Plastimatch 1.6 – Current Capabilities and Future Directions

James A. Shackleford[1], Nadya Shusharina[1], Joost Verberg[1], Guy Warmerdam[1], Brian Winey[1], Markus Neuner[2], Philipp Steininger[2], Amelia Arbisser[3], Polina Golland[3], Yifei Lou[4], Chiara Paganelli[5], Marta Peroni[5], Marco Riboldi[5,6], Guido Baroni[5,6], Paolo Zaffino[7], Maria Francesca Spadea[7], Aditya Apte[8], Ziad Saleh[8], Joseph O. Deasy[8], Shinichro Mori[9], Nagarajan Kandasamy[10], Gregory C. Sharp[1]

[1] Massachusetts General Hospital, Radiation Oncology
[2] Paracelsus University, Institute of Research and Development on Advanced Radiation Technologies (radART)
[3] Massachusetts Institute of Technology, Computer Science and Engineering
[4] Georgia Institute of Technology, Electrical and Computer Engineering
[5] Politecnico di Milano, Biomedical Engineering
[6] Centro Nazionale di Adroterapia Oncologica, Bioengineering Unit
[7] Magna Graecia University, Experimental and Clinical Medicine
[8] Memorial Sloan Kettering Cancer Center, Department of Medical Physics
[9] National Institute of Radiological Sciences, Research Center for Charged Particle Therapy
[10] Drexel University, Electrical and Computer Engineering

Abstract. Open-source software provides an economic benefit by reducing duplicated development effort, and advances science knowledge by fostering a culture of reproducible experimentation. This paper describes recent advances in the plastimatch open software suite, which implements a broad set of useful tools for research and practice in radiotherapy and medical imaging. The focus of this paper is to highlight recent advancements, including 2D-3D registration, GPU-accelerated mutual information, analytic regularization of B-spline registration, automatic 3D feature detection and feature matching, and radiotherapy plan evaluation tools.

Keywords: Open source software, medical imaging, software engineering, radiotherapy.

1 Introduction

The fields of medical imaging and radiotherapy use sophisticated software to achieve advanced medical results. Commercial software is generally closed source, which makes it difficult to use for research, and often lacks the flexibility to communicate with complementary software from other vendors. This gap in functionality is properly filled by open-source solutions, which are well suited for

research. This paper describes the plastimatch software suite for radiotherapy image processing [1]. Plastimatch is open-source software, distributed under a BSD-style license. The focus of plastimatch is on high-performance algorithms for medical image computing, and on flexible radiotherapy utilities. Using standard interchange formats such as DICOM and DICOM-RT, plastimatch can be easily used together with other open source tools, including CERR [13], Conquest DICOM [2], ImageJ [3], VV [23], and 3D Slicer [4].

2   Background and History

The origin of the plastimatch project was a collection of image processing software written to support medical physics research. The first utility, DRR generation, was written in 2003. Shortly thereafter, deformable image registration was developed using the Insight Segmentation and Registration Toolkit (ITK) [5], and cone-beam CT reconstruction using the Feldkamp, Davis and Kress (FDK) algorithm [14].

In an attempt to overcome the long computation times of ITK's deformable registration algorithms, we began exploring GPU accelerated algorithms in 2006. The first algorithms targeted were FDK filtered backprojection and demons deformable registration [30]. At the time, CUDA was not yet mature, and therefore these algorithms were deployed using the BrookGPU compiler and runtime library [9]. Working together with our institutional lawyers, we released the first open source version of the GPU Imaging Toolkit (GPUIT) in early 2007. Later in 2007, the GPUIT registration routines were consolidated with the ITK registration routines to create the plastimatch project.

Table 1: Summary of plastimatch algorithms for 3D image registration.

