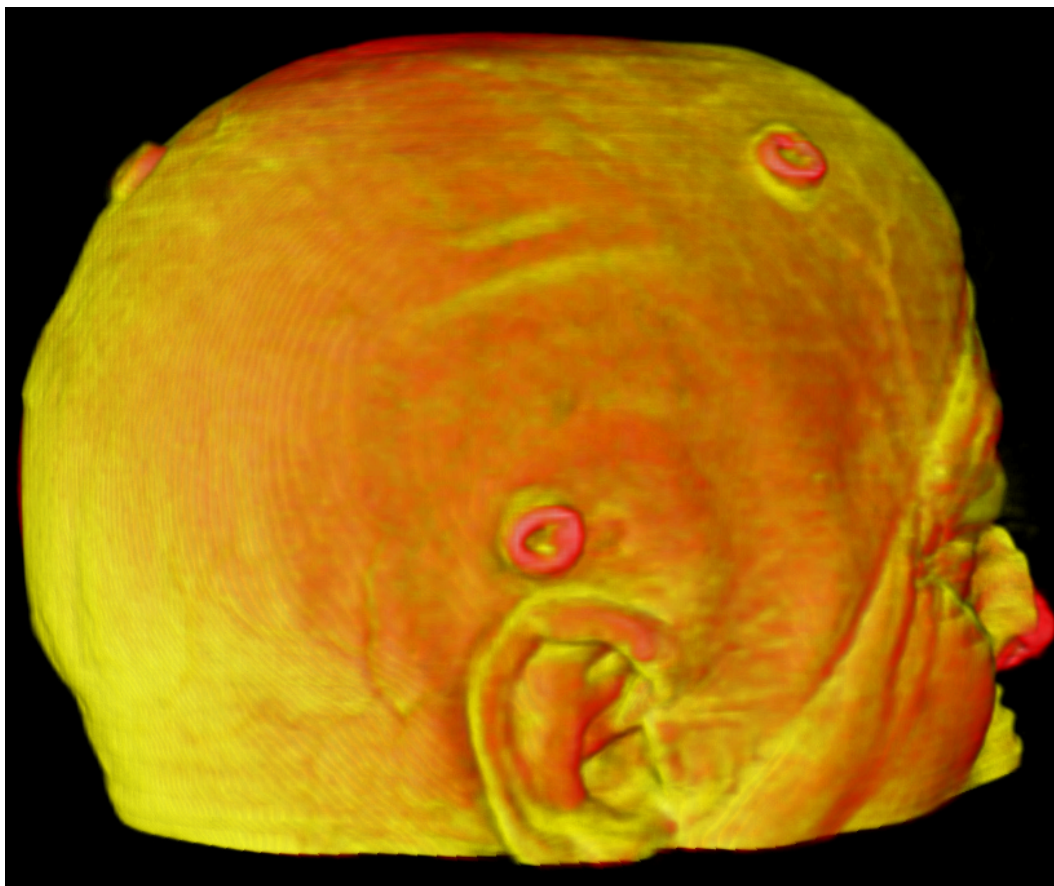


Figure 36: Results Tumor 2. Volume renderings of fMRI+T1 (red) and MRI T2 (yellow) after registration.

Parameter	Value
	# 113
Max no. of iterations	5000
Samples	5000
Max. step length	5.0
Min. step length	0.01

Table 6: Parameters used for results of registration of fMRI and T2, as seen in figure 36.

Parameter	Value
	# 113
<i>X</i> translation	0.205492
<i>Y</i> translation	-0.574043
<i>Z</i> translation	-0.967832
<i>MI</i> metric value	0.49915
No. of iterations used	15.

Table 7: Transform resulting from registration of fMRI and T2, as seen in figure 36.

12.3 Aneurism 1

The results for the registration of the MRI in the Aneurism 1 dataset are presented below.

12.3.1 Registration # 7

The seventh registration operation in the first phase is the only one on volumes in the Aneurism 1 dataset. In figure 37, the MRI T1 and MRA can be seen volume rendered, visibly not registered. Note especially the apparent offset in the blood vessels at the top, right hand side, and in the middle and top left hand side. In figure 38, the same scene after registration #149 can be seen. The transformation made on the moving volume is given in table 9. The result of another run, #161, of the registration application is seen in figure 39, also with the resulting transformation in table 9. These results are obtained using the parameters in table 8 with the `RegApp1`.

Parameter	Value	
	# 149	# 161
Max no. of iterations	2000	5000
Samples	5000	5000
Max. step length	1.0	1.0
Min. step length	0.001	0.001

Table 8: Parameters used for results seen in figure 38 and 39.

Parameter	Value	
	# 149	# 161
X translation	-0.271884	-0.229481
Y translation	0.8281	0.826222
Z translation	-0.856083	-0.849426
MI metric value	0.453039	0.47733
No. of iterations used	20.	24.

Table 9: Transform resulting from registration of MRI T1 and MRA, as seen in figure 38.

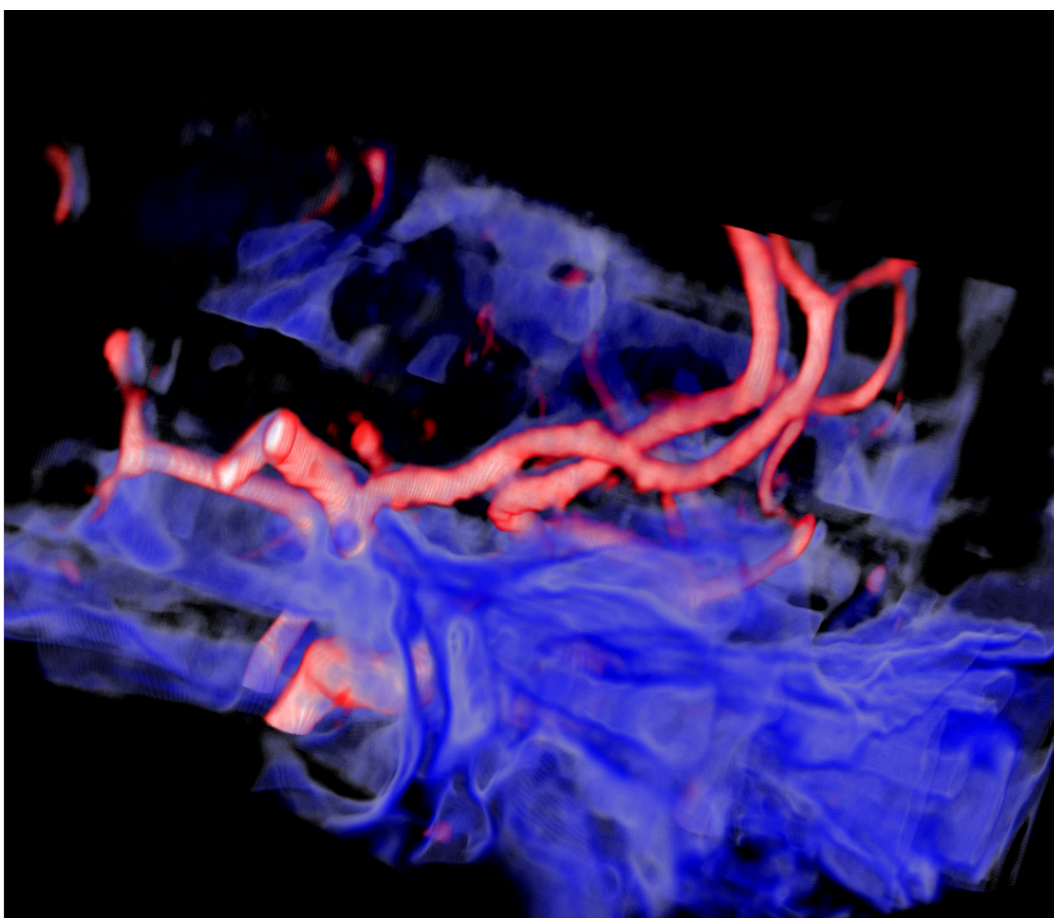


Figure 37: Results Aneurism 1. Volume renderings of MRI T1 (*blue*) and MRA (*red*) before registration.

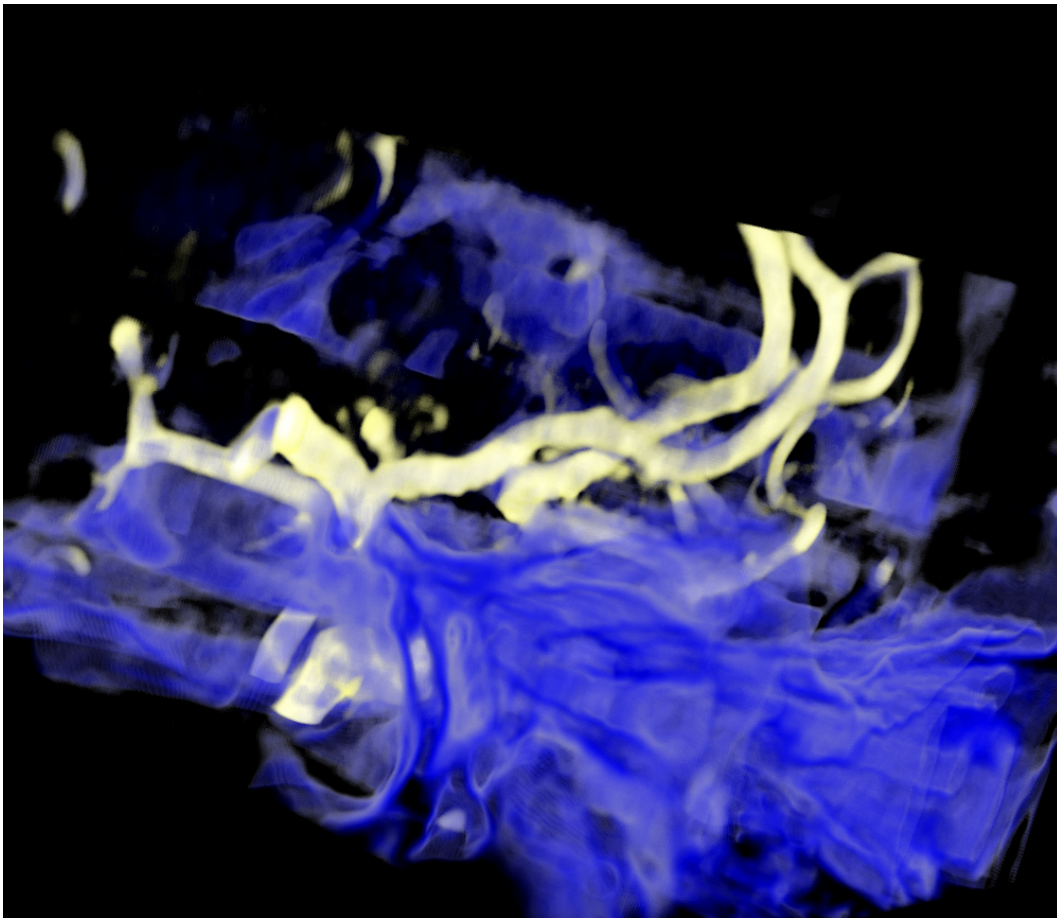


Figure 38: Results Aneurism 1. Volume renderings of MRI T1 (*blue*) and MRA (*yellow*) after registration.

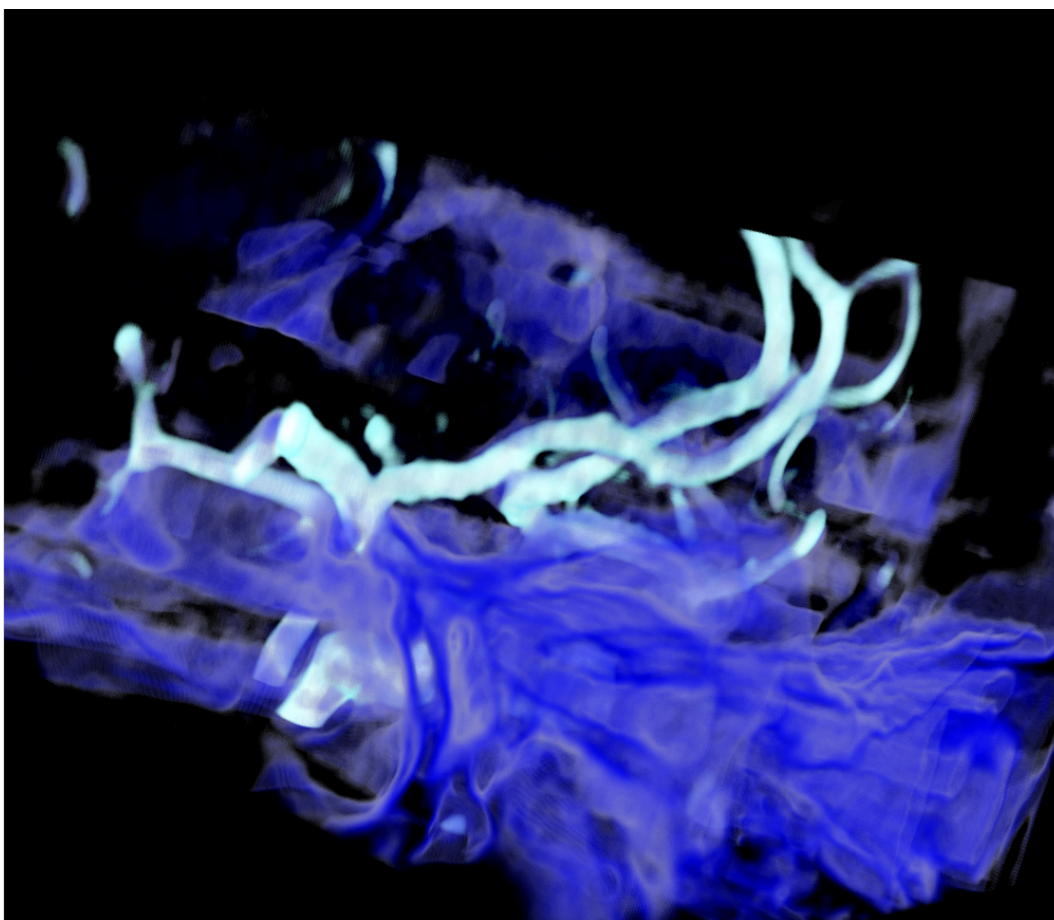


Figure 39: Results Aneurism 1. Volume renderings of MRI T1 (*blue*) and MRA (*cyan*) after registration.

13 MRI-to-3DUS registration results

Two separate registration applications have been developed to perform the multi-modal MRI-to-US registration of the second phase. Each application has been implemented using the methods and classes discussed in chapter 7.2.2 and 7.2.3, for `RegApp2` and `RegApp2b`, respectively. The `RegApp2` uses the Normalized Mutual Information similarity metric, a translational transform and a (1+1)-evolutionary optimizer. Meanwhile, the `RegApp2b` employs a Mutual Information metric (using the histogram based approach due to Mattes', see 7.2.3), a rigid transform accommodating translations and rotations, and a version of the gradient descent optimizer. Both applications also use gradient magnitude filtering with smoothing to extract edge information from the input images, to help the alignment.

The `RegApp2` application makes most parameters available to the end user, with all-round default values, with the exception of the transformation parameter scaling of the metric. As only translational transformations are considered, no scaling is deemed necessary. The remaining parameters to be set by the user are thus: number of histogram bins, initial search radius, maximum number of iterations, standard deviation of the Gaussian smoothing in the gradient magnitude filter, and ϵ ; the minimal value of the norm of the covariance matrix.

`RegApp2b` also gives the user the possibility of tuning the parameters for the underlying methods, apart from the scaling parameters, which are fixed according to the "rule of thumb" presented in chapter 7.2.3, and the multiresolution subdivision scheme, which is kept at the ITK default, also presented in chapter 7.2.3. The user definable parameters are: number of histogram bins, number of spatial samples, maximum and minimum step lengths for the optimizer, and number of levels in the multiresolution image pyramid.

13.1 Tumor 1

The Tumor1 dataset gives the opportunity to register both MRI and US angio volumes and ordinary tissue volumes. These are handled separately below.

