

A Novel Projection Based Approach for Medical Image Registration

Ali Khamene¹, Razvan Chisu², Wolfgang Wein², Nassir Navab², and Frank Sauer¹

¹ Imaging and Visualization Dept., Siemens Corporate Research,
755 College Road East, Princeton NJ 08540, USA

{ali.khamene, sauer.frank}@siemens.com

² Computer Aided Medical Procedures (CAMP) Group, TU Munich
Boltzmannstr. 3, 85748 Garching, Germany

{chisu,wein,navab}@cs.tum.edu

Abstract. In this paper, we propose a computationally efficient method for medical image registration. The centerpiece of the approach is to reduce the dimensions of each image via a projection operation. The two sequences of projection images corresponding to each image are used for estimating the registration parameters. Depending upon how the projection geometry is setup, the lower dimension registration problem can be parameterized and solved for a subset of parameters from the original problem. Computation of similarity metrics on the lower dimension projection images is significantly less complex than on the original volumetric images. Furthermore, depending on the type of projection operator used, one can achieve a better signal to noise ratio for the projection images than the original images. In order to further accelerate the process, we use Graphic Processing Units (GPUs) for generating projections of the volumetric data. We also perform the similarity computation on the graphics board, using a GPU with a programmable rendering pipeline. By doing that, we avoid transferring a large amount of data from graphics memory to system memory for computation. Furthermore, the performance of the more complex algorithms exploiting the graphics processor's capabilities is greatly improved. We evaluate the performance and the speed of the proposed projection based registration approach using various similarity measures and benchmark them against an SSE-accelerated CPU based implementation.

1 Introduction

Automatic image registration is nowadays an essential component in medical imaging systems. The basic goal of intensity based image registration techniques is to align anatomical structures in different modalities. This is done through an optimization process, which assesses image similarity and iteratively changes the transformation of one image with respect to the other, until an optimal alignment is found [1]. Computation speed is a critical issue and dictate applicability of the technology in practice. On the other hand feature based methods are computationally more efficient, however, they are notoriously dependant on the quality of the extracted features from the images [2].

In intensity based registration, volumes can directly be aligned by computing a volumetric similarity measure accessing the voxel intensities at each iteration. Since the amount of computations performed at each iteration is high, the overall registration process is very slow. In the cases, where Mutual Information (MI) is used, sparse sampling of volume intensity could reduce the computational complexity while compromising the accuracy [3, 4]. In [5], authors propose a projection based method for 2D-2D image registration. In this method, the projections along the two axes of the image are computed. Horizontal and vertical components of the shift is then computed using one-dimensional cross correlation based estimator. They show that the method is robust in the presence of temporal and spatial noise and computationally efficient compared to the 2D correlation based shift estimator. In [6], authors propose to formulate 3D-3D registration cost function as the summation of three 2D-3D optimization cost functions. The optimization is then done concurrently on the sum of the cost functions, which are identically parameterized. Furthermore, images are pre-processed to extract a binary segmentation. Projection images from the binary segmentation are used for computing the similarity measures.

Our proposed approach is a combination of the methods in [5] and [6]. We compute the projection images from two volumes and setup a cost function to register these images within a space, which is a subset of the space of the original rigid registration transformations. We perform these registrations successively for various projection geometries in order to estimate all the registration parameters of the original problem. We further optimize the performance of projection computation and 2D-2D registration similarity computation by using GPUs. We perform a validation study comparing the accuracy and the speed of the proposed method with a traditional volumetric 3D-3D MI-based approach.

2 Method

2.1 Proposed Registration Method

Conventional volumetric rigid registration algorithms, optimize 6 degrees of freedom (DOF) transformation parameters in order to maximize the volumetric similarity measure.