Registration Algorithm	ITK	SC	MC	GPU
Translation	✓			
Rigid	✓			
Affine	✓			
Demons	✓	✓		✓
B-spline (MSE,MI)	✓	✓	✓	✓
Viscous Fluid				✓
Thin-plate spline	✓			
Wendland, Gaussian Spline		✓		

In 2008, a native implementation of B-spline registration was designed to replace the ITK implementation, and the FDK reconstruction code was ported from BrookGPU to CUDA. In 2009, we added GPU accelerated B-spline registration [24], landmark-based interactive registration, and DICOM-RT. This year also saw the first usable user interfaces, as command-line programs launched from 3D Slicer and Aqualizer. In 2010, we considerably improved the landmark-based registration by offering three different algorithms: thin-plate spline, Wendland radial basis spline, and regularized Gaussian spline. The first GPU accelerated

mutual information algorithm was released this year, together with OpenCL acceleration, and dose volume histogram (DVH) computation capabilities. In 2011, we released Reg23, a full-featured 2D-3D registration algorithm, capable of automatic registration using five different cost functions, interactive initialization, and programmable pre-processing and post-processing. Analytic regularization of B-spline registration was also introduced, as was automatic feature detection and matching using the scale-invariant feature transform (SIFT) algorithm.

3 Overview of Plastimatch

Traditionally, plastimatch has been conceived and developed as an end-user application, rather than as a library or toolkit. The standard method of using plastimatch is on the command line, with configuration files and command line options. A typical invocation would be to specify a command, such as register, together with the necessary input files, configuration files, and options. A list of supported commands are shown in the usage screen:

```
$ plastimatch --help
plastimatch version 1.5.11-beta (3583M)
Usage: plastimatch command [options]
Commands:
  add         adjust      average     crop        compare
  compose     convert     diff        dvh         fill
  header      mask        probe       register    resample
  scale       segment     stats       synth       synth-vf
  thumbnail   warp        xf-convert
```

3.1 Automatic 3D-3D Registration

Plastimatch uses a multi-stage, multi-algorithm framework for automatic image registration. Only pairwise registration is supported. In the initialization stage, the images are loaded, together with any image masks or initial transformations. The framework runs a fixed sequence of registration stages, as directed by a parameter file. Each registration stage specifies the image resolution (for multi-resolution registration), the transform and metric to be optimized, and the optimization algorithm and parameters. If desired output files can be specified at each stage for saving intermediate results. A typical sequence of stages might include a single rigid alignment stage, followed by two to four deformable registration stages with increasing resolution and decreasing grid spacings.

Table 1 lists the summary of algorithms included in plastimatch, which includes six different core registration methods. Depending on the registration method, you can choose one of four implementations: ITK, single core (SC), multi-core (MC), or GPU. The six registration algorithms can operate on eight different transform types: six ITK transforms and two native transforms. At the end of each stage, the optimal transform is propagated to the next stage, and automatically converted to a new transform type by the plastimatch application framework.
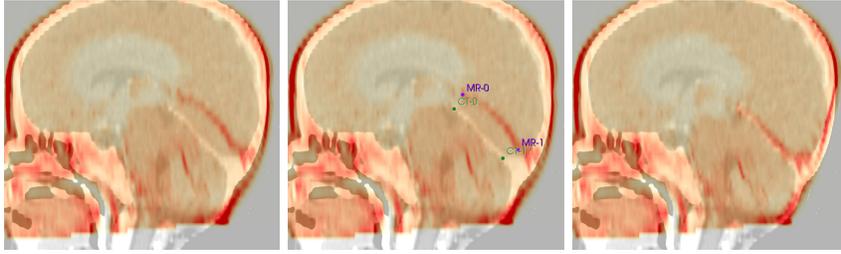
Fig. 1: Interactive registration is used to warp the MRI of a 6-month old infant onto the CT of the same patient at age 2. The initial registration properly matches the skull, but features within the brain are not properly aligned (left). Landmarks are placed (center), which improve the registration (right).

3.2 Cone-beam CT and Digitally Reconstructed Radiographs

A cone-beam CT reconstruction application is provided which implements filtered back projection using the Feldkamp, Davis and Kress (FDK) algorithm. Input images in either raw, pfm, or hnd format are read, filtered, and backprojected into a user defined volume geometry. Images in raw or pfm format must be accompanied by a geometry specification file, whereas files in the Varian hnd format use the geometry specified by the file header. Ramp filtering is performed on the CPU using the FFTW library [16], while backprojection is performed on either CPU or GPU. The plastimatch DRR generator implements three variants of the Siddon ray tracing method [26]. The fastest and most popular method uses the original exact pathlength method based on the intersection of rays with the image voxels. In addition, two voxel interpolation methods are included, which can be used to increase the apparent resolution of the DRR construction. Both multi-core and GPU versions are available.