13.1.1 MR-US Tissue

The possible second phase registrations of tissue volumes of the Tumor1 dataset are one using the fMRI+T1, and another using the MRI T2, as fixed volumes, with the first US tissue volume as moving. Recall that while the US volumes are denoted as "moving", in this implementation it is the MRI (fixed image) that will be transformed according to the result of the registration (using the inverse of the transform found by the registration framework).

The `RegApp2` (NMI-based) is used on the T2 and UST#1 volumes, seen respectively in figures 40 and 41 — and together, in figure 42, with the parameters specified in table 10. This gave the output indicated in table 11, which led to the positioning of the MRI T2 with respect to the US, as seen in figure 43. Here, the US-mastered version of the volumes is used; the `RegApp2` was not able to get a decent registration of the MRI-mastered versions of these volumes.

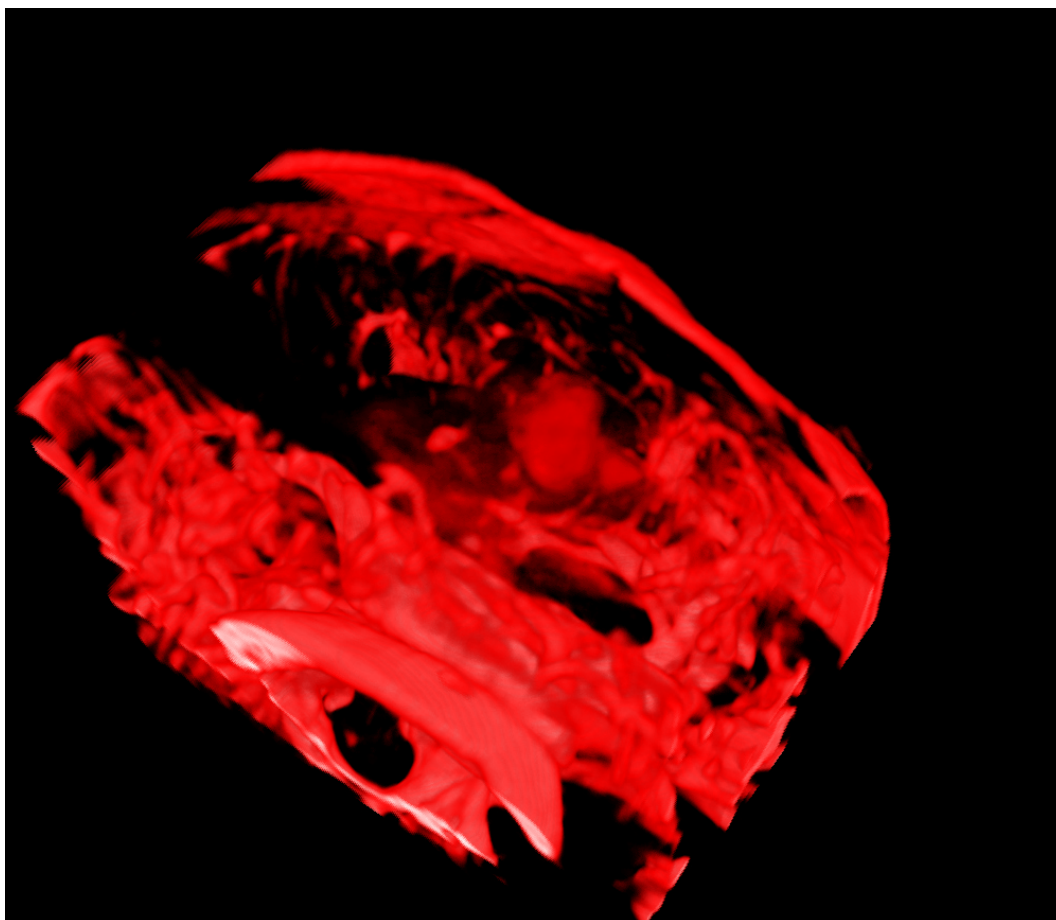


Figure 40: Tumor 1, phase 2. Volume rendering of MRI T2 (*red*) before registration.

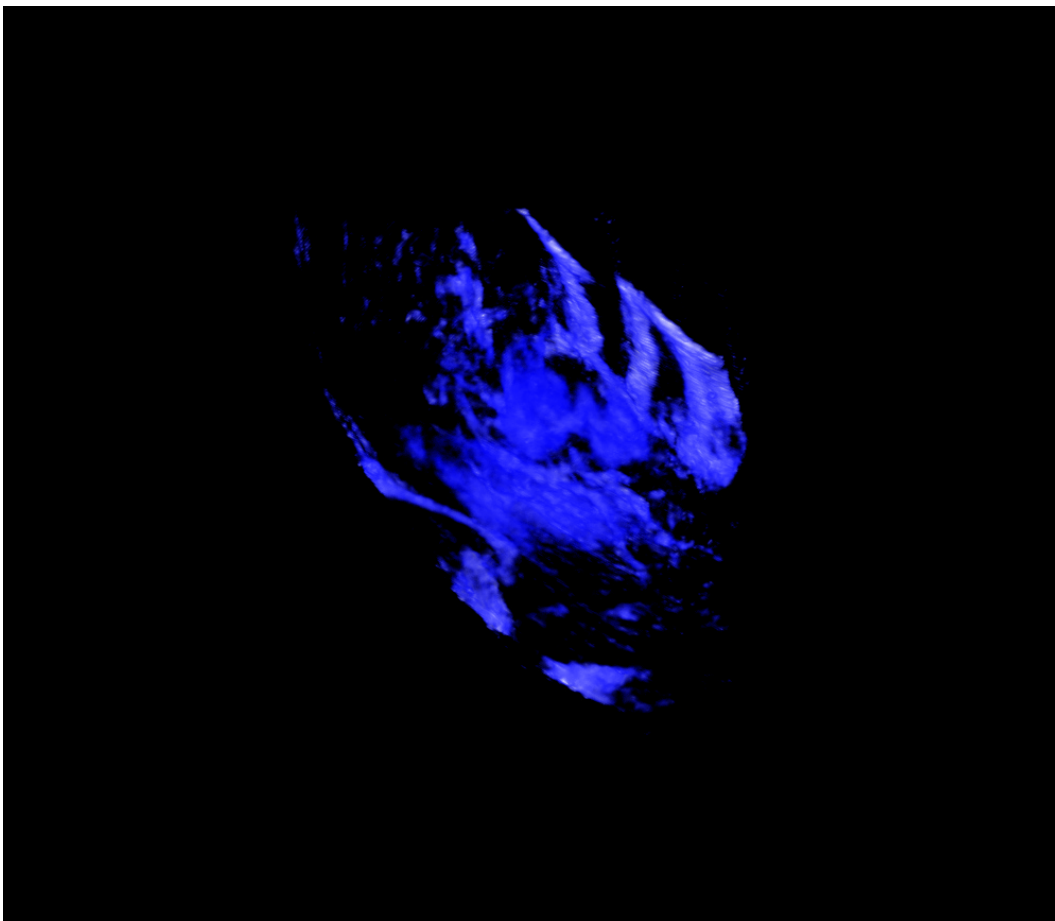


Figure 41: Tumor 1, phase 2. Volume rendering of UST (*blue*) before registration.

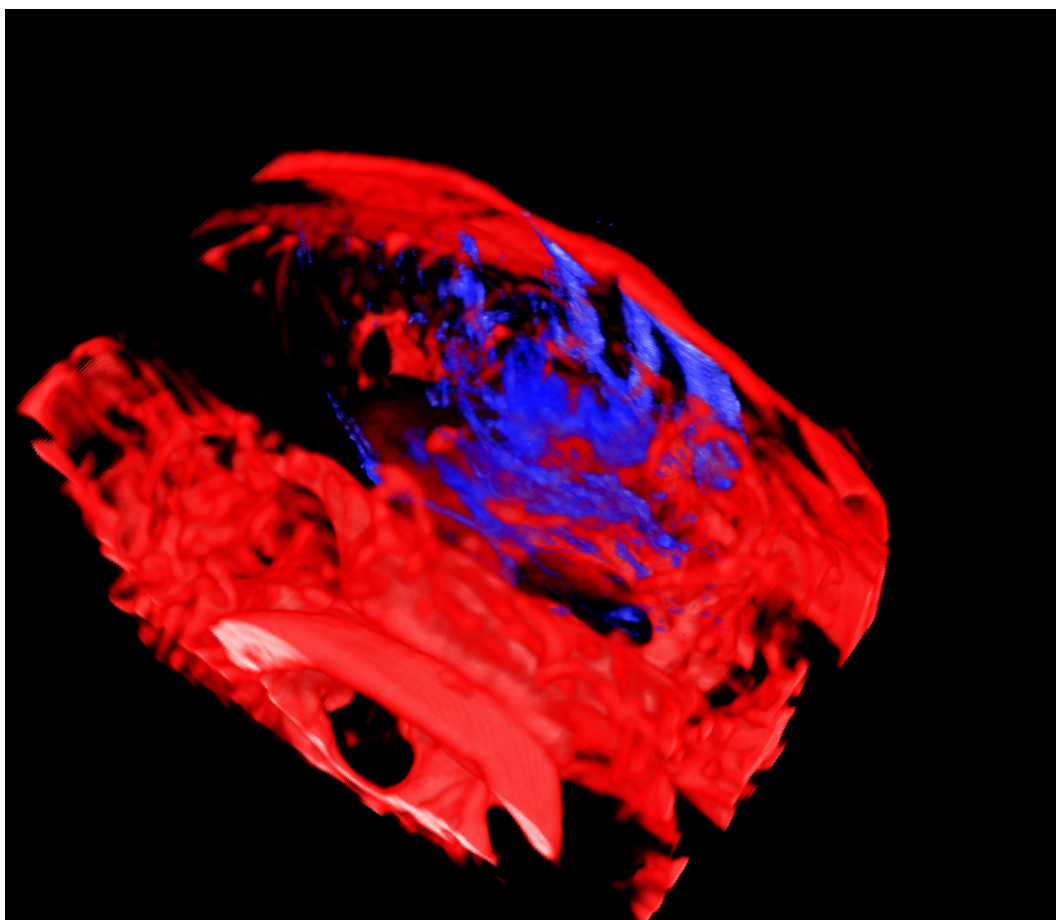


Figure 42: Tumor 1, phase 2. Volume rendering of MRI T2 (*red*) & UST (*blue*) before registration.

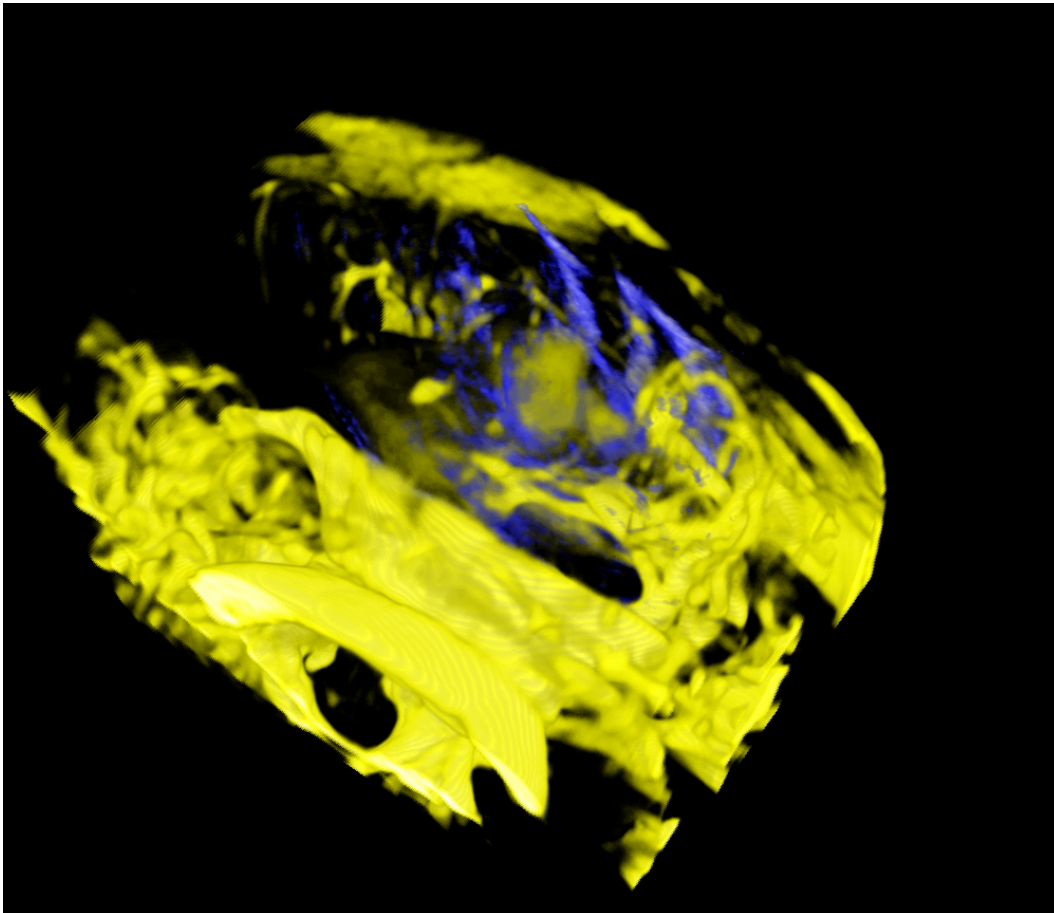


Figure 43: Results Tumor 1, phase 2. Volume renderings of UST (*blue*) and MRI T2 (*yellow*) after registration.

Parameter	Value
	# A.t2.2 NMI
Max no. of iterations	200
Samples	8000
Initial radius	1.0
Histogram bins	128
Gradient smoothing	0.5

Table 10: Parameters used for registration of Tumor1 MRI T2 and UST#1, with results seen in figure 43.

Parameter	Value
	A.t2.2 NMI
X translation	-0.305337
Y translation	1.19892
Z translation	11.4751
MI metric value	1.02012
No. of iterations used	200.

Table 11: Transform resulting from registration of MRI T2 and UST #1, as seen in figure 43.

Also the `RegApp2b` has been used on this dataset, to register the T2 and the first UST. The result of this can be seen in figures 44 and 45 (before) and figure 46 (after). The arrows attempt to mark the middle top of the tumor, in the T2 before registration (grey) and in the US (yellow). Note how this point of the tumor in the T2 shifts after registration, in figure 46. The parameters used for this registration is included in table 12, with the output — including rotations, given as a versor — in table 13.