$$\widetilde{\mathbf{T}}^6 = \arg \max_{\mathbf{T}^6} \mathcal{S}^3(I_f, \mathcal{T}_{\mathbf{T}^6}^3(I_m)). \quad (1)$$

where \mathbf{T}^6 is a six DOF homogenous transformation matrix, \mathcal{T}^3 is the six DOF mapping operator, \mathcal{S}^3 estimates the similarity metric between two volumes, and I_f and I_m are the fixed and moving volumetric data, respectively. Let us define an orthographic projection operator \mathcal{P} , which projects the volume points onto an image plane using a projection matrix $\mathbf{\Pi}$. Without loss of generality, assuming that the volume world coordinates are located at the center of the fixed volume, any plane projection can be written as:

$$\mathbf{\Pi}_P = \mathbf{\Pi}_0 P, \quad (2)$$

where $\mathbf{\Pi}_0$ is a trivial 3×4 orthographic projection matrix with the principle axis along z and \mathbf{P} is a 4×4 homogenous transformation matrix encoding the principle axis of the orthographic projection matrix $\mathbf{\Pi}_P$. Since we only consider orthographic projections and assume that the center of the volume maps to the center of the plane of projection, the translation part of the matrix \mathbf{P} is zero. Once we compute the projection images using operator \mathcal{P}_P , which uses the projection matrix $\mathbf{\Pi}_P$, we can re-formulate the registration optimization as follows:

$$\widetilde{\mathbf{T}}_P^3 = \arg \max_{\mathbf{T}_P^3} \mathcal{S}^2 \left(\mathcal{P}_P(I_f), \mathcal{T}_{\mathbf{T}_P^3}^2(\mathcal{P}_P(I_m)) \right), \quad (3)$$

where \mathbf{T}_P^3 is a three DOF homogenous transformation matrix defined in the plane of projection specified by \mathbf{P} , \mathcal{T}^3 is a three DOF mapping operator, \mathcal{S}^2 computes the similarity metric between the 2D images. There exists an explicit relationship between the reduced dimension homogenous transformation matrix \mathbf{T}_P^3 and the original six DOF transformation matrix \mathbf{T}^6 . Let us formulate \mathbf{T}_P^3 as follows:

$$\mathbf{T}_P^3 = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & t_x \\ \sin(\theta) & \cos(\theta) & t_y \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

where θ is in-plane rotation and t_x and t_y are in-plane translations. It can easily be shown that:

$$\mathbf{T}^6 = \mathbf{P}^{-1} \mathbf{\Pi}_0^\top \mathbf{T}_P^3 \mathbf{\Pi}_0 \mathbf{P} \quad (5)$$

where \top depicts the transpose operation. From equation 5, it is apparent that only part of the space represented by \mathbf{T}^6 is covered by the in-plane transformation from \mathbf{T}^3 . And that depends on the projection transformation in \mathbf{P} . In-plane translations from \mathbf{T}^3 are translations along the first two axes of the transformed coordinate system by \mathbf{P} and the in-plane rotation is the rotation about the third axis. In order to cover the whole space of 3D rigid transformation, one easy solution is to consider three projections, as follows:

$$\begin{aligned} \mathbf{\Pi}_{P_0} &= \mathbf{\Pi}_0 \mathbf{P}_0 = \mathbf{\Pi}_0 \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{r}_3 & \mathbf{0} \\ \mathbf{0}^\top & & & 1 \end{bmatrix} \\ \mathbf{\Pi}_{P_1} &= \mathbf{\Pi}_0 \mathbf{P}_1 = \mathbf{\Pi}_0 \begin{bmatrix} \mathbf{r}_2 & \mathbf{r}_3 & \mathbf{r}_1 & \mathbf{0} \\ \mathbf{0}^\top & & & 1 \end{bmatrix} \\ \mathbf{\Pi}_{P_2} &= \mathbf{\Pi}_0 \mathbf{P}_2 = \mathbf{\Pi}_0 \begin{bmatrix} \mathbf{r}_3 & \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{0} \\ \mathbf{0}^\top & & & 1 \end{bmatrix} \end{aligned} \quad (6)$$

where $\mathbf{0}$ is vector of zero and \mathbf{r}_i for $i \in [1 \ 3]$ are columns of the rotation matrix embedded in the corresponding homogenous transformation. Finally the successive optimization approach is outlined in Algorithm 1.