3.3 Interactive (Landmark-based) Image Registration

While automatic registration yields acceptable results in many cases, we are often confronted with difficult registration problems where automatic registration fails. For this purpose, plastimatch includes two manual registration tools: a global landmark-based tool based on thin plate splines, and a tool based on radial basis functions (RBF) which allows us to make local registrations by adjusting the RBF support .

The global tool is implemented as an ITK wrapper, takes a list of corresponding points in 3D, and generates a complete vector that interpolates all of the input landmarks. This method requires a minimum of six landmarks, which are used to find a global affine transform superimposed with a minimum energy deformation field [8]. The global landmark registration results can be used as a standalone method, or to initialize the automatic registration. In contrast, the RBF tool is a native warper, and does not perform global rigid or affine mapping. Instead, it uses a small number of landmark pairs to correct failed deformable registration results. The algorithm utilizes two types of radial basis functions,

Wendland function with finite support [7, 15] and non-truncated Gausian function [6, 25]. In both cases, a deformation is found by solving a system of linear equations which is computationally very efficient as compared with algorithms based on complex multidimensional minimization. In addition, Gaussian RBF have a distinct feature with respect to regularization, because the regularized vector field can be solved exactly with a simple equation. An independent regularization parameter is defined to control the balance between the fidelity of the alignment of landmark pairs, and the smoothness of the deformation field. An example of this idea is shown in Fig. 1, where the failed registration (left), is corrected using two pairs of landmarks (center, right).

3.4 2D-3D Registration

The Reg23 module of plastimatch along with the oraifutils-sublibrary enables rigid registration of a 3D volumetric image (e.g. a computed tomography) with an essentially arbitrary number of projective 2D images (e.g. X-rays). The transformation parameters (3 rotations, 3 translations) are iteratively optimized with respect to a cost function which assesses the similarity between the X-rays and on-the-fly DRRs computed from the volume. Uniform ray-casting DRR computation is implemented on the GPU using the openGL shading language (GLSL). Beside of selected similarity metrics (normalized mutual information, normalized cross correlation, gradient difference, mean reciprocal square difference) derived from ITK, stochastic rank correlation [27] is a further configurable cost function. All input images can be configured to be pre-processed prior to registration including resampling, rescaling, cropping and unsharp masking. Downhill simplex (AMOEBA) and 1+1 evolutionary algorithms are available for optimization. In order to restrict similarity evaluation to a certain region of interest (ROI) in the X-rays, a so-called auto-masking module is available [22]. Based on RT structure sets which are typically generated in the pre-planning stage, an entity-specific heuristic is configurable which allows logical combination, dilation/erosion and projection of structures onto the X-ray planes which produces binary mask images that constrain metric evaluation. For example, in the case of pelvis registration, this mechanism enables automatic determination of ROIs that exclude the femora which are prone to move relatively over the treatment course [22, 28, 29]. Fig. 2 gives a schematic overview of the main components using the example of dual 2D/3D pelvis registration.

In addition to the core algorithm offering the mentioned capabilities, a Qt-based general user interface (GUI) is provided as shown in Fig. 2. The GUI enables the user to monitor the registration process, and simultaneously to influence registration by mouse interactions (translation, rotation, registration initialization). The overall program is configurable via a simple ASCII-based file in order to enable easy integration with other applications such as recordverify-systems. Moreover, utilizing this mechanism batch processing is achievable where the registration results are stored in output files. Current work aims at providing more convenient means for setting up the imaging geometry, extending the port-
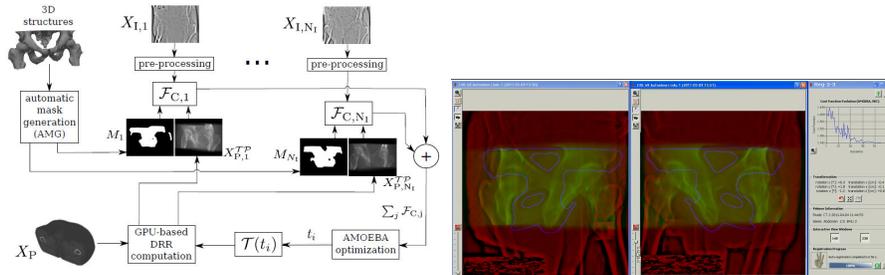
Fig. 2: (left) Schematic overview of Reg23 components and their interconnections. (right) Reg23 GUI showing colored overlays of X-rays and DRRs. The ROI generated by the auto-masking module is shown as a blue contour. Registration parameters are displayed in the control panel on the right.