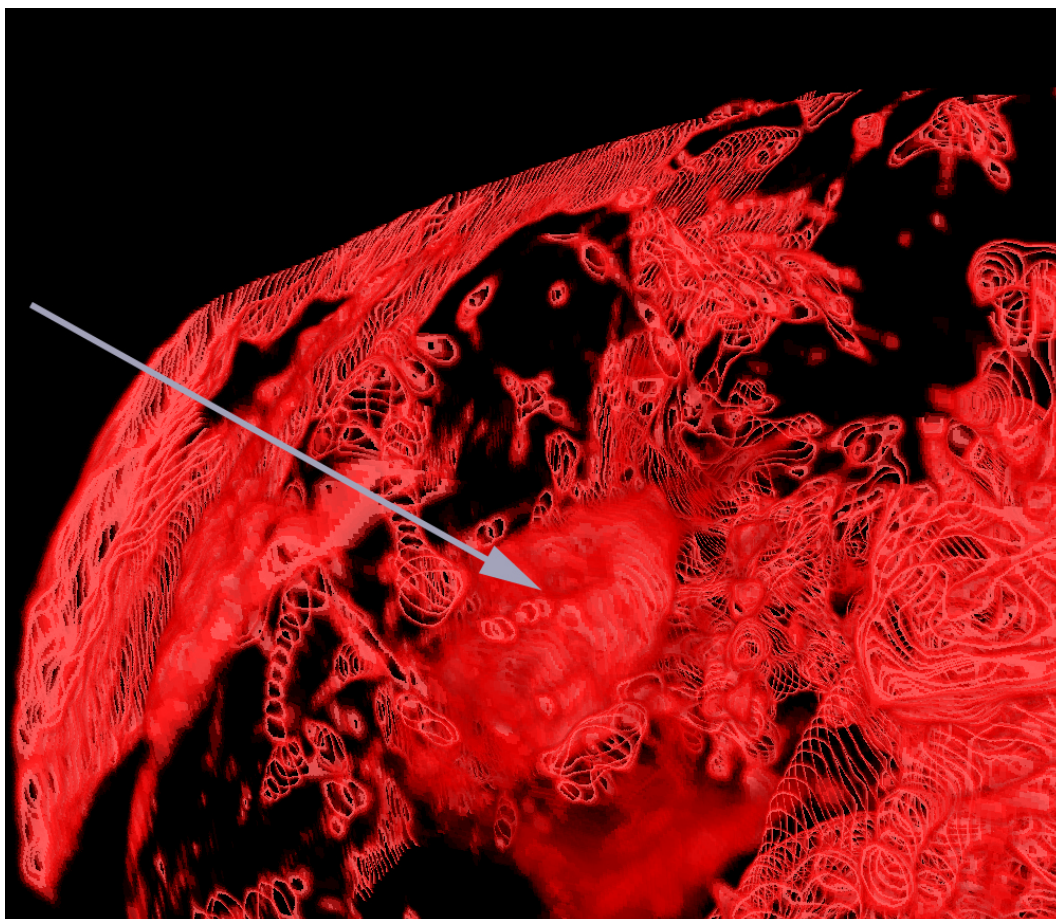


Figure 44: Tumor 1, phase 2. Volume rendering of MRI T2 (red) before registration. Arrow suggests mid-top of tumor.

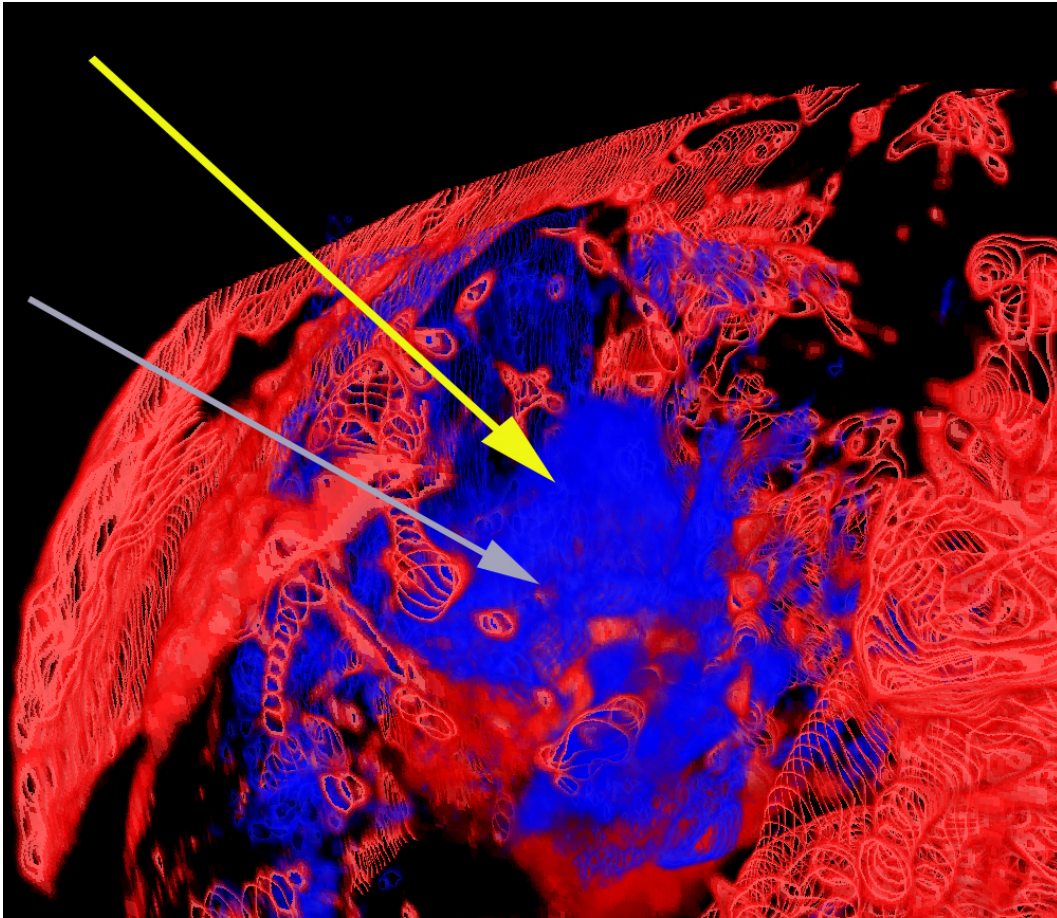


Figure 45: Tumor 1, phase 2. Volume rendering of MRI T2 (*red*) and UST (*blue*) before registration. Arrows suggest mid-top of tumor; (grey in T2; yellow in US)

Parameter	Value
	# B_woGM_rev_3 MI
Max no. of iterations	1000
Samples	20000
Max. step length	5.0
Min. step length	0.5
Histogram bins	70
Gradient smoothing	N/A

Table 12: Parameters used for registration of Tumor1 MRI T2 and UST#1, with results seen in figure 46.

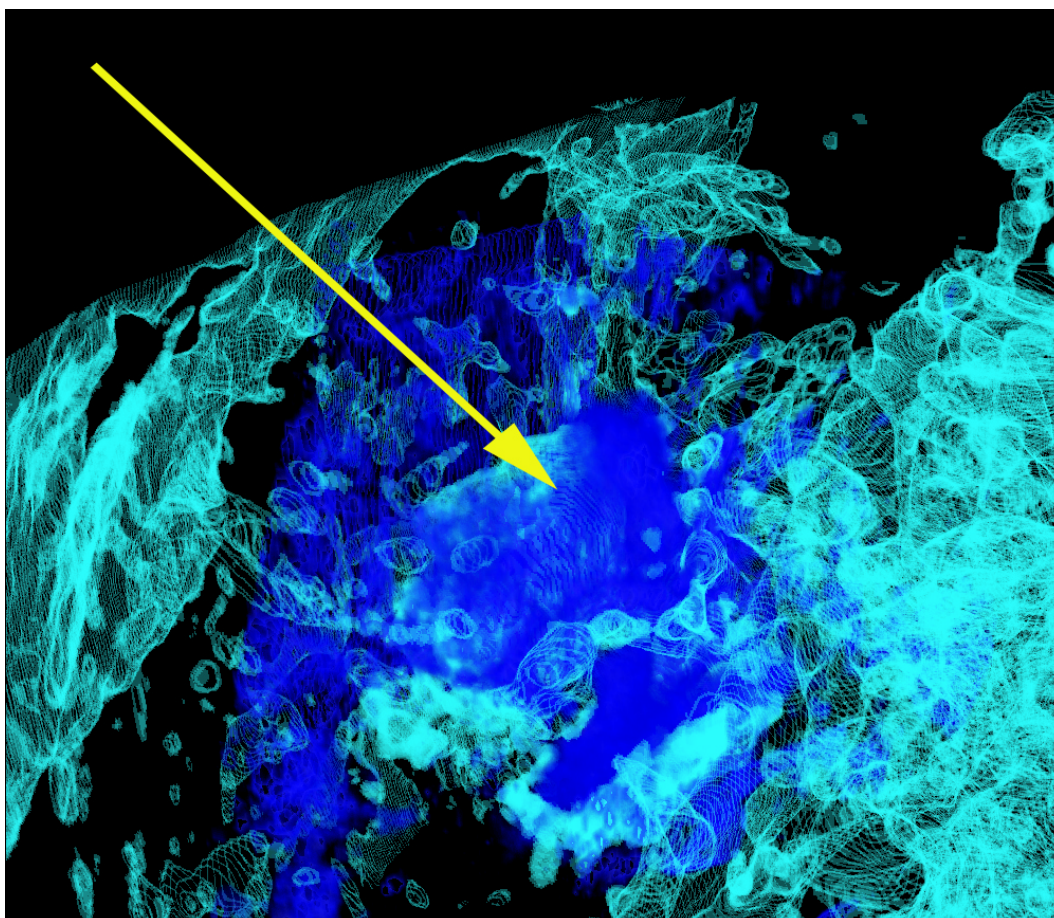


Figure 46: Results Tumor 1, phase 2. Volume renderings of UST (*blue*) and MRI T2 (*cyan*) after registration. Arrow suggests mid-top of tumor, as in figure 45.

Parameter	Value
	B_woGM_rev_3 MI
<i>X</i> translation	-2.05412
<i>Y</i> translation	-2.64718
<i>Z</i> translation	-8.65504
<i>versor</i> rotation	(0.0235116,-0.07168,-0.0068904)
<i>MI</i> metric value	-0.193936
No. of iterations used	(7, 4, 25, 83)

Table 13: Transform resulting from registration of MRI T2 and UST #1, as seen in figure 46.

13.1.2 MR-US Angio

The other registration in this phase on data from the Tumor 1 data set, involves the MRA and USA volumes. These can be seen in a volume rendering together in figure 47, and in a more selective view in figure 48, with the offset quite notable. Using `RegApp2b`, with the parameters repeated in table 14, the MRA has been registered to the USA, in execution number 9995. The result of this registration is seen in figure 49. Note the alignment of the main vessel that the two images share, and the apparent remaining offset of the vessel “loop” at the right hand side. The resulting transform is included in table 15.

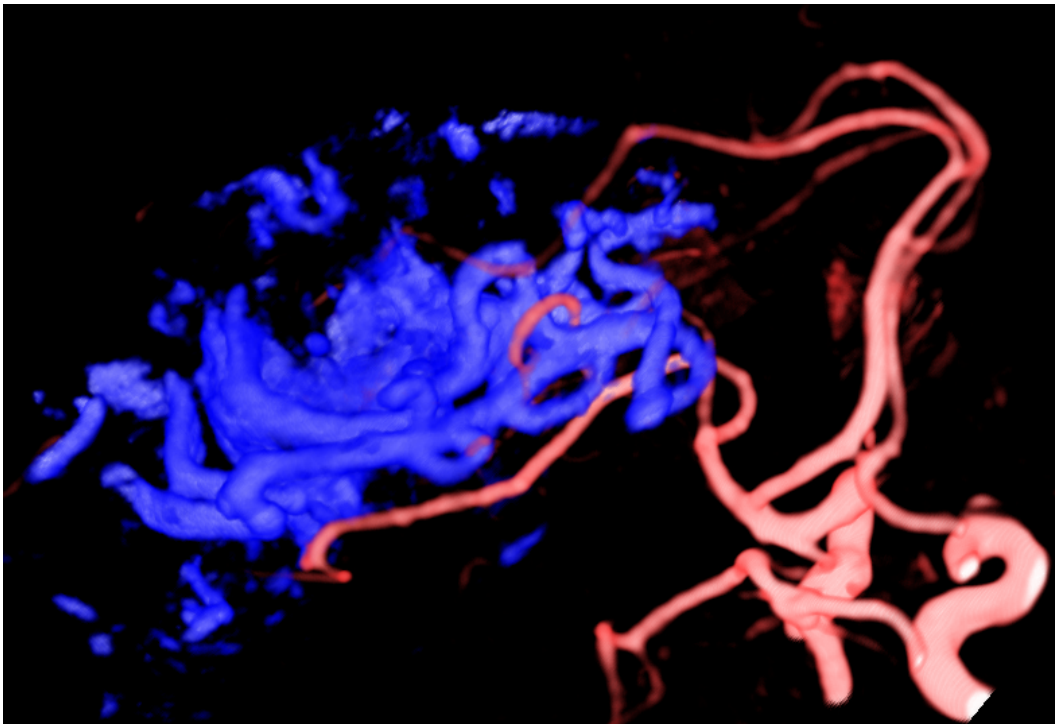


Figure 47: Tumor 1, phase 2. Volume rendering of MRA (*red*) and USA (*blue*) before registration.

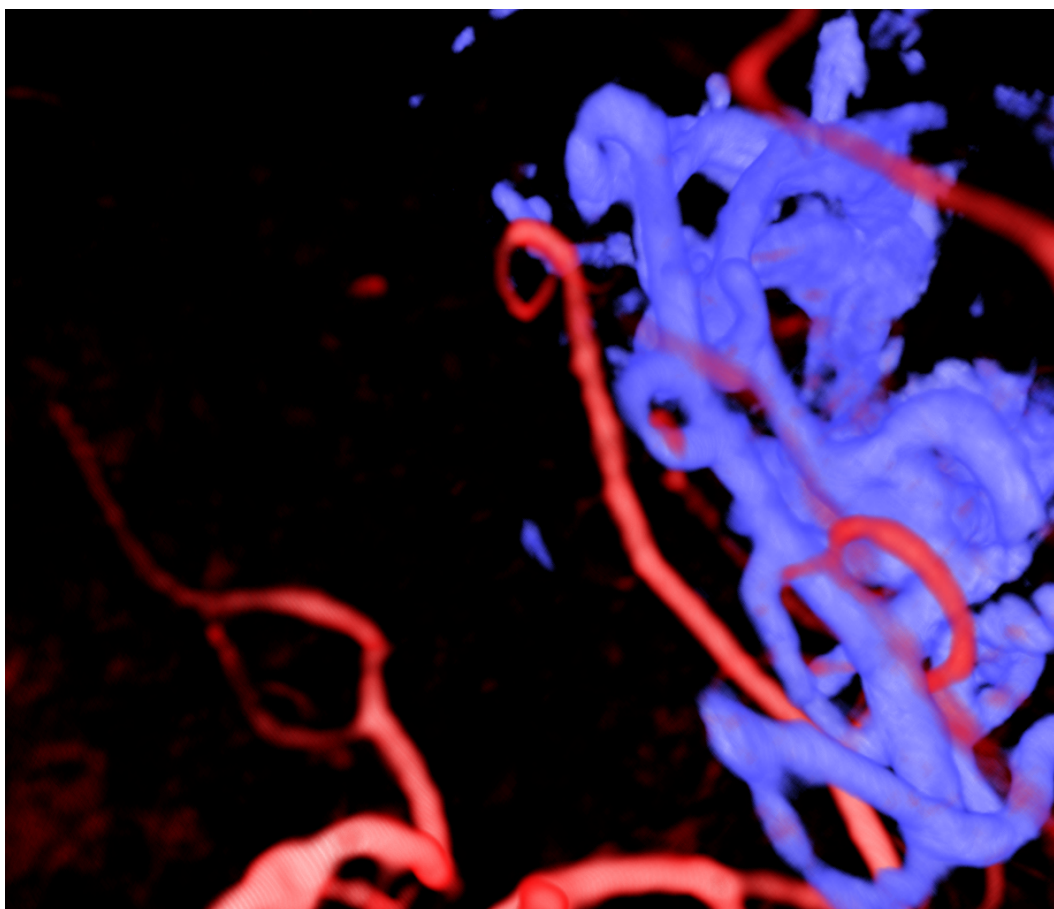


Figure 48: Tumor 1, phase 2. Volume rendering of most interesting part of MRA (*red*) and USA (*blue*) before registration.

Parameter	Value
	# B_woGM_rev_3 MI
Max no. of iterations	1000
Samples	80000
Max. step length	5.0
Min. step length	0.01
Histogram bins	128
Gradient smoothing fixed	0.5
Gradient smoothing moving	0.1

Table 14: Parameters used for registration of Tumor1 MRA and USA#1, with results seen in figure 49.