Regardless of how computationally expensive a similarity measure estimation is, a volumetric registration method requires $k * N^3$ operations for an optimization loop with k iterations, where N is the number of voxels. For identical number

```

precompute the projections of the fixed volume using  $\mathbf{\Pi}_{P_0}$  to  $\mathbf{\Pi}_{P_2}$ 
registration transformation is set to identity
while there is a significant incremental change in registration parameters do
    successively choose a projection matrix from  $\mathbf{\Pi}_{P_0}$  to  $\mathbf{\Pi}_{P_2}$ 
    compute the projection of the transformed moving volume
    perform registration between the corresponding projection images as in
    equation 3
    use the equation 5 to update the registration transformation
end

```

Algorithm 1: The proposed registration approach.

of iterations, assuming two rounds of registrations per projection, the number of operations for the proposed method is $2 * N^3 + k * N^2$. Furthermore, the projection operation in this case can be performed using a graphics processing unit, as it is explained in the following section.

2.2 Implementation

The computation performance offered by today's video boards by far surpasses that of currently available CPUs - while a Pentium 4 3GHz CPU can theoretically reach 6 GFLOPS³, synthetic benchmarks have shown the NVIDIA Geforce 6800 Ultra GPU to reach 40 GFLOPS. This fact, together with the inherently parallel architecture of graphics processors, has made the approach of GPU programming highly attractive for accelerating algorithms in different domains. For instance, robot motion planning [7], flow visualization [8], segmentation [9], and solving sets of algebraic equations [10] are proposed to be implemented using GPUs. In the domain of medical image registration, GPUs have been mainly used to speed up the generation of Digitally Reconstructed Radiographs (DRRs) using hardware-accelerated volume rendering techniques. Our aim is now to implement the proposed algorithm including the computation of image similarity using a GPU.

Graphics processors are highly specialized for a single application - handling three-dimensional geometric data, project it onto the two-dimensional computer display and apply a range of visual effects, like texturing, lighting and shading. There are two major steps in the rendering process. In the vertex shading step vertex-specific data like normals or texture coordinates are manipulated. Whereas, in the fragment shading step pixel-specific data like texturing or blending are processed. Both these steps can be customized by loading a so-called *shading-program* or *shader* onto the GPU. A different program is needed for each of these two steps. In the process of computing similarity measures, the major part, i.e., examining the image intensities, has to be done for each of the image pixels. This part is implemented as a fragment shader. However, the computation of the texture coordinate increments that have to be added to the current coordinate, in order to examine the pixel's neighborhood, can

³ Billions of floating-point operations per second

also be implemented within the vertex shading unit. Aside from these, all the other processes taking place in the rendering pipeline (e.g., rasterization) must be regarded as computational overhead. Therefore, simple similarity measures perform rather poorly in comparison to a classical CPU implementation, while for the more complex measures, i.e., the ones requiring a fragment shader with a high number of operations, the pipeline’s overhead becomes less significant. Our results show that, depending on the measure’s complexity, the GPU-based computation can be from three times slower to ten times faster than a non-accelerated CPU-based implementation.

The specialized nature of the GPU environment imposes several technical restrictions that do not apply to classic CPU programming. For example, the pipelined structure of the GPU implies that data can flow only in one direction, which means that the respective programs cannot modify their input values, nor can they read from the output buffers. This means that information-theoretic similarity measures like MI cannot be implemented. Furthermore, the pixels rendered onto the screen are usually represented as positive 8 bit RGBA color values. Thus in each rendering pass, up to four values can be computed for each pair of pixels in the images to be registered. Similarity measures requiring more values accordingly have to be split up into multiple rendering passes. Finally, with respect to the precision of the computed color data, 16 and 32 bit floating-point color buffers are also supported by the more modern video boards. However, these color buffers are always off-screen rendering buffers that cannot be directly displayed, and are both more difficult to use and impose a drop in performance. However, in the results section, we show that using a low-accuracy 8 bit color buffer can result in a registration just as accurate as using 32 bit floating point data on the CPU. Because 8 bit color intensities can take only positive values, any negative number automatically gets clamped to zero. As some of the measures we compute, and also the image gradients, yield negative values, we decided to use two color channels to store the value by employing the following scheme: $R = x, G = -x \Rightarrow x = R - G$.