folio of available DRR algorithms and implementing appearance-model-based 2D/3D registration.

3.5 Automatic Feature Detection and Matching

Several algorithms have been developed to perform automatic landmarks extraction and matching, with the goal of increasing the accuracy of detection and decreasing the cost in terms of time. First operators for the automatic extraction of points have examined an image at a single scale, thus limiting the accuracy and stability of feature detection. Scale Invariant Features Transform (SIFT) is a method that provides extraction and matching of stable and prominent points at different scales between two images. The algorithm supported by Plastimatch was derived from Cheung and Hamarneh [11] and implemented in C++ using the Insight Toolkit (ITK) [5]. This method takes in inputs two 3D (isotropic or anisotropic) images and generates lists containing stable landmarks for each image as well as feature matches between the two images. The output files (.fcsv) contain landmarks in physical coordinates that can be used with 3D Slicer Fiducial module [4]. Figure 1 shows examples of successful individuation of corresponding features in the original (left) and synthetic (right) image of a phantom (RANDO phantom, The Phantom Laboratories, Salem, NY). The synthetic image is obtained by applying rigid and non-rigid transformations to the phantom.

3.6 Data Interchange

Plastimatch supports a wide variety of file input types for data interchange. Using ITK wrappers, most image formats are supported, including DICOM, Analyze, Metaimage, and NRRD. In addition, partial support exists for DICOM-RT, XiO, and RTOG formats. Plastimatch is capable of rasterizing DICOM-RT structure sets into images, as well as converting images back into DICOM-RT structure sets. In addition, a utility is provided for attaching existing DICOM-RT structure sets onto arbitrary DICOM series.
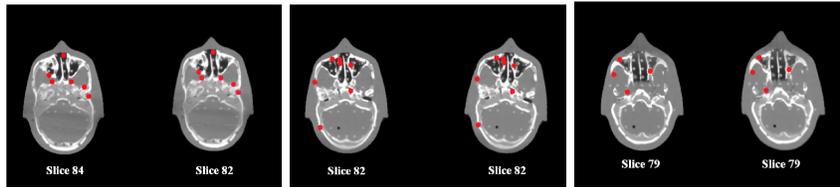
(a) Translation 6mm   (b) Rotation 5 degrees   (c) Non-rigid transform

Fig. 3: Successful examples of successful corresponding features detection (red codes) in the original (right) and synthetic (left) image of RANDO phantom. Rigid Transforms: (a) Translation [6 mm] in right-left, anterior-posterior, superior-inferior directions and (b) Rotation [5] in superior-inferior direction. Non-rigid transform: (c) maximum deformation of 15.42, 5.72, 4.16 mm in right-left, anterior-posterior, superior-inferior directions, respectively.

3.7   User Interface

While a native user interface is supported by Reg23, the plastimatch module offers a user interface only as a plugin for Aqualizer [21] and 3D Slicer [4]. Aqualizer is a specialized research software for four-dimensional treatment planning. Deformable image registration is used to map radiation dose from all breathing phases onto a reference phase, and accumulate the time-averaged dose. 3D Slicer is a general purpose research software for medical image computing. Plastimatch plugins are available for automatic registration, landmark-based registration, and DICOM-RT import.