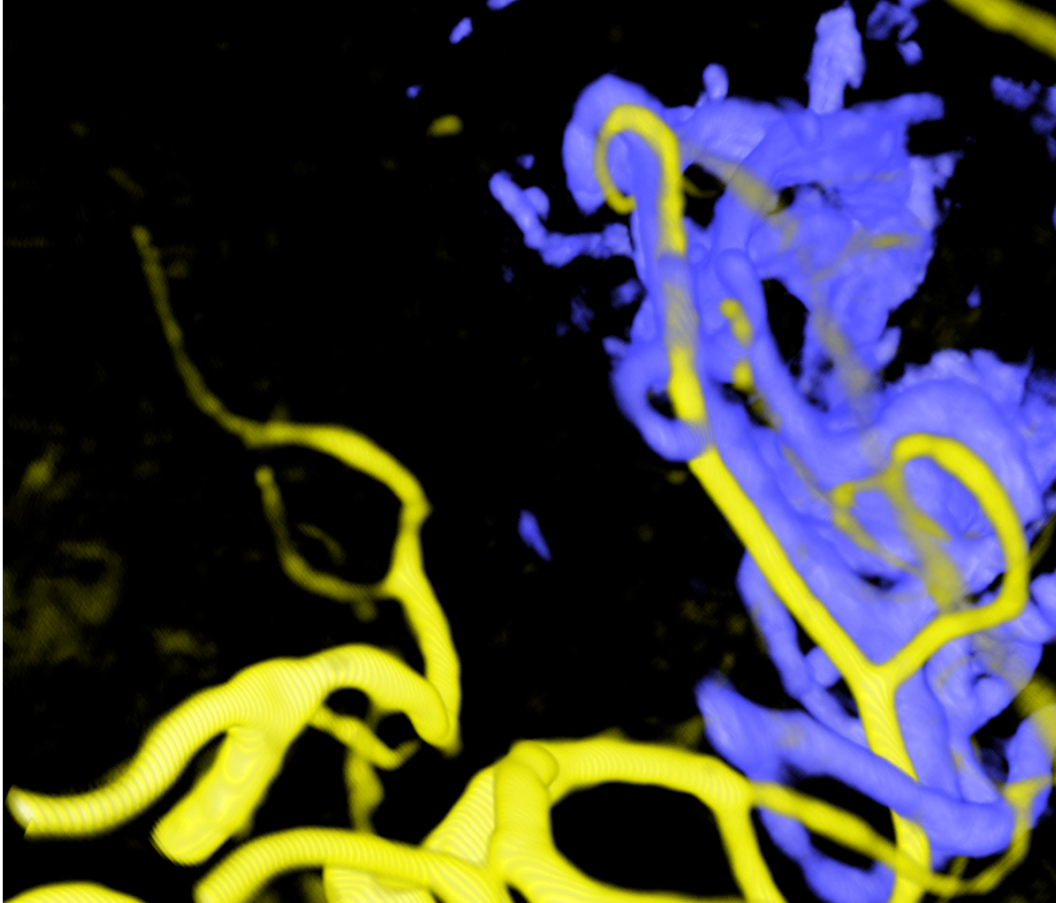


Figure 49: Results Tumor 1, phase 2. Volume renderings of USA (*blue*) and MRA (*yellow*) after registration.

Parameter	Value
	99995 MI
<i>X</i> translation	1.19466
<i>Y</i> translation	-1.91114
<i>Z</i> translation	8.79746
<i>versor</i> rotation	(0.040806,-0.00131552,-0.072796)
<i>MI</i> metric value	-0.0188259
No. of iterations used	(23, 24, 291, 55)

Table 15: Transform resulting from registration of MRA and USA #1, as seen in figure 49.

13.2 Aneurism 1

The Aneurism 1 dataset only has one registration operation in the second phase, between the MRA and the first US Angio, as seen in figure 50. This registration has been attempted using both the NMI-based registration procedure, and the multi-resolution MI-based registration application. In figure 52, the volumes can be seen registered with the latter method, in registration run #2b_9 with the parameters included in table 16. Only translations were applied in this run, and the resulting transformation parameters are those seen in table 17. In the same table is also the transform resulting from the NMI-based run #reg2a_i100_r2, using the parameters also seen in table in 17. The volumes after this registration are visible in a 3D rendering in figure 51. These results are also used in an article[10] submitted to the *International Journal of Medical Robotics and Computer Assisted Surgery*.

Parameter	Value	
	# i100_r2 NMI	# 2b_9 MI
Max no. of iterations	100	200
Samples	5000	8000
Max. step length	–	1.0
Initial radius	2.0	–
Min. step length	–	0.01
Histogram bins	80	30
Gradient smoothing	0.1	0.5

Table 16: Parameters used for results seen in figure 51 and 52.

Parameter	Value	
	# i100_r2	# 2b_9
X translation	-4.36111	-4.40235
Y translation	1.07498	1.42716
Z translation	1.57767	2.203
MI metric value	1.00166	-0.0170659
No. of iterations used	100.	28.

Table 17: Transform resulting from registration of MRA and USA #1, as seen in figure 51 and 52.

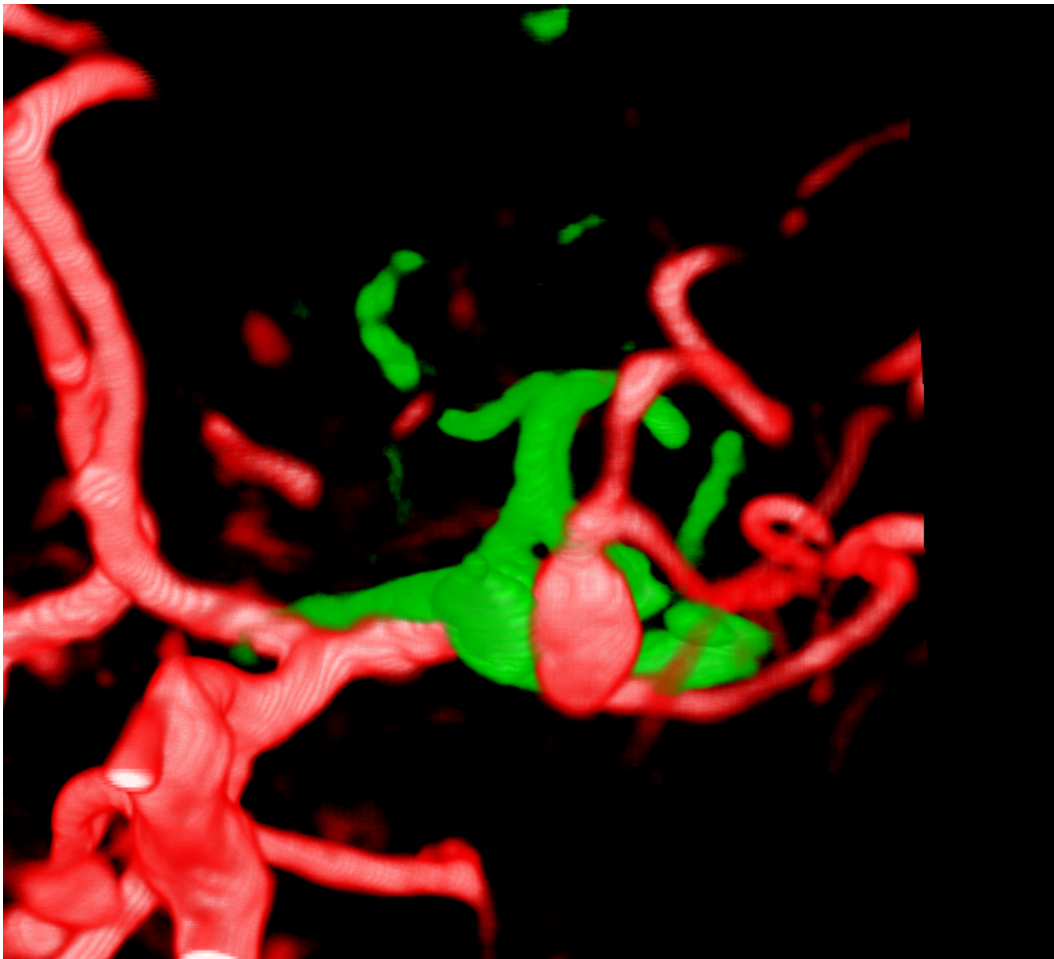


Figure 50: Results Aneurism 1. Volume renderings of MRA (*red*) and USA (*green*) before registration.

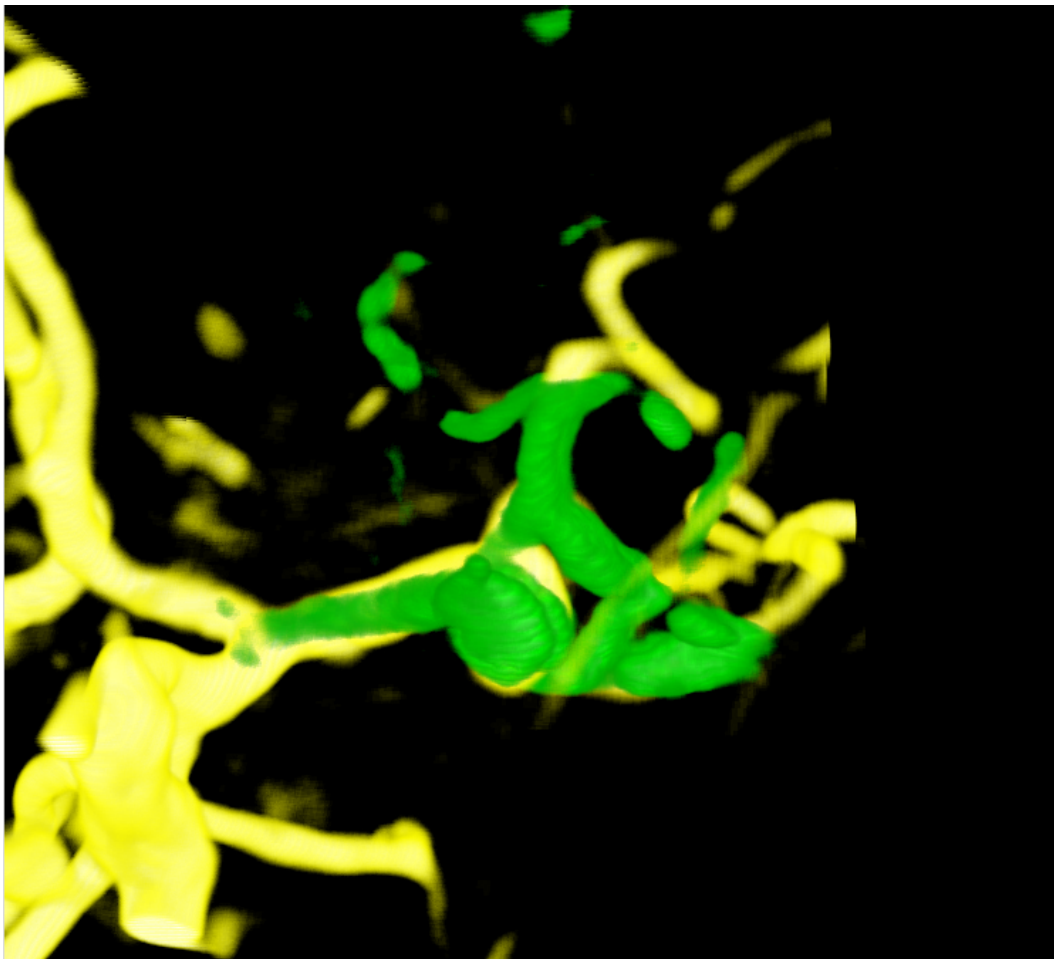


Figure 51: Results Aneurism 1. Volume renderings of USA (*green*) and MRA (*yellow*) after registration.

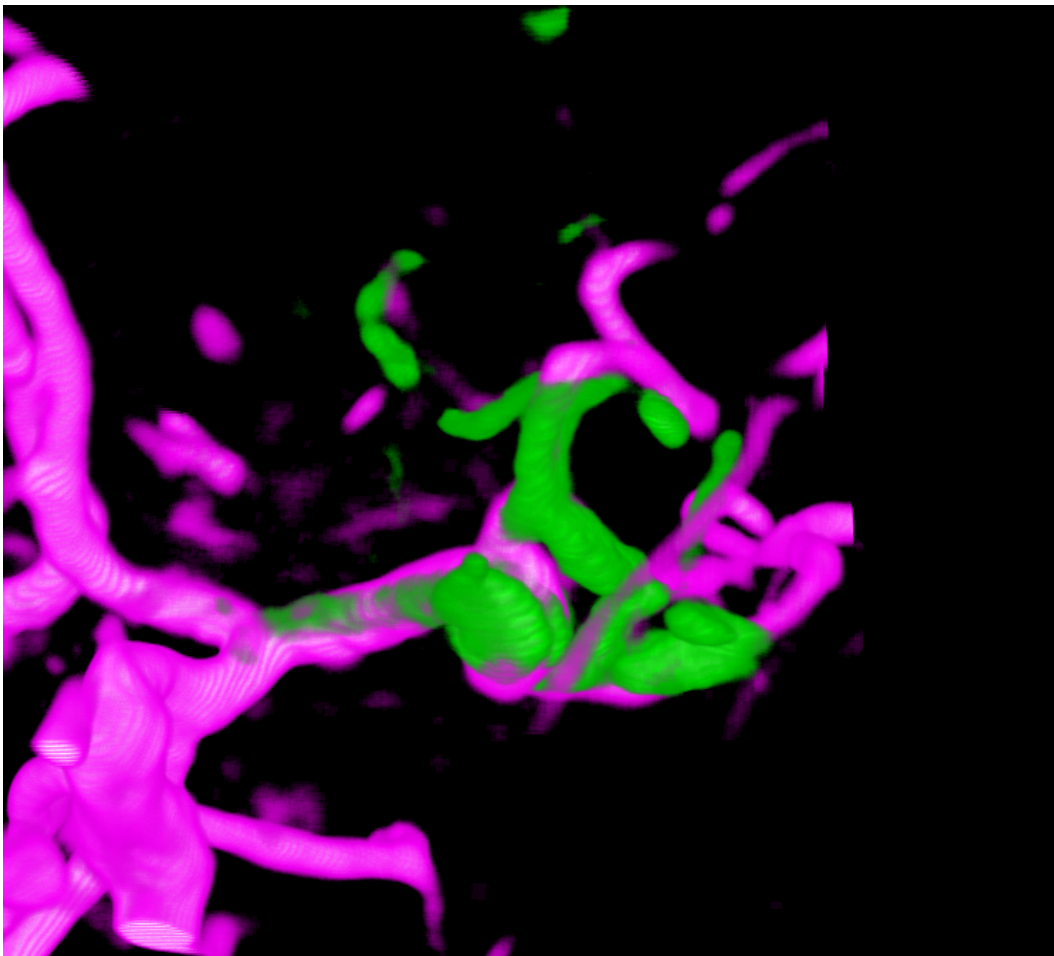


Figure 52: Results Aneurism 1. Volume renderings of USA (*green*) and MRA (*magenta*) after registration.

14 3DUS-to-3DUS registration results

Unfortunately, the `RegApp3`, for non-rigid registration of the ultrasound volumes was not completed in time to warrant its use for producing reliable results, with a sufficient degree of confidence. This was mainly due to unforeseen problems in the operation of the phase 2 registration applications `RegApp2` and `RegApp2b`, further discussed in part V.

Part V

Interpretation and discussion of results

The results presented in the prior part are examples of, more or less, successful registrations. However, the different applications differed in their performance, and especially regarding stability and reliability. These issues, along with a discussion on how to measure success in the performed registrations, is covered in the following part. This will lead up to the thoughts on improvements and other necessary future work, and a preliminary conclusion, in part VI.

15 Stability and reliability discussed

As indicated in chapters 12, 13 and 14, the registration applications did not all perform perfectly. While the phase 3 application for US-US-registration was not completed in time to have results included, the two second phase applications experienced some operational problems. Despite this, they were able to produce the results presented in the previous chapters. The nature of these problems, and possible reasons for them, is discussed in further detail in this chapter — with each application handled separately. This includes considering the accuracy, repeatability and reliability of the applications. Performance, in terms of time of computation, processing and memory requirements, will be handled in chapter 16, and the challenge of determining successful registrations is postponed to chapter 17. The correctness and accuracy of the achieved registrations is also covered in the latter chapter.