The similarity of two images must be provided as a scalar value, e.g. the sum or average of all pixels, in order to represent a cost function to be optimized. However, summing up the color values in the frame buffer or within a texture is not a typical graphics application. In order to avoid the slow copying of the computed image from the GPU to the main RAM for averaging on the CPU, we use the mipmapping capability of the video card. Mipmaps represent successively smaller versions of an initial texture, having half the width and height of the previous mipmap. They are obtained by taking four neighbored pixels from the input image and averaging them into one pixel in the output image. The main problem of mipmaps arises from the fact that the repeated summation and averaging of 8 bit data (e.g., 8 times for a 256×256 image) is followed by a truncation/rounding to 8 bits. These rounding errors can potentially affect the registration accuracy. To work around this problem, we use a hybrid averaging approach. Instead of using the last mipmap consisting in a single pixel, we generate the 4th mipmap, which is 16×16 pixels in the case of 256×256 images,

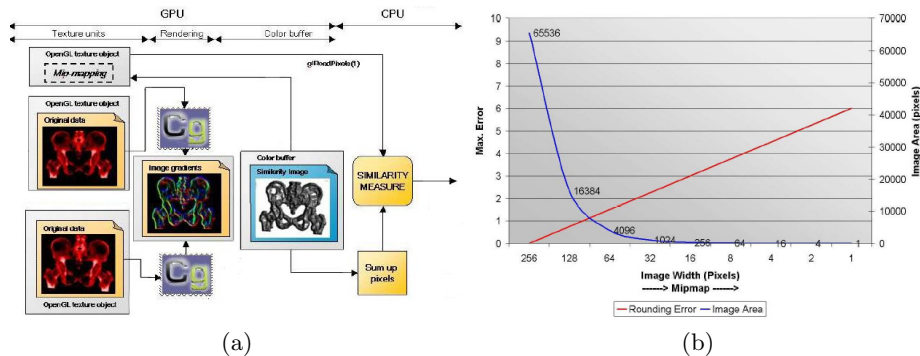


Fig. 1. (a) Similarity measure computation process on GPU, and (b) depicts the rounding error and image size with respect to mipmapping level.

and then copy this very small image to the main memory for CPU averaging. By doing this, we significantly reduce the amount of data that has to be transferred from video to system memory without introducing more rounding errors than necessary, resulting in a good compromise between speed and accuracy.

We implemented ten similarity measures of different complexity: Sum of Squared Differences (SSD)[11], Sum of Absolute Differences (SAD), Correlation Coefficient (CC)[11], Ratio of Image Uniformity (RIU)[12], Pattern Intensity (PI)[13], Gradient Correlation (GC), Gradient Difference (GD)[14], Gradient Proximity (GP), Sum of Local Normalized Correlation (SLNC)[15] and Variance-Weighted Sum of Local Normalized Correlation (VWC)[15]. In order to compute the projection images, we used a 3D texture based volume rendering technique. Both intensity accumulation or maximum intensity projection (MIP) techniques can be used for generating the projection images. Vector valued projection images can also be derived from the volumes. These images provides much richer information regarding the volume at each specific projection. The proposed method can be used along with all the various volume projection options.

3 Results

We validated our methods by performing intensity-based registration on two data sets. The first comprised two CT scans of a patient’s pelvis, before and after treatment, respectively. The second data was a CT scan of a megavolt cone-beam CT volume of a skull phantom. The standard registration approach, fully sampling the moving volume using the MI similarity metric, yielded very stable results and was thus used as ground truth. A registration run took several minutes using this method. A cube with 10cm width placed at the volume center was used for computing the Target Registration Error (TRE). Table 1 lists the computation times for computing the respective similarity measures for 256×256 pixel 8 bit grayscale images on the CPU with/without SSE-II acceleration, and