4   Future Directions

4.1   Embedded Scripting

Plastimatch 1.6 features a new Lua-based embedded scripting engine that allows users to construct custom workflows quickly and easily with minimal programming experience. Images, transforms, and core plastimatch algorithms are exposed through the scripting interface as simple abstract objects, which the user can use to design an image processing pipeline specific to their particular application. Fig. 4 demonstrates the usage of image, registration, and transform objects to construct a 4D registration workflow in just 17 lines, excluding comments. For users who need more flexibility and dynamic control, the scripting engine can be ran in interactive mode. In this mode, the user is provided with a command driven interface similar to MATLAB where they may navigate the filesystem, use all available objects to operate on their data, run pre-written scripts, and preview resulting images using an integrated slice viewer before writing them to disk.

4.2   Viscous Fluid Registration

The viscous fluid model [12] for registration assumes that the deformation is governed by the Navier-Stokes equation of viscous fluid motion. This approach takes

```
-- load breathing phase 3 as the reference image for each pair-wise registration
reference_phase = Image.load ("data/phase_03.mha")
registration = Register.load ("data/stages.txt")
registration.fixed = reference_phase
-- define remaining phases in a phase array
phases = {
  { image = "data/phase_01.mha", warp = "output/warp_01.mha", xf = "output/xform_01.mha" },
  { image = "data/phase_02.mha", warp = "output/warp_02.mha", xf = "output/xform_02.mha" },
  { image = "data/phase_04.mha", warp = "output/warp_04.mha", xf = "output/xform_04.mha" },
  { image = "data/phase_05.mha", warp = "output/warp_05.mha", xf = "output/xform_05.mha" }
}

-- loop through every input/output pair "p" in our phases array
for _,p in pairs(phases) do
  -- load the phase and use as moving image for pair-wise registration
  destination_phase = Image.load (p.image);
  registration.moving = destination_phase
  -- perform the registration and save the transform
  xf = r:go()
  -- warp the destination phase and save outputs
  warp = destination_phase + xf
  warp:save (p.warp);
  xf:save (p.xf);
end
```

Fig. 4: The new scripting system uses a familiar object-oriented approach to provide users with direct access to core Plastimatch algorithms. This allows for rapid design of custom workflows, such as the 4D registration workflow shown here.

into account the physical model of tissue motion to regularize the deformation fields, thus able to handle larger deformation compared with optical flow [17] and elastic models [10]. A similarity measure is first established to quantify how close two image volumes are to each other according to some chosen metric. For multimodal registration, mutual information (MI) is a popular choice. Recently a new measure is proposed by Lou et al. [18], called Bhattacharyya distance (BD). It can avoid the instability issues that are present in MI. Next the deformation that maximizes this similarity is computed by solving a N-S equation, in which the force field that drives the deformation in an appropriate direction is parallel to the gradient of the chosen similarity measure. The optimal solution is iteratively updated by convolving the force field with a Gaussian filter as an approximation. Moreover a multi-resolution scheme is used to increase the speed and robustness. We implement the viscous fluid model with either MI or BD as the similarity measure on the GPU [19]. It takes 6 seconds to register two images of size 256x256x60 for 50 iterations using three multi-resolution levels.

4.3 Deformable Image Registration in CERR

While it is possible to use plastimatch with CERR via DICOM-RT file exchange, this method is somewhat time consuming and complicated for end-users. During 2012, we intend to supply an interface for using plastimatch directly from within CERR. The first stage of development will define a file-based method for exchanging images, doses, and structure sets; thereby enabling dose warping, dose accumulation, structure set warping, and automatic contouring. This interface will provide storage of the B-spline coefficients or dense vector fields

for further external processing. Second stage development will investigate batch processing methods and memory-based data exchange.