15.1 MRI-MRI registration

The `RegApp1` is the simplest of the four programs, and it is therefore not a surprise that it works with less problems than the more complex ones. In part, this is probably due to the application's use of standardized methods, but the task it performs is likely to be an equally important factor. Registration of the different MRI volumes, which in most cases only have a small translational and no rotational offset, and which are of comparable physical extent and visual quality, is a much more gratifying task than the second phase registration of MRI to US volumes. The phase-1 application parameters can be set so the program makes only smaller translations, thus in effect reducing the search space dramatically. Compared to the `RegApp2b`, the `RegApp1` also has a reduced number of degrees of freedom, which further eases the computational burden.

We are assuming for now that the results presented in the previous part represent sufficiently good registrations, as this will be covered as a separate topic in chapter 17. The transformations resulting from the MRI-MRI registrations correspond to translations of at most 2 millimeters, with most being less than 0.5mm. This is equal to sub-voxel size translations, and as no higher accuracy than half-voxel-size can be expected from the application, this indicates that the volumes were indeed very close to perfectly aligned also before this registration. For most of the performed registrations, the volumes show a near perfect alignment along the X and Y axes, while almost all volume pairs gave a Z -axis shift of ~ 1 mm or more. This may indicate that during the MRI acquisition, or upon import of the DICOM data sets in the navigational system, there is an inherent inaccuracy in Z -axis calibration. However, using `RegApp1`, this shift can be accounted for, and the MRI volumes brought into alignment.

Results seen in chapter 12 indicate that a uniform set of parameters is likely to give consistent and reliable results. The number of iterations should have little influence on the performance, as long as it is set above the usually used 15–25. The Mutual Information metric seems to require the number of sample points to be set to at least 5000; little performance is lost, and some accuracy may be gained, by setting it usually to 8000 points. The maximal initial step length used to obtain the shown results varies between 1.0 and 5.0. Taking into account that the resulting translations seldom exceed 1mm, it should be sufficient to use a maximal step length of 1.0 in most cases. The minimal step length, as long as a mono-resolution approach is used, should be set sufficiently low to give the optimizer time to ensure convergence, and a sufficient accuracy. A value of 0.001 seems to be a useful compromise between computation time and convergence/accuracy.

Both for the ordinary tissue modalities, and the special angiographic volumes, the application seems to perform equally well. It is also apparent that the `RegApp1` works quite well on both the US- and MRI-mastered volumes. The application additionally gives quite consistent results; subsequent executions of the program with the same input data and parameters will give results that differ only modestly. This is further evidence to suggest that the application indeed works as intended, and usually converges to give the optimal transform.

15.2 MRI-US registration — NMI implementation

While the `RegApp1` seems to perform well at its intended operation, the operation of `RegApp2` has not been quite as easy to control. The reliability of its results seems to be somewhat mixed, as well as its ability to return consistent results from repeated executions. Due to the stochastic nature of both the point sampling function and the evolutionary optimizer, a certain degree of randomness is to be expected in the application. However, these random events should ideally be canceled out over the many iterations of the optimizer. The histograms based on the sampled points is not likely, on average, to be biased enough to distort the metric substantially. The optimizer's derivation of the "child" parameters as stochastic samples from a multinormal distribution centered on the current transformation parameters, should neither give widely differing results — only *better* children are kept, and unless the normal distribution from which the children are sampled is distorted, the randomness of each iteration should be largely canceled out over the iterations.

When this does not fully happen, this is probably in part due to lack of convergence when the algorithm is not given sufficient iterations. With more iterations, the program seems to give more consistent results, although more iterations also seems to increase the number of unsuccessful registrations. This latter effect probably occurs because more iterations increase the chances of the optimizer finding a far-away, yet incorrect, set of transformation parameters that gives a very good NMI result. The evolutionary optimizer operates differently from other optimizers, in that each iteration does not necessarily result in a new transform estimate, unless the candidate transform gives a better NMI value. This means that it is not apparent from the results whether a particular registration has converged, or is merely temporarily stuck, even when long successions of identical transform parameters occur. In the following, it is assumed that the results presented in chapter 13 are correct enough, as this is covered separately in chapter 17.

To be able to determine when a good enough transform has been found, the minimal bound on the norm of the covariance matrix, dubbed ϵ , is used. The registration is terminated when the norm gets below ϵ , as well as when the maximal number of iterations have been exhausted. The norm of the covariance matrix decreases as the values in the matrix decrease, as they are multiplied with the shrink factor each time the candidate transformed did not return a better value than the parent. Thus, a certain number of "misses" in the optimization will bring the norm below the limit. However, in none of the registration attempts performed the norm reached the ϵ , and the allotted iterations were exhausted before the covariance matrix' values decreased enough to pass the ϵ limit. This indicates that reaching $\epsilon = 1.5e - 4$ will require an even higher number of iterations than the 100-500 limit usually used with this application. This might substantially increase the running time of the application, but should be weighed against the added benefit of being able to tell when a registration reaches sufficient convergence.

As for the remaining parameters, it seems that 5000 to 10000 sample points, an ini-

tial radius of 1.0 to 2.0 and between 60 and 130 histogram bins give overall good results. As is expected, more sample points give generally better solutions, but increases the running time. Setting the initial radius much higher than 2.0 is likely to drive the algorithm towards too-far-away, and incorrect solutions. The number of histogram bins does not affect the algorithm much, as long as it is kept within reasonable bounds; experience shows this might be above 40 and below 150. Using these parameters, and increasing the number of iterations, should reduce the problems with inconsistent results. The problems regarding inconsistent and incorrect results diminish further when angio-specific images are used, as especially witnessed in data set Aneurism 1. Also, when only the region of the MRI that is present in the US is used (i.e. using “US-mastered” volumes) in the registration, the results are more reliable, and in general better, such as those seen in chapter 13. The accuracy of those examples are discussed further in chapter 17.

The `RegApp2` application has limitations as to the volumes it manages to register correctly, as well as how accurate transformations it may represent, all the while only translations are used. Regardless, ensuring that only US-mastered volumes (or equivalent) are used, and using the parameter ranges specified above, the program should in most cases be able to give good results within its three degrees of freedom.

15.3 MRI-US registration — MI implementation

The `RegApp2b` registration application has been more erratic in its behaviour than the NMI-based implementation, and while some decent results have been obtained, this program has serious shortcomings that will need to be addressed before it may be used in real-life applications. Each of the identified problems will in the following be presented and discussed, including whether they result from errors in this particular implementation, problems with the underlying methods or from a combination of erratic methods, incorrect use and/or unsuitable input parameters. As above, the presented results in chapter 13 will be treated as sufficiently accurate, and a discussion of their correctness is postponed to chapter 17.

15.3.1 Consistency, reliability and determinism

As mentioned in the previous chapter, the `RegApp2` sometimes returned differing results in consecutive runs with the same parameter. However, with `RegApp2b`, this seems to be more of a rule than an exception. While some results that seem fair have been obtained, especially regarding the Aneurism1 data set in chapter 13, the results of the full-fledged six-degrees-of-freedom MI-based registration application are quite unreliable.

In table 18, the results of running `RegApp2b` repeatedly with the same input volumes and parameters, can be seen. The parameters used are included in table 19. It is evident from the computed averages and standard deviations that the resulting transformation parameters from each execution are not reliably estimated; the standard deviation for most the parameters is almost as large as, or larger than, the average. Neither of the runs used more than 40 iterations at the last resolution level, and generally much fewer at the first levels, while up to 1000 iterations were allowed. This indicates that the registrations were forced to converge very rapidly.

The number of samples used in this example session was set as high as 500 000, to ensure at least that the histogram estimations had sufficient data to be statistically

a

No.	Iterations	X	Y	Z	r_1	r_2	r_3	-MI value
1	7	1.54122	3.25636	3.56452	-0.0782791	0.0555749	-0.0191095	-0.0100643
2	13	0.935737	-4.31799	15.2548	-0.163771	0.0161986	-0.130156	-0.0181744
3	27	4.76635	1.06432	5.59793	-0.0287298	0.0685305	-0.0426385	-0.0109351
4	32	0.473696	-1.19881	17.5186	-0.10608	-0.0136972	-0.107754	-0.0175057
5	8	3.64281	-1.14786	11.1396	0.0229164	0.0210948	-0.0844113	-0.0126984
6	24	5.56744	0.169778	11.4879	-0.0239955	0.0244249	0.0064195	-0.0116114
7	40	6.2011	0.96338	10.9348	-0.0775538	-0.0106923	0.0281685	-0.0104018
8	11	-0.917293	-7.06778	17.0972	-0.0228333	-0.0187335	-0.132797	-0.019275
9	15	-0.385941	-0.0316593	7.47014	0.0926192	-0.0309531	-0.129716	-0.00962599
10	14	6.42151	1.65114	10.8825	-0.0275165	0.00269991	-0.00146746	-0.0098675
11	21	4.39046	-1.98076	6.50929	-0.0584041	0.0222585	-0.050142	-0.0105228
12	8	4.34196	-3.39452	9.18021	-0.0142309	0.000765856	-0.0363754	-0.0116849
13	14	0.39795	0.525459	3.31664	-0.0110344	0.0429	-0.0421764	-0.00984099
14	23	4.00376	-1.06362	4.14043	-0.0838856	0.108121	-0.0231795	-0.0103474
Average:		2.96	-0.898	9.58	0.0415	0.0206	-0.0547	
Std.dev.:		2.437	2.568	4.600	0.05897	0.03663	0.005174	

Table 18: The resulting transform of successive executions of RegApp2b, using the same parameters and inputs.

Parameter	Value
Dataset	Tumor 1
Fixed image (MR)	MRA
Moving image (US)	USA
Max no. of iterations	1000
Samples	500000
Max. step length	6.0
Min. step length	0.1
Histogram bins	128
Gradient smoothing (fixed)	0.5
Gradient smoothing (moving)	0.1
Resolution levels	4

Table 19: Parameters used for reliability test reported in table 18

sound. A number of histogram bins of 128 should also be unproblematic; this parameter is considered to have little impact on the calculations, as long as it is kept above a reasonable limit of about 30. The step length limits of 6.0 and 0.1 are more arbitrary, but an initial step length of 6 seemed in retrospect rational for this operation, as seemingly good registrations included Z-axis translations of about 10mm. Lowering the lower step length limit below 0.1 will indeed increase the number of iterations used at each level. However, these limits apply only to the first resolution level, and the limits for the subsequent levels are calculated from the first lower boundary. Setting the lower limit too low will principally increase the number of iterations used at the first level, and most of the additional iterations will be used on transformations that are well below the voxel size at that resolution level. In addition, the maximum and minimum step lengths for the subsequent levels will also all be lower than they should. For the parameters in table 19, this set of step length limits for the different levels will result:

Level	Max.	Min.
1	6.0	0.1
2	0.2	0.05
3	0.1	0.025
4	0.05	0.0125

Optimally, at each level, a quite exhaustive search should be performed within the limits of that level's resolution. This implies that the upper limit should allow for as large corrections as is likely at that level. Correspondingly, the lower limit should be small enough to give the optimizer time to converge, yet large enough to ensure that the optimizer's correctional steps do not become insignificant compared to the voxel size. The scheme used for calculating the step length limits at each level follows guidelines in the ITK, and should be reasonable. However, even when this is taken into consideration, the application is seemingly unable to ensure that the feature space is searched thoroughly enough to grant a reliable convergence.

The current combination of problem set, similarity measure and transform does apparently not work very well with the gradient descent-derived optimizer. Even if the particular optimizer used has been designed to work specifically with the `itk::VersorRigid3DTransform`, it may be that a change of optimizer would increase the stability and reliability of the registration procedure. On the other hand, it may

also be that this transform/optimizer pair would benefit from a different preprocessing of the images, other than the gradient magnitude filtering. If the optimizer were to be replaced, the results indicate that a more complex one is needed. Perhaps the evolutionary 1+1 optimizer after all can be used also to find the 6 transformation parameters of the versor rigid transform? Whether this should be implemented by replacing the transform in `RegApp2` or the optimizer in `RegApp2b`, remains an open question. Other more exotic approaches to the optimization problem might also be considered. One might perceive an optimizer inspired by a simulated annealing scheme, and akin to the evolutionary 1+1, where each iteration's parameters are drawn from a probability distribution centered on the current parameters. Differing from the 1+1 evo, however, the result will not only be accepted when it gives a better MI score than the current, but with a certain probability also when it scores lower. In following the simulated annealing approach, the probability of accepting a lower-scoring parameter vector declines successively by each iteration. The rationale behind accepting also parameters that score lower than the current ones, is that the feature space landscape may have local optimums where an optimizer tends to get trapped, unless it is allowed to take one step in the "wrong" direction, which will give it a higher chance of reaching the true optimum. For the application considered here, such an approach would have to be carefully considered, as it is far from certain that the correct registration corresponds to the *global* maximum of the Mutual Information. Ensuring that smaller false, local optimums are surpassed, while the *true*, but still local optimum, is not, will be the challenge of implementing such an optimization scheme for this program.

As mentioned, the gradient magnitude filtering may also be partly to blame for the problems. An informal test of the `RegApp2b`, with the gradient magnitude (GM) filtering disabled seemed indeed to give more consistent results; at least subsequent executions using the same parameters gave less divergent results than those in table 18. Whether these results were more or less correct than those obtained using GM, has not been investigated. Nor is it very interesting, as long as the application with GM gives such varying answers — to which should the non-GM version be compared? One possible way to improve the application is to study in further detail different manners in which an edge-map of the images may be obtained. One problem with the GM filtering is that it does not only enhance edges that are due to tissue boundaries. Noise-induced edges and other high-frequency artifacts also become more pronounced, and this is likely to lead to a less smooth parameter space for the optimizer to search through. To alleviate these problems other filtering techniques should be considered. Particularly, using an anisotropic diffusion filter might be useful, which would even out noise in areas of the image with relatively uniform intensity values, while preserving edge (tissue boundary) information. Combining this with the gradient filter could possibly give the optimizer an easier job, and thereby increasing the likelihood of reliable convergence.

Another possible solution, which is much less intrusive to the application, is to make sure that the initial translation parameters are set close to the actual offset. Then, the step length parameters may be set much more strict, as no long steps will have to be considered, and more stringent boundary conditions may be applied. This approach is in any case a natural part of an integration process between the registration application and any surgical visualization/navigation tool, such as CustusX. Both technicians at Sintef MedTech and clinicians at St Olavs Hospital in Trondheim, have indicated to me that an easy-to-use graphical user interface for rough alignment, to initialize a registration algorithm, is tolerable, and even desirable. This initial alignment might be found either by approximate indication of one or more homologous points in the volumes to be registered, or by moving the "moving volume" freely around in a 3D scene until it approximately matches the "fixed volume". Both these

approaches should be feasible within the framework of CustusX. Further investigations on the step length and resolution level parameters that might work optimally with this approach is needed.

15.3.2 Tissue and angio images

In addition to the characteristics and experiences of `RegApp2b`, it also seems that this application handle angiographic volume-registration better than that of ordinary tissue data. However, the unreliable example results in table 18 are gathered from MRI and US angio registration. The reason for the subjective observation that angio-volumes were better registered, might thus be that correct MRA/USA registrations are easier to discern, than ordinary MR/US tissue registrations. This is covered in further detail in chapter 17.

15.3.3 Varieties of the program

As is indicated both above and in the presentation of results in chapter 13, several slight variations of the `RegApp2b` application have been used. The most important changes, that also have produced identifiable results, include using only a translational transform, removing the gradient magnitude filtering and reversing the fixed- and moving images (while inverting the transform). One of the varieties of the application, without the gradient magnitude filter, has already been discussed above.

The best, and most reliable results that have been obtained with any version of the program are those of the MRA/USA registration in data set Aneurism 1. This was done using a version of the program that only used a translational transform, and as such had only half the number of degrees of freedom as the original program. This is likely to have eased the burden for the optimizer, which was also replaced with the ordinary regular step gradient descent optimizer used in `RegApp1`.