Table 1. Similarity Measure computation times

Measure	CPU	CPU, SSE	GPU, mipmaps	GPU, CPU	GPU, hybrid
SAD	0.17ms	0.05ms	0.54ms	3.52ms	0.60ms
SSD	0.17ms	0.05ms	0.54ms	3.54ms	0.59ms
RIU	1.20ms	0.32ms	0.55ms	3.56ms	0.62ms
NCC	0.87ms	0.17ms	0.53ms	3.56ms	0.59ms
GC	2.46ms	0.55ms	1.15ms	7.12ms	1.29ms
GD	2.86ms	0.70ms	0.73ms	3.71ms	0.76ms
GP	1.82ms	0.45ms	0.72ms	3.70ms	0.75ms
PI	14.83ms	7.93ms	1.21ms	4.20ms	1.27ms
LNC	18.32ms	6.69ms	3.14ms	6.11ms	3.17ms
VWC	18.35ms	6.59ms	1.78ms	4.77ms	1.81ms

Table 2. Registration results for pelvis CT

	GPU_{mm1}	GPU_{mm2}	GPU_{mm3}	GPU_{CPU1}	GPU_{CPU2}	GPU_{CPU3}
Measure	GD	SAD	PI	PI	GD	RIU
TRE [mm]	1.57	2.29	2.86	0.95	0.98	1.02
$\sigma(TRE)$	0.60	0.83	4.3	0.22	0.31	0.21
t [s]	2.4	2.1	2.8	7.04	5.92	5.26

	GPU_{hyb2}	GPU_{hyb3}	CPU_{SSE1}	CPU_{SSE2}	CPU_{SSE3}
Measure	PI	GD	RIU	GD	PI
TRE [mm]	0.9	0.98	1.04	0.82	0.98
$\sigma(TRE)$	0.23	0.31	0.23	0.25	0.30
t [s]	4.00	3.5	2.9	6.76	13.02

on the GPU with mipmap and/or CPU-based averaging. These benchmarks were executed on an Intel Pentium4 2.4 Ghz system with an NVidia GeForce 6800GT graphics accelerator.

For validating our implementation, we used two pairs of volumetric data sets. We will present the three most accurate measures for four implementations: GPU-based using mipmaps for averaging (GPU_{mm}), GPU-based using the CPU for averaging (GPU_{CPU}), GPU-based using a combination of mipmaps and the CPU for averaging (GPU_{hyb}) and CPU-based with SSE (CPU_{SSE}) acceleration. First set of experiments are done on two CT scans of the same patient’s pelvis taken at different points of time. The image intensity and contrast differed slightly, with some features being visible in only one of the volumes.

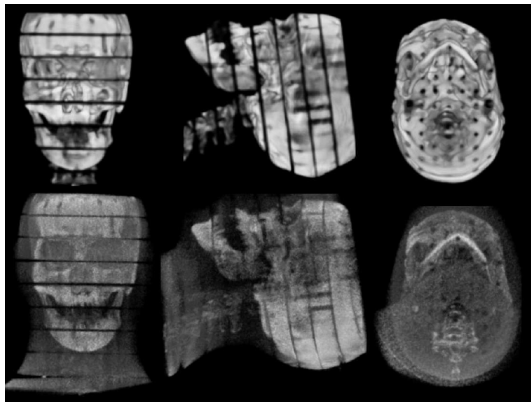
For the second run we used scans of a skull acquired with different types of CT scanners. The fixed volume was acquired using a regular CT scanner, while a megavolt cone-beam CT was used for the moving volume. Thus, the two volumes differed quite strongly with respect to brightness and contrast, and some features, i. e. skin, were visible in one volume but not in the other.

The results from tables 2 and 3 illustrate that using modern graphics hardware for similarity measure computation can significantly accelerate the process

Table 3. Registration results for skull CT / MV-CT

	GPU_{mm1}	GPU_{mm2}	GPU_{mm3}	GPU_{CPU1}	GPU_{CPU2}	GPU_{CPU3}
Measure	SAD	SSD	RIU	SSD	NCC	SAD
TRE [mm]	2.32	3.27	5.65	1.57	1.65	1.67
$\sigma(TRE)$	0.71	1.94	3.58	0.09	0.18	0.04
t [s]	2.34	2.27	2.59	5.35	5.38	5.29