4.4 Multi Atlas-based Segmentation

Plastimatch will soon be adding support for multi-atlas-based segmentation, which aims to automatically delineate structure boundaries for an unlabeled target CT scan. Provided an atlas consisting of CT scans from other patients with the relevant structure labeled, each CT scan in the atlas is registered to the unlabeled target scan and the resulting transform is applied to each of the labeled atlas structures. This results in the labels from each of the atlas scans being mapped into the space of the target as shown in Fig. 5. To distill these multiple delineations into a single label for the target, a vote is conducted for each target scan voxel to determine if it is within the structure. With this label fusion, multi-atlas-based segmentation can delineate each relevant structure in the target CT.
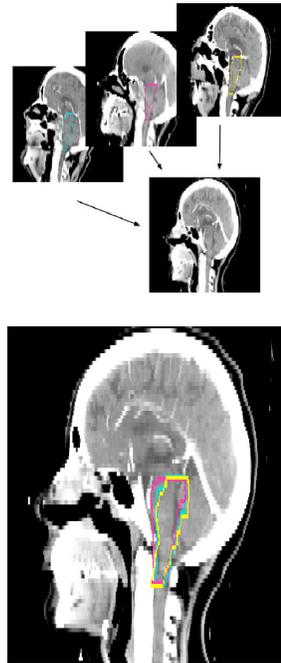

Fig. 5: Multi-atlas based segmentation uses voting to combine multiple registration results into a best contour for a structure.

5 Software Engineering

As of June 2012, plastimatch comprises 163,000 lines of cross-platform code, and makes heavy use of the external libraries: ITK, VTK, Qt, DCMTK, FFTW, CUDA, and SQLite. Platform specific build configurations are managed using CMake [20], which accepts a build description language and generates a sane build environment specific to the available external libraries and compiler features. The absence of a library results in reduced functionality; for example, if CUDA is not installed plastimatch will still build but CUDA-accelerated implementations will be unavailable. The plastimatch philosophy is that external libraries are optional.

Software version control and defect tracking are managed through a custom web-enabled install of GForge Advanced Server. Source code and documentation are disseminated though a subversion repository, which allows anonymous access. As a measure to maintain and improve software quality, the CTest system is used to perform automated software testing. Test cases are included with the software distribution and can be executed using the CMake generated build environment. Prior to committing changes, developers execute a battery of tests to ensure that functionality has not regressed. Additionally, automatic regression testing is performed nightly across various build configurations.

6 Licensing

The plastimatch software is licensed under a BSD license for Reg23, and a custom BSD-style license for plastimatch. These licenses specifically allows royalty-free non-exclusive license to use, modify, and redistribute our software. The primary restrictions on licensing are that (1) attribution and copyright notices be retained, (2) modified versions must be clearly marked, and (3) names, logos, and trademarks of our institutions are not used for promotion. Our software is provided AS IS, without warranty. The custom license clearly states that the software has been designed for research purposes only, and that clinical applications are neither recommended nor advised. A complete copy of the license is available on-line [1].

Acknowledgments

References

1. Plastimatch. http://plastimatch.org
2. Conquest DICOM. http://www.xs4all.nl/~ingenium/dicom.html
3. ImageJ. http://rsb.info.nih.gov/ij
4. 3D Slicer. http://slicer.org
5. The Insight Segmentation and Registration Toolkit. http://itk.org
6. Arad, N., Dyn, N., Reisfeld, D., Yeshurun, Y.: Image warping by radial basis functions: Application to facial expressions. CVGIP: Graphical Model and Image Processing 56(2), 161–172 (1994)
7. Arad, N., Reisfeld, D.: Image warping using few anchor points and radial functions. In: Computer Graphics Forum. vol. 14, pp. 35–46. Wiley Online Library (1995)
8. Bookstein, F.: Principal warps: Thin-plate splines and the decomposition of deformations. Pattern Analysis and Machine Intelligence, IEEE Transactions on 11(6), 567–585 (1989)
9. Buck, I., Foley, T., Horn, D., Sugerman, J., Fatahalian, K., Houston, M., Hanrahan, P.: Brook for GPUs: stream computing on graphics hardware. In: ACM Transactions on Graphics (TOG). vol. 23, pp. 777–786. ACM (2004)