In another version of the program that has been attempted, the fixed and moving images were switched. The MRI would typically be used as “moving”, and the US as “fixed”. This change also included inverting the transform, compared to the original program where the transform was applied to the fixed (MRI) volume. The reason for doing this was to ensure that as much information from the smaller US volume as possible, would be used in calculating the MI metric. The original version often seemed to drive the translational parameters far away in the early resolution levels, just to align generally the volumes’ centers of mass, instead of their features. This change did in fact decrease the occurrence of this problem, and was kept in the final version.

15.3.4 Other observations

The problems described here, especially regarding the repeatability and stability of the `RegApp2b` application, made it near impossible to find well working parameters for this program in the same manner as was done for `RegApp2`. The parameters that produced the most promising results however, seem to be in line with the optimal parameter ranges found for the NMI-based program. The most prominent difference between the two programs is in the optimizer and in the multi-resolution scheme used for the latter, and this is also where the most difficult parameter estimations were to be found. With an approximate initial translation, possibly a more fitting

preprocessing, and using the US as the “fixed” volume, it should be possible to use standardized parameters also for this application. This would typically include 5000 to 10000 sample points, around 60 histogram bins, 4 resolution levels, up to 1000 iterations, and a maximum and minimum step length of 2mm and 0.1mm, respectively.

16 Performance discussed

Volume-to-volume registration is a computationally intensive task, requiring multiple iterations with calculation of the similarity metric, transformations and interpolation of voxel values. This means that the developed registration applications have a notable running time. The volumes used are also of significant proportions; with a resolution of 1mm/voxel or less, most volumes span around 150-300 voxels in each dimension.

16.1 MRI-MRI registration

The `RegApp1` has a more limited time of execution, as it only uses one resolution level. This relieves it from having to resample the volume, which can take a considerable amount of time, and memory resources. However, using only the original volume makes each iteration more time-consuming. Applying the transformation, and using the linear interpolation to derive new intensity values voxel-by-voxel, is done on a large volume. Most executions of this application used less than 5 minutes to complete, on an Apple PowerMac G5 running at 2GHz¹² with 2GB RAM. These times include the reading in of the fixed and moving images, and the writing out of the transformed, moving image.

16.2 MRI-US registration

The applications for the second registration phase differ in methods and algorithms, but only slightly in performance. The NMI-based implementation, `RegApp2`, using only one resolution level, generally uses fewer iterations in total, but spends longer time performing the calculations at each iteration. Mostly, this is due to the increased complexity of the optimizer. Compared with `RegApp1`, both `RegApp2` and `RegApp2b` use considerably longer time to set up the registration framework before starting the iterations. A significant portion of this time is spent calculating the gradient magnitude images. In the case of `RegApp2b`, some time is also spent building up the multi-resolution image pyramid.

Using the same computer as above, `RegApp2` usually completed the registration in 15–30 minutes, depending on the number of iterations. Using maximum 200 iterations, 10–15 minutes was normally used, while doubling the number of iterations also nearly doubled the time spent.

Meanwhile, `RegApp2b` normally used 5–15 minutes to perform the registration. The number of spatial sample points influenced the running time only slightly, Increasing the number of sample points to 500000 from the usual 8000, gave a running time of close to 10 minutes, compared to the usual 4-6 minutes. The maximum and minimum step lengths also influence the computation time. Especially when the lower step length limit is set so low that the optimizer uses all the available iterations at all levels, can the execution times exceed 15 minutes. However, with the instability of the application, and its inability to repeat results reliably, the running times vary quite a bit. It is also uncertain whether these timings will be indicative of the execution times a revised, stable version of this application will require.

¹²A dual CPU PowerMac was used, but as the applications are only single-threaded, little performance will have been gained compared to a single-CPU setup.

16.3 Parallelism

All the developed applications are single-threaded, and run in a sequential order only. This means that any inherent parallelism in the algorithms is not exploited, and all computations are handled completely iteratively. However, it may be possible to get speed gains in all the applications by considering a more parallel implementation. The primary targets for parallelization should probably be the point sampler functions, as each sampled point in the moving image is not dependent upon any of the other voxels in the *moving* image. This parallelism extends also to the interpolator and transform used in the metrics to do the point sampling. Revising the MI metrics in ITK to perform $1 : n$ point transformations and interpolations in parallel, may speed up the computation of the metric value. On the other hand, quite a bit of overhead will be used on collecting the point samples and collating them into the basis for the metric calculations. These aspects would have to be weighed against each other, if a parallel version of the metrics were to be implemented.

17 How to measure success

Volume-to-volume registration is not an easy problem. Rather, it is under-constrained and ill-posed. This does not just make it difficult — as we have seen — to implement a well-behaving registration application, but also to validate the results obtained with such an application. The question is, how do one decide whether a particular registration was successful, or *how* close it came to the true registration? And analogously, how can a set of registration results be compared, and one decided to be the best? This is the problem of measuring success in registration methods, and especially volume-to-volume registrations.

It could be envisioned that some objective measure of likeness should be applied to the registered volumes, but this is just what has been done to *perform* the registration. Using the same similarity measure would produce the same result as the registration; the best match is that with the highest mutual information value, or equivalent. Using another measure, one might never get a final answer, as different metrics weigh different aspects of similarity unequally. An image pair that has a high mutual information value, might very well have a low correlation ratio, for instance. Then the question arises; if the correlation ratio is used to judge the registration results, why would it not be used as the similarity metric of the registration itself? Thus, some other method for measuring the success, detached from the registration program, has to be devised.

17.1 Use of “gold standard”

A common, and very accurate way of measuring the accuracy and correctness of registrations, is to use a known transform between the registered images as a “gold standard”. This transform could be obtained either by careful manual selection of homologous points in the two volumes, or by registering input images that have a known offset. The latter approach is not useful in this context, as the images that need to be registered really have an unknown transformational relationship.

Using homologous points selected manually from the volumes that were to be registered is considered the most accurate way of finding the transformation, and gives a good ground for comparison against the results obtained using some devised automatic registration application. If only translational offsets were to be considered, selecting more than three points would enable a least-squares fitting of the point pairs to a translation vector. In the case of rigid transforms, such as that used in `RegApp2b`, more than 6 points would be needed to obtain a reference-transform.

Unfortunately, no tools that are able to perform this kind of operation have been available at Sintef MedTech in the course of my work with this thesis. This is certainly one of the most imperative necessities for future work in volume-to-volume registration.

17.2 Experts’ statements

As there has been no objective, gold standard reference available, against which the results could be compared, another evaluation criterion would have to be used. Remember, the goal of the registrations covered by this thesis has been to facilitate the use of multiple imaging modalities in neurosurgical planning and guidance. This leads to one of the most relevant subjective evaluation method, which would be to

ask trained clinicians whether they thought the obtained results would be useful in clinical procedures.

In an experiment, two highly experienced neurosurgeons, as well as seven medical technology and imaging researchers and three computer science/imaging students, were asked to assess the suitability of the registration results presented in part IV. The particular questions, and questionnaire form used is included in appendix D. The number of results that were presented was limited, due to time constraints, and only the registrations seen in table 20 were included in the evaluation.

No.	Dataset	Fixed volume	Moving volume	Program used
1	Tumor 1	MRI T1/fMRI	MRI T2	RegApp1
2	Aneurism 1	MRI T1	MRA	RegApp1
3	Tumor 1	MRI T1/fMRI	UST #1	RegApp2
4	Tumor 1	MRI T1/fMRI	UST #1	RegApp2b
5	Tumor 1	MRA	USA #1	RegApp2b
6	Aneurism 1	MRA	USA #1	RegApp2

Table 20: Registrations used in the evaluation by clinicians and researchers.

The registered volumes were presented using *CustusX v.2.0 α* , first showing the reference volume (“fixed” in phase 1, ultrasound in phase 2), and then the registered volume (“moving” in phase 1, MRI in phase 2). Upon request, the original of the registered volume would also be shown. The scene displaying the volumes was further rotated, zoomed and shifted upon request from the participants. The results of the questionnaire are included in table 21.

No.	Overall			Clinicians			Researchers			Students		
	Not sufficient/null	Possibly sufficient	Sufficient	Not sufficient/null	Possibly sufficient	Sufficient	Not sufficient/null	Possibly sufficient	Sufficient	Not sufficient/null	Possibly sufficient	Sufficient
1	1	7	4	0	2	0	0	3	4	1	2	0
2	2	1	9	0	0	2	1	0	6	1	1	1
3	4	7	1	1	1	0	1	5	1	2	1	0
4	3	7	2	0	2	0	1	4	2	2	1	0
5	3	4	5	0	0	2	1	3	3	2	1	0
6	1	3	8	0	0	2	0	3	4	1	0	2

Table 21: Result of the evaluation by clinicians and researchers. Number of responses per registration no. and option.

It is evident that only two of the presented results were clearly sufficiently registered; number 2 and 6, corresponding to MRI T1 and MRA in phase 1, and MRA and USA #1 in phase 2, respectively. Both these registrations are from data set Aneurism 1. The other phase 2 angio registration, number 5, was also mostly regarded as good,

especially by the clinicians.

However, for the remaining results, the result must be said to be inconclusive. Informal inquiries showed that the reason for the uncertainty was mainly the way the data was presented. While the MRA and USA are easily portrayed with high visual quality in a 3D volume rendering scene, like the one CustusX 2 provides, the details of tissue volumes are often difficult to communicate in a similar scene. In the tissue modalities, important structural and anatomical information lies on the inside of other structures. Without the ability to intersect the volumes with cutting planes, it is near impossible to give a volume representation of these data that will enable even trained professionals to tell a successful registration from a mediocre registration. Ideally, the two volumes should have been rendered in a 3D scene, perpendicular planes inserted to cut out a part, and the cross section of the planes be placed across an important anatomical structure such as a tumor. This would allow both the general structure to be seen from the volume rendering, and the anatomical details to be easily identified in the cross planes. By moving the 2D planes up/down and left/right or forward/backwards, it would be quite easy to see whether tumor boundaries and other important structures are perfectly aligned along all axes.

In order to be able to assess the correctness of future registrations, the accuracy of current and future registration applications, *and* to safely use the registered volumes in a surgical setting, cross plane visualization capability will need to be implemented in CustusX 2. The lack of this ability in the available visualization tools, has severely limited the ability to verify and evaluate the results obtained in this thesis.

Regardless, based on the evaluation session with the clinicians and researchers, it is fair to say that the implemented registration methods have potential to become clinically worthy. This requires that repeatable and stable results of the same quality as those seen for the phase 1 registration and the phase 2 registration of MRA and USA volumes, can be obtained, and verified through cross plane visualization.

17.3 Other measures of success

A last method for performing registration evaluation, used at least by Letteboer et al. [15], is based on both manual selection and identification of important anatomical structures. By having a trained radiologist, or similar, manually segment out the tumor in both modalities, and calculating, a set of binary images is obtained. After applying the transformation resulting from the registration on the moving image's segmented tumor, one can calculate the correlation between this and the fixed image's segmented tumor. At perfect registration, the overlap should ideally be 100%, and the two segmented images should be completely correlated. Any lower values will be due to misregistration, or to differences that the particular transformation could not accommodate; such as rotations when translational transforms are used, or deformations when a rigid transform is used.

While the manual segmentation of tumors in both modalities is a time-consuming and laborous task, and may introduce errors of its own, this method should possibly be employed to further test future registration methods that have already been verified by the subjective evaluation discussed above.

Part VI

Conclusion

In the course of the discussion of the results, in the previous part, several important considerations for future improvement of the registration applications have been mentioned. These will be collected and commented upon in this part, in chapter 18. The thesis will be completed with a conclusion, summarizing the results and experiences that have been obtained. The work of this thesis represents one of the first steps towards a fully operational volume-to-volume registration system. Focus is therefore kept mainly on the road ahead, on the most important lessons that should be brought along for future improvements, also in the conclusion.

18 Future work

Comparing the implemented registration applications and the obtained results, with the goals set forth for the project in the roadmap in chapter 3, it is clear that much remains. Three of the most important factors for this has been identified. First, not all the proposed metrics and feature extraction filters have been tested. Second, the implemented methods have not all been fully operational, and have not produced as reliable and consistent results as one could wish. Third, the tools for visualizing — and thereby also evaluating and validating — the registration results, has not been available.

However, the results suggest that the approaches used herein may give substantial results, but that they require some improvement. Some points for future development of volume-to-volume registration systems have already been made in part V. These, and others, are commented upon below.

18.1 Third phase application

Due to the stability problems with the second phase applications, the program for the US-to-US registration was not completed entirely. The current version employs non-rigid registration and the other methods discussed in part III, but remains untested. Testing, parameter tuning and bug-fixing of this application is remaining.

18.2 Optimizers and second phase applications

As discussed in chapter 15, both the applications developed for MRI-to-US registration have been unstable. However, the `RegApp2b`, using Mattes' MI is quite more unable to give consistent and reliable results. A different choice of optimizer should very possibly be considered for this program, or focus be placed entirely on the more stable NMI-based `RegApp2`. The NMI-based approach seems at the moment to offer a much better platform for future development. Incorporation of rotations into the `RegApp2`, with its evolutionary optimizer, should probably be evaluated before more time is spent on the `RegApp2b`.

18.3 Similarity measures and second phase applications

Of the three methods proposed in chapter 5, only one has been implemented currently. The other two methods, bivariate correlation ratio with gradient magnitude and intensity images, and cross-correlation of multilocal creaseness-filtered volumes, still warrant further investigation as the NMI-based approach is not working perfectly. Development of the required filters and metrics in ITK should be a priority, as this would further enhance the versatility and usefulness of the toolkit.

In addition, to improve the stability and accuracy of the current registration methods, different preprocessing filters available in ITK should be tested with the `RegApp2` and, possibly, `RegApp2b`. This includes, but is not limited to, anisotropic diffusion filtering and gradient filtering without Gaussian smearing. This will possibly alleviate some of the stability problems the applications currently experience.

18.4 Tools for visualization and validation

Also apart from the registration applications themselves is there room for improvements. Particularly, at the moment, an integrated visualization and volume manipulation platform is needed at SINTEF MedTech. The new version of CustusX used to render the volumes seen in this thesis, is still at alpha stage, and requires a number of additional features to be able to take this role. However, it has a strong volume visualization module, and a modular and robust design, which makes it a good vantage point for implementing these features. There are three components that are paramount to be able to visualize volumes both before and after registration, and to evaluate and validate their registration.

First, the visualization scene must enable the user to specify an approximate, initial transformational relationship between the volumes that should be registered. This includes, necessarily, that the registration applications themselves be incorporated into CustusX. Having this interface will enable the registrations to be performed with much stricter parameters, as only smaller offsets would have to be considered. The accuracy and reliability of the applications would naturally also increase.

Second, a method for inserting cutting planes into the volume rendered scene is highly needed. Planes where slices from the volumes could be visualized, in combination with a 3D-rendered scene, would facilitate quite accurate evaluation and presentation of registered data sets. The cutting planes, deciding which part of the volume should *not* be rendered, could either be allowed to travel only along the established axes, or allowed to be placed freely. The first approach is easier to implement, while the second offers more freedom, and is likely to lead to more powerful visualization possibilities.

Third, to further validate the registration results, and be able to compare different applications quantitatively, a manual registration interface is needed. This should enable the operator to select corresponding points in two volumes, either on a slice-by-slice basis, or by selecting points in a 3D scene, or a combination of the two. As long as the number of selected point pairs exceed the degrees of freedom of the registration transform, this can be used to calculate a “gold standard” transformation, against which the results from the various applications can be compared.

All these three suggestions should be considered for inclusion in the roadmap for future development of CustusX 2.

19 Conclusion

The distant goal towards which this thesis has been aiming is the implementation of automated volume-to-volume registration applications for three separate registration steps desired in enhancing neurosurgical navigation. For each of these phases, MRI-to-MRI registration, MRI-to-US registration and US-to-US registration, prototype implementations have been made using registration methods available in the Insight Toolkit. All implementations use variants of the Mutual Information similarity metric.