	GPU_{hyb2}	GPU_{hyb2}	GPU_{hyb3}	CPU_{SSE1}	CPU_{SSE2}	CPU_{SSE3}
Measure	SSD	NCC	SAD	RIU	NCC	SSD
TRE [mm]	1.68	1.85	1.86	1.37	1.52	1.54
$\sigma(TRE)$	0.35	0.50	0.17	0.23	0.10	0.09
t [s]	2.99	2.98	2.99	7.19	6.21	6.49

**Fig. 2.** First row depicts the projection images from a CT volume, second row shows the corresponding images from Mega-Voltage CT volume.

of medical image registration. A hybrid approach, as explained in the previous section, yields results that are as accurate as a CPU-based implementation, in as much as a quarter of the time. The experiments also show that some similarity measures, namely Pattern Intensity and Gradient Difference, deliver very accurate registration even when mipmaps are used for completely summing up the resulting pixels. When registering data sets acquired by using different CT energy levels, similarity measures examining spatial information instead of just the intensity at the individual pixel location deliver very poor results. Still, measures examining pixel intensities alone, especially the Sum of Absolute Differences, yield a very accurate registration. The centerpiece of the approach, which enabled using the GPUs conveniently for registration of volumetric data has been the projection based method.

4 Summary

In this paper, we presented a projection based approach for registration of medical images. Successive registration of two dimensional projection images can yield an accurate volumetric registration. The dimension reduction property of this method, enables convenient usage of GPU for both computing the projections and furthermore estimating the two dimensional similarity images. A series of comparative studies are presented to demonstrate both the accuracy and computational efficiency of the proposed method.

References

1. Maintz, J., Viergever, M.: A survey of medical image registration. *Medical Image Analysis* **2** (1998) 1–36
2. Hill, D.L.G., Hawkes, D.J., et. al.: Registration of MR and CT images for skull base surgery using pointlike anatomical features. *British Journal of Radiology* **64** (1991)
3. Colignon, A., et al.: Automated multi-modality image registration based on information theory, *IPMI* (1995) 263–274
4. Wells, W., Viola, P., et al.: Multi-modal volume registration by maximization of mutual information. *Medical Image Analysis* **1** (1996) 32–52
5. Cain, S.C., Hayat, M.M., Armstrong, E.E.: Projection-based image registration in the presence of fixed-pattern noise. *IEEE Transactions on Image Processing* **10** (2001) 1860–1872
6. Chan, H., Chung, A.C.S.: Efficient 3D-3D vascular registration based on multiple orthogonal 2D projections, *Second International Workshop on Biomedical Image Registration (WBIR)* (2003) 301–310
7. Lengyel, J., Reichert, M., Donald, B.R., Greenberg, D.: Real-time robot motion planning using rasterizing computer graphics hardware, *SIGGRAPH* (1990) 327–335
8. Weiskopf, D., Hopf, M., Ertl, T.: Hardwareaccelerated visualization of time-varying 2D and 3D vector fields by texture advection via programmable per-pixel operations, *VMV* (2001) 439–446
9. Rumpf, M., Strzodka, R.: Level segmentation in graphics hardware. Volume 3., *ICIP* (2001) 1103–1106
10. Krüger, J., Westermann, R.: Linear algebra operators for GPU implementation of numerical algorithms, *SIGGRAPH* (2003) 1–9
11. Hajnal, J.V., Hill, D.L., Hawkes, D.J.: *Medical Image Registration*. CRC Press (2001)
12. Woods, R., Grafton, S., Holmes, C., Cherry, S., Mazziotta, J.: Automated image registration: I. general methods and intrasubject, intramodality validation. *Journal of Computer Assisted Tomography* **16** (1992) 620–633
13. Weese, J., Buzug, T., Lorenz, C., Fassnacht, C.: An approach to 2D/3D registration of a vertebra in 2D X-ray fluoroscopies with 3D CT images. In: *Processing CVMed/MRCAS*. (1997) 119–128
14. Penney, G.P., Weese, J., Little, J.A., Desmedt, P., Hill, D.L.G., Hawkes, D.J.: A comparison of similarity measures for use in 2D-3D medical image registration. *IEEE Transactions on Medical Imaging* (1998) 586–595
15. LaRose, D.: *Iterative X-ray/CT Registration Using Accelerated Volume Rendering*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA (2001)