10. Chefd'Hotel, C., Hermosillo, G., Faugeras, O.: A variational approach to multimodal image matching. In: Variational and Level Set Methods in Computer Vision, 2001. Proceedings. IEEE Workshop on. pp. 21–28. IEEE (2001)
11. Cheung, W., Hamarneh, G.: n-SIFT: n-Dimensional Scale Invariant Feature Transform. Image Processing, IEEE Transactions on 18(9), 2012–2021 (2009)
12. D'Agostino, E., Maes, F., Vandermeulen, D., Suetens, P.: A viscous fluid model for multimodal non-rigid image registration using mutual information. Medical image analysis 7(4), 565–575 (2003)
13. Deasy, J., Blanco, A., Clark, V.: CERR: a computational environment for radiotherapy research. Medical physics 30, 979 (2003), http://radium.wustl.edu/CERR
14. Feldkamp, L.A., Davis, L.C., Kress, J.W.: Practical cone-beam algorithm. J. Opt. Soc. Am. A 1(6), 612–619 (Jun 1984)
15. Fornefett, M., Rohr, K., Stiehl, H.: Radial basis functions with compact support for elastic registration of medical images. Image and Vision Computing 19(1), 87–96 (2001)
16. Frigo, M., Johnson, S.: The design and implementation of FFTW3. Proceedings of the IEEE 93(2), 216–231 (2005)
17. Horn, B., Schunck, B.: Determining optical flow. Artificial intelligence 17(1-3), 185–203 (1981)
18. Lou, Y., Irimia, A., Vela, P., Chambers, M.C., van Horn, J., Vespac, P.M., Tannenbaumd, A.: Multimodal Deformable Registration of Traumatic Brain Injury MR Volumes via the Bhattacharyya Distance. Medical Image Analysis (2012 Submitted)
19. Lou, Y., Jia, X., Gu, X., Tannenbaum, A.: A GPU-based Implementation of Multimodal Deformable Image Registration Based on Mutual Information or Bhattacharyya Distance. http://hdl.handle.net/10380/3268 (May 2011)
20. Martin, K., Hoffman, B.: Mastering CMake. Kitware, Inc (2010), http://cmake.org
21. Mori, S., Chen, G.: Quantification and visualization of charged particle range variations. International Journal of Radiation Oncology* Biology* Physics 72(1), 268–277 (2008)
22. Neuner, M., Steininger, P., Mittendorfer, C., Sedlmayer, F., Deutschmann, H.: Automatic mask generation for 2D/3D image registration with clinical images of the pelvis. International Journal of Computer Assisted Radiology and Surgery 6 (Supplement 1), S54–S55 (6 2011)
23. Seroul, P., Sarrut, D.: VV: a viewer for the evaluation of 4D image registration. MIDAS Journal (Medical Image Computing and Computer-Assisted Intervention MICCAI'2008, Workshop - Systems and Architectures for Computer Assisted Interventions), 1–8 (2008), http://vv.creatis.insa-lyon.fr/
24. Shackleford, J., Kandasamy, N., Sharp, G.: On developing B-spline registration algorithms for multi-core processors. Physics in Medicine and Biology 55, 6329 (2010)
25. Shusharina, N., Sharp, G.: Analytic regularization for landmark-based image registration. Physics in Medicine and Biology 57(6), 1477 (2012), http://stacks.iop.org/0031-9155/57/i=6/a=1477
26. Siddon, R.: Fast calculation of the exact radiological path for a three-dimensional CT array. Medical Physics 12, 252 (1985)
27. Steininger, P., Neuner, M., Birkfellner, W., Gendrin, C., Mooslechner, M., Bloch, C., Pawiro, S., Sedlmayer, F., Deutschmann, H.: An ITK-based Implementation of the Stochastic Rank Correlation (SRC) Metric. Insight Journal (11 2010)

28. Steininger, P., Neuner, M., Mittendorfer, C., Scherer, P., Sedlmayer, F., Deutschmann, H.: Clinical implementation of gpu-accelerated n-way 2d/3d image registration for inter-fractional patient positioning in radiotherapy. International Journal of Computer Assisted Radiology and Surgery 6 (Supplement 1), S56–S57 (6 2011)

29. Steininger, P., Neuner, M., Weichenberger, H., Sharp, G., Winey, B., Kametriser, G., Sedlmayer, F., Deutschmann, H.: Auto-masked 2d/3d image registration and its validation with clinical cone-beam computed tomography. Physics in Medicine and Biology 57(13), 4277 (2012)

30. Thirion, J.: Image matching as a diffusion process: an analogy with Maxwell's demons. Medical image analysis 2(3), 243–260 (1998)