These applications have been tested on clinical data from relevant surgical operations. There have been some severe problems in the robustness of the methods, and in the ability to validate the results. Especially one of the MRI-to-US registration methods, using a Mutual Information similarity metric, has been very unstable, and has given quite unreliable results. The other MRI-to-US registration method, using Normalized Mutual Information, has proved somewhat more reliable, but with significantly longer running times. Both the MRI-to-US registration methods use a gradient magnitude filter to extract edge information from the images prior to registration, and there are indications that this filtering is in part the reason for the stability problems. Further, it has been experienced that gradient descent optimizers are inherently difficult to parameterize in order to obtain a stable and reliable optimization. While a multi-resolution registration framework is believed to encourage registrations that are correct on both a large and a small scale, the current results indicate that a rough initialization of the registration transform should be applied. This will enable the use of stricter bounds on the optimizers, further alleviating the reliability problems.

The quality of the evaluation and validation of the obtained results has further been reduced by the characteristics of the available visualization tools. To be able to assess the quality of registration results, visualization of volume slices in a 3D scene is needed. However, in the case of angiographic modalities the volume renderings used gave very good grounds for evaluating the registration.

Despite the problems regarding reliability and stability of the prototype implementations and the difficulties in the presentation of the results, typical results show that the angiographic volumes are registered. According to clinicians, the obtained results represent sufficient accuracy to enable its use in clinical applications. This requires, however, that the stability issues be resolved.

Ultimately, the obtained results show that automatic volume-to-volume registration using Normalized Mutual Information should be feasible for the neurosurgical application considered here. Both in registration of preoperative MRI volumes, registration of an MRI volume to an intraoperative US volume, and in registration of subsequent US volumes should further developed versions of the implemented applications be able to give sufficiently accurate results. The further developments include both evaluation of different optimization schemes, similarity metrics and preprocessing filters, and development of the necessary tools for performing accurate validation of registration results.

References

- [1] Richard D. Bucholz, David D. Yeh, Jason Trobaugh, Leslie L. McDurmont, Christopher D. Sturm, Carol Baumann, Jaimie M. Henderson, Ari Levy, and Paul Kessman. The correction of stereotactic inaccuracy caused by brain shift using an intraoperative ultrasound device. In *Proceedings of the First Joint Conference on Computer Vision, Virtual Reality and Robotics in Medicine and Medical Robotics and Computer-Assisted Surgery*, pages 459–466. Springer-Verlag, 1997.
- [2] E. Demidov. B-spline basis functions. <http://www.ibiblio.org/e-notes/Splines/Basis.htm> (2005-05-26), August 2001.
- [3] P.J. Edwards, D.J. Hawkes, G.P. Penney, and M.J. Clarkson. *Medical Image Registration*, volume 5 of *Biomedical Engineering*, chapter 12. Guiding Therapeutic Procedures, pages 253–280. CRC Press, 2001. Ed. Hajnal, J.V. and Hill, D.L.G. and Hawkes, D.J.
- [4] J.M. Fitzpatrick, D.L.G. Hill, and C.R. Maurer Jr. *Handbook of Medical Imaging*, volume 2. Medical Image Processing and Analysis, chapter 8. Image Registration. SPIE Press, 2004. Ed. M. Sonka and J.M. Fitzpatrick.
- [5] James D. Foley, Andries van Dam, Steven K. Feiner, and John F. Hughes. *Computer Graphics - Principles and Practice*, chapter 11.2. Parametric cubic curves, pages 478–515. Systems Programming Series. Addison-Wesley, 2nd edition, 1996.
- [6] David G. Gobbi, Roch M. Comeau, Belinda K.H. Lee, and Terry M. Peters. Integration of intra-operative 3d ultrasound with pre-operative mri for neurosurgical guidance. In *Engineering in Medicine and Biology Society, 2000. Proceedings of the 22nd Annual International Conference of the IEEE*, volume 3, pages 1738–1740. IEEE EMBS Society, 2000.
- [7] David G. Gobbi, Belinda K. H. Lee, and Terence M. Peters. Correlation of pre-operative mri and intraoperative 3d ultrasound to measure brain tissue shift. In Seong K. Mun, editor, *Medical Imaging 2001: Visualization, Display, and Image-Guided Procedures*, volume 4319, pages 264–271. SPIE, 2001.
- [8] David G. Gobbi and Terry M. Peters. Interactive intra-operative 3d ultrasound reconstruction and visualization. In Takeyoshi Dohi and Ron Kikinis, editors, *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2002, 5th International Conference, Tokyo, Japan, September 25-28, 2002, Proceedings, Part II*, volume 2489 of *Lecture Notes in Computer Science*, pages 156–163. Springer, 2002.
- [9] Aa. Gronningsaeter, A. Kleven, S. Ommedal, T.E. Aarseth, T. Lie, F. Lindseth, T. Langø, and G. Unsgård. Sonowand, an ultrasound-based neuronavigation system. *Neurosurgery*, 47(6):1373–1380, December 2000.
- [10] T.A.N. Hernes, F. Lindseth, T. Selbekk, O.M. Rygh, G.A. Tangen, I. Rasmussen, A. Wolff, O.V. Solberg, E. Harg, S. Augdal, F. Couweleers, and G. Unsgaard. Technological developments for improved computer assisted 3d ultrasound guided neurosurgery. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 2005. In press.
- [11] D.L.G. Hill and P. Batchelor. *Medical Image Registration*, volume 5 of *Biomedical Engineering*, chapter 3. Registration Methodology: Concepts and Algorithms, pages 39–70. CRC Press, 2001. Ed. Hajnal, J.V. and Hill, D.L.G. and Hawkes, D.J.
- [12] Luis Ibañes, William Schroeder, Lydia Ng, and Josh Cates. *ITK Software Guide*, chapter 8. Registration, pages 215–312. Kitware Inc., 2003.

- [13] Thomas Lange, Sebastian Eulenstein, Michael Hünerbein, and Peter-Michael Schlag. Vessel-based non-rigid registration of mr/ct and 3d ultrasound for navigation in liver surgery. *Computer Aided Surgery*, 8:228–240, 2003.
- [14] M.O. Leach. *The Physics of Medical Imaging*, chapter 8. Spatially Localised Nuclear Magnetic Resonance, pages 389–487. Medical Science. IOP Publishing Ltd., 1988. Ed. Webb, S.
- [15] M. Letteboer. *Intraoperative 3D ultrasonography for image-guided neurosurgery*. PhD thesis, University of Utrecht, 2004.
- [16] M.M.J. Letteboer, P.W.A. Willems, M.A. Viergever, and W.J. Niessen. Non-rigid registration of 3d ultrasound images of brain tumors acquired during neurosurgery. In R.E. Ellis and T.M. Peters, editors, *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003 6th International Conference, Montréal, Canada, November 15-18, 2003, Proceedings, Part II*, volume 2879 of *Lecture Notes in Computer Science*, pages 408–415. Springer, 2003.
- [17] F. Lindseth, J.H. Kaspersen, S. Ommedal, T. Langø, G. Unsgaard, and T.A.N. Hernes. Multimodal image fusion in ultrasound-based neuronavigation: improving overview and interpretation by integrating preoperative mri with intraoperative 3d ultrasound. *Computer Aided Surgery*, 8:49–69, 2003.
- [18] F. Lindseth, S. Ommedal, J. Bang, G. Unsgård, and T.A.N. Hernes. Image fusion of ultrasound and mri as an aid for assessing anatomical shifts and improving overview and interpretation in ultrasound-guided neurosurgery. In *Proceedings of Computer Assisted Radiology and Surgery (CARS)*, pages 247–252, Berlin, Germany, 2001. Elsevier Science B.V.
- [19] D. Lloret, J. Serrat, A.M. López, and J.J Villanueva. Ultrasound to magnetic resonance volume registration for brain sinking measurement. In F.J. et al. Perales, editor, *IbPria 2003*, number 2652 in *Lecture Notes in Computer Science*, pages 420–428. Springer-Verlag, Berlin, 2003.
- [20] D. Mattes, D. R. Haynor, H. Vesselle, T. Lewellen, and W. Eubank. Nonrigid multimodality image registration. In M. Sonka and K. M. Hanson, editors, *Medical Imaging 2001: Image Processing*, volume 4322 of *Proceedings of SPIE*, pages 1609–1620. SPIE, 2001.
- [21] C. Matula. Intra-operative ct and image-guided surgery: an introduction. *MedicaMundi*, 42(1):2–5, 1998.
- [22] C. Nimsky, O. Ganslandt, P. Hastreiter, and R. Fahlbusch. Intraoperative compensation for brain shift. *Surgical Neurology*, 56(6):357–364, December 2001.
- [23] X. Pennec, P. Cachier, and N. Ayache. Tracking brain deformations in time sequences of 3d us images. *Pattern Recogn. Lett.*, 24(4-5):801–813, 2003.
- [24] X. Pennec, A. Roche, P. Cathier, and N. Ayache. Non-rigid mr/us registration for tracking brain deformations. In R.S. Blum and Zh. Liu, editors, *Multi-Sensor Image Fusion and Its Application*. Marcel Dekker Inc, 2005.
- [25] Terry Peters, Bruce Davey, Patrice Munger, Roch Comeau, Alan Evans, and André Olivier. Three-dimensional multimodal image-guidance for neurosurgery. *Medical Imaging, IEEE Transactions on*, 15(2):121–128, 1996.
- [26] D.W. Roberts, A. Hartov, F.E. Kennedy, M.I. Miga, and K.D. Paulsen. Intraoperative brain shift and deformation: A quantitative analysis of cortical displacement in 28 cases. *Neurosurgery*, 43:749–760, 1998.
- [27] A. Roche, X. Pennec, G. Malandain, and N. Ayache. Rigid registration of 3-d ultrasound with mr images: A new approach combining intensity and gradient information. *Medical Imaging, IEEE Transactions on*, 20(10):1038–1049, 2001.

- [28] A. Roche, X. Pennec, G. Malandain, N. Ayache, and S. Ourselin. Generalized correlation ratio for rigid registration of 3d ultrasound with mr images. Rapport de recherche 3980, INRIA, Sophia Antipolis, France, 2000.
- [29] T.W. Sederberg, J. Zheng, A. Bakenov, and A. Nasri. T-splines and t-nurccs. *ACM Trans. Graph.*, 22(3):477–484, 2003.
- [30] C. Studholme, D.L.G. Hill, and D.J. Hawkes. Automated 3-d registration of mr and ct images of the head. *Medical Image Analysis*, 1(2):163–175, 1996.
- [31] C. Studholme, D.L.G. Hill, and D.J. Hawkes. An overlap invariant entropy measure of 3d medical image alignment. *Pattern Recognition*, 32:71–86, 1999.
- [32] P. Viola and W.M. III Wells. Alignment by maximization of mutual information. *Computer Vision. Proceedings, 5th Int. Conf on*, pages 16–23, 1995.
- [33] C.R. Wirtz, F.K. Albert, M. Schwaderer, C. Heuer, A. Staubert, V.M. Tronnier, M. Knauth, and S. Kunze. The benefit of neuronavigation for neurosurgery analyzed by its impact on glioblastoma surgery. *Neurological Research*, 22:354–360, 2000.

Part VII

Appendix

A Characteristics of the evolutionary (1+1) optimizer

Evolutionary optimizers try, as suggested by their name, to mimic the mechanisms of natural evolution, applying evolution-like rules to the iterative process of searching for the optimum. They do this by successively changing the parameter vector (or *individual*), which in the case of registration consists of the estimated transformation coefficients. The optimality of a certain parameter vector, called *fitness* in evolutionary programming, is determined using the supplied similarity metric. At each iteration, a number of *children* of the current individuals are generated by mutating the parents. Then the entire population (children and parents), is reduced back to the size of the last generation, but only keeping the fittest individual. In a common variant of the evolutionary optimizer, called the (1+1) evolutionary optimizer, only one individual survives each iteration, and only one competing children is generated each time. Thus, at any given iteration, the parent only has to be compared to its single child, to determine which of the two will be kept. The mutation that is applied to the parent, in order to generate its children, is modelled by a multi-normal distribution over the parameter vector space. This multi-dimensional Gaussian has mean $\mu = \bar{p}_{\text{parent}}$, and a covariance matrix Σ^2 , where \bar{p}_{parent} is the parameter vector of the parent. Further, the covariance matrix is changed for each iteration, by increasing it by a growth factor, c_{growth} , when the child is more fit, or by decreasing it by c_{shrink} when the parent survives. This ensures that the next step will either continue along good trends, while maintaining a wider search area when the fitness does not improve. In this basic formulation of the (1+1) evolutionary optimizer the same variance is used for all parameters, and the scaling of the covariance matrix is also done using a common scaling in all parameters. A more elaborate version, able to scale the covariance matrix coefficients independently, has also been suggested for use where the parameters of the feature space has an unequal scaling.

B File names and descriptive names

These are the relationships between the descriptive names used on each volume, and their corresponding file names:

Master:	US.
Dataset sub-name:	Axial Set.20050505T0020
fMRI-Finger w/ contrast	20041006T160127.mha
fMRI-Sentence w/ contrast	20041006T160309.mha
fMRI-Tongue w/ contrast	20041006T160445.mha
MRI-T2	20041006T160749.mha
MRI-T1 w/o contrast	20041006T161007.mha
MRA	20041006T161259.mha
MRI-T1 w/ contrast (underlay for fMRI)	20041006T161642.mha
UST (Tissue) #1	20041007T110127.mha
USA (Angio) #1	20041007T110555.mha
USA #2	20041007T110944.mha
USA #3	20041007T121921.mha
UST #2	20041007T122034.mha
UST #3	20041007T122213.mha
UST #4	20041007T124221.mha

Table 22: Tumor 1 — Corresponding descriptions and file names — US mastered.

Master:	MRI.
Dataset sub-name:	Axial Set.20050505T0012
fMRI-Finger w/ contrast	20041006T160127.mha
fMRI-Sentence w/ contrast	20041006T160309.mha
fMRI-Tongue w/ contrast	20041006T160445.mha
MRI-T2	20041006T160749.mha
MRI-T1 w/o contrast	20041006T161007.mha
MRA	20041006T161259.mha
MRI-T1 w/ contrast (underlay for fMRI)	20041006T161642.mha
UST (Tissue) #1	20041007T110127.mha
USA (Angio) #1	20041007T110555.mha
USA #2	20041007T110944.mha
USA #3	20041007T121921.mha
UST #2	20041007T122034.mha
UST #3	20041007T122213.mha
UST #4	20041007T124221.mha

Table 23: Tumor 1 — Corresponding descriptions and file names — MRI mastered

Master:	MRI.
fMRI - Tongue	20050308T075550.mha
MRI - T1 w/o contrast	20050308T080204.mha
MRI - T1 w/ contrast	20050308T080711.mha
MRI - T2	20050308T081151.mha
UST #1	20050308T102401.mha
USAngio #1	20050308T102756.mha
UST #2 (after tumor resection)	20050308T122737.mha
USAngio #2	20050308T123019.mha
UST #3	20050308T132358.mha
UST #4	20050308T135622.mha
USA #3	20050308T135841.mha
UST #5	20050308T142621.mha

Table 24: Tumor 2 — Corresponding descriptions and file names — MRI mastered

Master:	US.
MRI(T1?)	20050126T091032.mha
MRAngio	20050126T091242.mha
USAngio #1	20050126T105634.mha
USAngio #2	20050126T121206.mha

Table 25: Aneurism1 — Corresponding descriptions and file names — US mastered

C On calculating Mutual Information — the Viola & Wells way

The following describes how the Mutual Information (MI) metric is calculated, according to the implementational scheme by Viola & Wells [32], as used by e.g.

`itk::MutualInformationImageToImageMetric`.

An interesting starting point to look at how the MI is calculated, is by considering the parameters objects of the `itk::MutualInformationImageToImageMetric` class take, primarily: number of sample points and standard deviations for the fixed and moving images.

The number of sample points corresponds to the number of points throughout the two volumes that will be used to calculate the MI. Or, as the authors of the ITK puts it: “This is the number of image samples used to calculate the joint probability distribution” [12].

So, two further questions arise: 1) How are the sample points calculated, or more precisely, will the points be positioned in any uniform manner, or simply scattered randomly about the volume? 2) How are the sample points used to calculate the joint probability distribution, and how does this relate to the calculation of the MI?

C.1 How are sample points calculated?

As far as the first question goes, the Mutual Information Metric uses a point sampler function that works like this:

procedure *pointSampling(numberOfSamples)*

Require: *fixedImage* is a defined Image

Require: *movingImage* is a defined Image

Require: *latestTransform* is a mapping of space coordinates of *fixedImage* to *movingImage*

```

1:  $S \leftarrow \text{numberOfSamples}$ 
2: for all  $s$  such that  $s = 0 \dots S - 1$  do
3:    $N_f \leftarrow \text{numberOf Voxels In Fixed Volume}$ 
4:    $R \leftarrow \text{rand}(0, N_f)$ 
5:    $v_{xyz} \leftarrow \text{voxelNumberToImageCoordinates}(R)$ 
6:    $F_s \leftarrow \text{getIntensityAtCoordinates}(v_{xyz}, \text{fixedImage})$ 
7:    $p_{xyz} \leftarrow \text{imageToSpaceCoordinates}(v_{xyz})$ 
8:    $p_{x'y'z'} \leftarrow \text{transformSpaceCoordinates}(p_{xyz}, \text{latestTransform})$ 
9:   if  $p_{x'y'z'}$  is inside boundaries of moving image then
10:     $v_{x'y'z'} \leftarrow \text{spaceToImageCoordinates}(p_{x'y'z'})$ 
11:     $M_s \leftarrow \text{getIntensityAtCoordinates}(v_{x'y'z'}, \text{movingImage})$ 
12:   else
13:     $M_s \leftarrow 0.0f$ 
14:   end if
15:    $\text{sampleValues}[s] \leftarrow [F_s, M_s]$ 
16: end for
17: return  $\text{sampleValues}$ 

```

This shows that all the samples should be drawn randomly from the entire *fixed* volume. This will go a long way in explaining why you get different results when you swap the fixed and moving images.

However, in the ITK code, the “random” number generator is always given the same “seed” (starting at the same value) thus giving the exact same set of sample points every time the program is run with the same number of sample points input.

Technically speaking, the ITK library class of `itk::MutualInformationImageToImageMetric`, or its iterator, `itk::ImageRandomConstIteratorWithIndex`, does not seed the system’s `srand()` function, through the `vn1_sample_reseed()`, which the iterator at least at some point was meant to do: It contains methods like `ReinitializeSeed()`, that calls the `vn1_sample_reseed()`, but these are never activated. The reason for this, upholds the ITK administrators¹³, is to ensure test conformance. All ITK classes is tested each night, using a number of predefined tests, with specified inputs and assumeably correct outputs. To avoid getting errors in the testing scheme because of random differences in the sampling, the `itk::MutualInformationImageToImageMetric` deliberately does not make use of the supplied `ReinitializeSeed()` function. Thus, this has to be done in user-space, and the documentation for the ITK will presumably be updated to specifically mention this in future releases.

C.2 How do the samples relate to MI?

The more complex second question is then, how are the samples used to calculate the joint PDF, and what does this have to do with calculating the MI?

It may be useful to recap what MI is, and how it is, theoretically, calculated, before we look at how the practical, but approximate implementation in ITK does this.

MI measures how much one *random variable* tells about another random variable. When we operate with images, we need to treat the *images* as the random variables; that is multi-dimensional random variables in terms of image intensity. How should we ideally calculate MI? “As the difference between the sum of the entropies of the two random variables, and their joint entropy” [12]. Since the sum of the entropies correspond to the joint entropy when the two variables are completely independent, an MI value of 0 would indicate functional independence¹⁴, and a positive value of MI would indicate some degree of dependency.

Thus, MI is calculated as:

$$MI = H(A) + H(B) - H(A, B), \quad (3)$$

where $H(A)$, $H(B)$ are the entropies of the random variables A and B , respectively, and $H(A, B)$ is the joint entropy of A and B .

Then, $H(A)$, $H(B)$ and $H(A, B)$ can be calculated using:

$$H(A) = - \int p_A(a) \log(p_A(a)) da \quad (4)$$

$$H(B) = - \int p_B(b) \log(p_B(b)) db \quad (5)$$

$$H(A, B) = \int p_{AB}(a, b) \log(p_{AB}(a, b)) da db \quad (6)$$

¹³Ref. communication with the itk-users mailing list

¹⁴Nevertheless, it is possible to achieve a zero MI without independence, merely as a coincidence.

What is the problem with calculating these integrals for our images A and B? Apart from the mismatch between the continuous nature of integrals and the discrete nature of the images (i.e. the image consists of voxels of a finite size, and the integrals thus would have to be replaced with summations), the problem is that we don't have the precise probability distributions (pdfs) of the images. I.e. we do not have access to the functions $p_A(a)$, $p_B(b)$ and $p_{AB}(a, b)$.

This means that we will have to estimate these pdfs, using the image data as estimator data. To do this, a method known as kernel density estimation, a standard technique in statistical estimation, is used. This works like this, for one image/random variable:

- A number of (at least ideally) random samples are taken from the random variable. In the case of our images, this is done by the iterator and point sampler discussed in question no. 1.
- A kernel function, such as a Gaussian, is associated with each sample point's value, so that for each sample point the kernel function "reaches" a little further in both directions on the intensity scale.
- To construct the pdf for the random variable, you could either just traverse the intensity scale from the minimal to the maximal value (whether it is 0-255 or in floating point precision such as -1.0 to +1.0), or select some points on the intensity scale where one wants to find the pdf value:

All the small kernel functions are spread out so that, summed up at each intensity value, they define the pdf, and we want a robust way to estimate this pdf.

Using a uniform stepping, in terms of intensity value, between the points where we calculate the pdf value (from the sum of the kernel) functions at that intensity value), from minimal to maximal intensity, will give us a complete pdf. However, we are probably more interested in the pdf being more accurate in those areas of the intensity scale where most of our pixels are, than elsewhere.

Thus, we use the *intensity values* of another set of sample points from the image, as points on the intensity scale where we want to calculate the pdf. This will increase the probability that the pdf we get is most accurate in those parts of the intensity axis where we have the most voxels.

The second sample set, of the same size as the first, is thus used to find the intensity values where we want to calculate the (estimated) pdf. The pdf is thus never really constructed as a function, we just find its estimated value for the intensity values of each of the points in the second sample set, using this equation:

$$p(r) \simeq \frac{1}{N} \sum_{\forall s \in S} K_{\text{gauss}}(r - s) \quad (7)$$

and, $p_{AB}(r) = p_A(r) \cdot p_B(r)$, as we are assuming independence of the two variables for the joint pdf.

In the previous equations, r is the intensity value for which we want to know the estimated pdf, $\forall s \in S$ are the intensity values at all the sample points of the first sample set, and $K_{\text{gauss}}()$ is the Gaussian kernel function, with the standard deviation given as parameters to the objects of the MI metric class.

These pdf values are directly used in calculating the entropy, using another approximate function. Instead of the accurate function:

$$H(A) = - \int p_A(a) \cdot \log(p_A(a)) da \quad (8)$$

we use the “sample mean” approximation:

$$H(A) \simeq -\frac{1}{N} \sum_{\forall r \in R} (\log P(r)) \quad (9)$$

where the factor N corresponds to the number of samples, and $\forall r \in R$ means the intensity values r of all the samples in the second sample set R . This way of calculating the entropy is obviously less accurate than the integral one, but the best we can do. It may seem odd that we just crudely take the mean of all the log probabilities, but this gives a good estimate of the original $\sum(p(a) \cdot \log p(a))$, since the $p(a)$ part is maintained by the fact that we are likely to have a second sample set where the most probable intensities occur more often than less probable ones. That is: $R \sim p(a)$, making it unnecessary to include $p(a)$ in our calculation.

Doing this for both the fixed image (A), moving image (B) and joint ((A, B)) probabilities/entropies, we get the $H(A)$, $H(B)$ and $H(A, B)$. Then we calculate the MI by:

$$MI = H(A) + H(B) - H(A, B) \quad (10)$$

Technically, a few tricks have been used in the calculation, most probably to save accuracy. This includes using:

$$P(r) \simeq \sum_{\forall s \in S} K_{\text{gauss}}(r - s) \quad (11)$$

$$H(A) \simeq \sum_{\forall r \in R} (-\log(P_A(r))) \quad (12)$$

$$H(B) \simeq \sum_{\forall r \in R} (-\log(P_B(r))) \quad (13)$$

$$H(A, B) \simeq \sum_{\forall r \in R} (-\log(P_A(r) \cdot P_B(r))) \quad (14)$$

and subsequently correcting for the lack of $1/N$ factors, using:

$$MI = \frac{1}{N} \cdot (H(A) + H(B) + H(A, B)) + \log(N) \quad (15)$$

the first $1/N$ corresponding to the lacking $1/N$ in the $H(A)$ etc., and the $+\log(N)$ corr. to the lacking $1/N$ inside the $P(r)$.

C.3 Summary

The Mutual Information metric is calculated by estimating the intensity probability distributions of each of the images, and their joint probability distribution, by taking a number of samples and constructing a pdf using kernel density estimation.

This pdf is used as a basis for further estimating the images' respective entropies, and their joint entropy, using another set of samples (of the same size as the first).

The MI is then calculated "as usual"; as the difference between the sum of entropies and the joint entropy.

More sample points is likely to give a more accurate estimate of both the probability distribution functions and the entropies, but as the computation involves a $N \times N$ loop (to find the N entropy sample points, each of them includes finding N values to sum up the pdf at that intensity value), the number of sample points will greatly affect the running time.

D Questionnaire for evaluation of registration results

Volume-to-volume registration
evaluation of results
for
Erik Harg

A presentation of volumes from three data sets will be performed; including preoperative MRI and intraoperative 3D ultrasound. Automatic programs for volume-to-volume registration have been run on these volumes, to register, respectively, different MRI volumes, MRI-and-US volumes and different US volumes, from the same patient.

You are kindly requested to answer the following question for each registered volume pair:
"Is the registered volume sufficiently adapted to the reference volume, to be used for clinical purposes?"

- *reference volume* refers to the volume that remains unchanged
- *registered volume* refers to the volume that is changed by registration
- *clinical purposes* primarily refers to planning of, and navigation during, neurosurgery

Answer forms are included on the following pages, with one page per registration phase.
Other comments may be added at the last page.

First, please indicate your profession:

Clinician/Surgeon	Tech/Researcher	Student
-------------------	-----------------	---------

Figure 53: Questionnaire used in evaluation of registration results — Page 1.

MRI-MRI
Dataset 1

R1:

Not sufficient	Possibly sufficient	Sufficient
----------------	---------------------	------------

R2:

Not sufficient	Possibly sufficient	Sufficient
----------------	---------------------	------------

Figure 54: Questionnaire used in evaluation of registration results — Page 2.

MRI-US
Dataset 1
R1:

Not sufficient	Possibly sufficient	Sufficient
----------------	---------------------	------------

R3:

Not sufficient	Possibly sufficient	Sufficient
----------------	---------------------	------------

Dataset 2
R5:

Not sufficient	Possibly sufficient	Sufficient
----------------	---------------------	------------

Dataset 3
R6:

Not sufficient	Possibly sufficient	Sufficient
----------------	---------------------	------------

Figure 55: Questionnaire used in evaluation of registration results — Page 3.

Other comments:

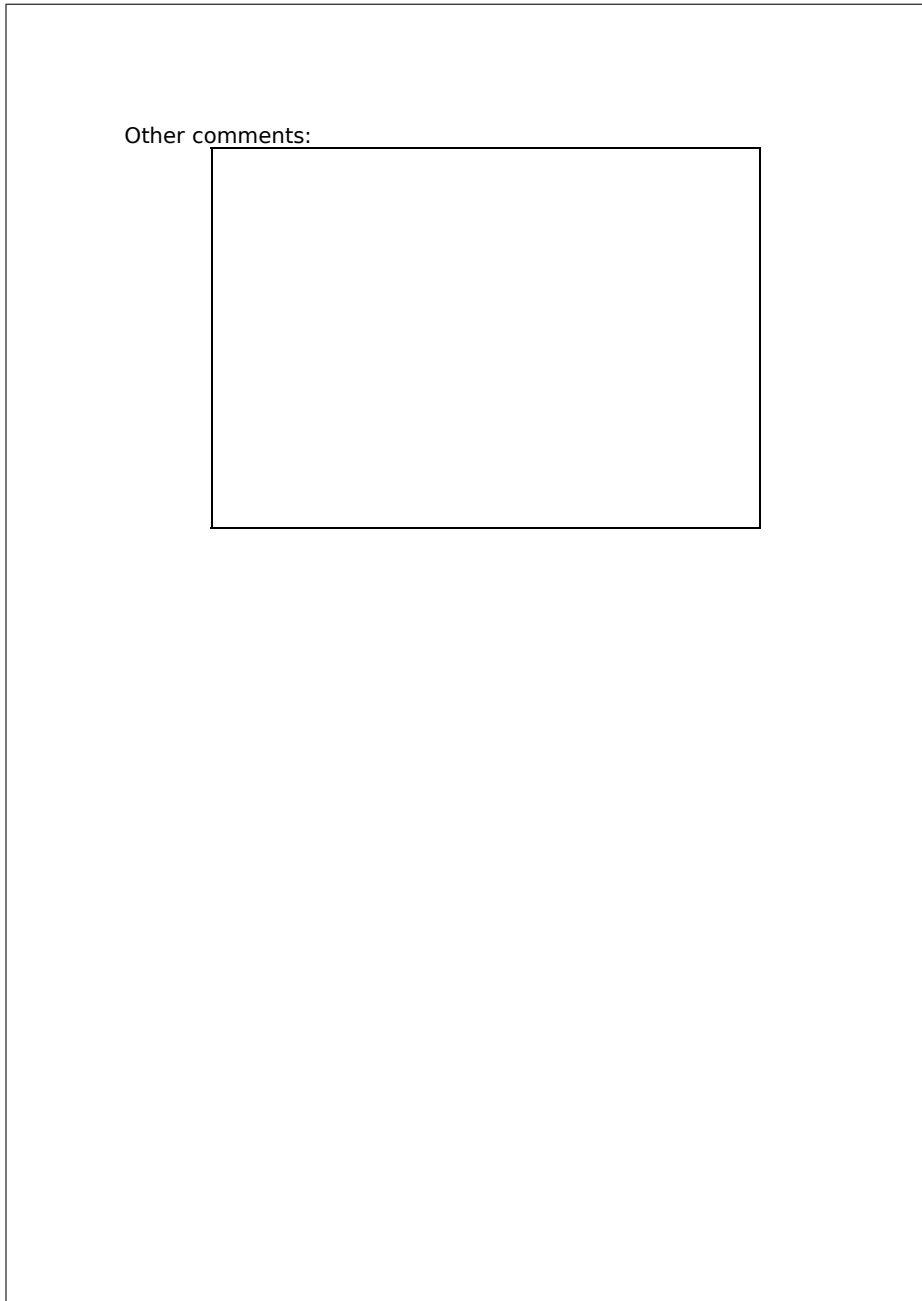
The image shows a large, empty rectangular frame representing a questionnaire page. Inside this frame, in the upper-left quadrant, is the text "Other comments:" followed by a smaller, empty rectangular box intended for handwritten or typed input.

Figure 56: Questionnaire used in evaluation of registration results — Page 4.

E Notes

BSD is a registered trademark of Berkeley Software Design, Inc.

GNU is a registered trademark of the Free Software Foundation.

Java is a registered trademark of Sun Microsystems, Inc.

Linux is a registered trademark of Linus Torvalds.

Mac OS X is a registered trademark of Apple Computer, Inc.

Microsoft is a registered trademark of Microsoft Corp.

PowerPC is a registered trademark of International Business Machines Corp.

Python is a registered trademark of the Python Software Foundation.

SonoWand is a registered trademark of MISON AS.

UNIX is a registered trademark of The Open Group.

Windows is a registered trademark of Microsoft Corp.