MAGISTERARBEIT

# Void Tracking in SiC Particle Reinforced Al

ausgeführt am Institut für Computergraphik und Algorithmen
der Technischen Universität Wien

unter der Anleitung von
Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Eduard Gröller
Institut für Computergraphik und Algorithmen

und

O.Univ.Prof. Dipl.-Ing. Dr.techn. Hans-Peter Degischer
Institut für Werkstoffwissenschaft und Werkstofftechnologie

sowie

Dipl.-Ing. Michael Schöbel
Institut für Werkstoffwissenschaft und Werkstofftechnologie
als verantwortlich mitwirkendem Universitätsassistenten

durch
Wolfgang Altendorfer, Bakk. techn.
Matr.-Nr. 0104746
96 Ard Keale, Mount Oval Village
Rochestown, Co. Cork
Ireland

Wien, am 25. März 2008                    . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

# Contents

# Abstract

The expansion behavior of particulate reinforced AlSiC metal matrix composites (MMCs) during a thermal cycle shows anomalies at temperatures above 250℃, explained by visco-plastic matrix deformations and changing void volume fractions. Volumetric images of the sample acquired by synchrotron tomography at several temperatures between 30℃ and 400℃ are used to analyze the void kinetics. These high-resolution low-contrast $3D + T$ images are not aligned, thus only allowing quantitative comparisons over areas comprising many voids. Analysis of specific voids at various temperatures was not feasible, due to the three-dimensional misalignment.

We present a 3D registration approach based on mutual information that allows for sub-pixel accurate registration. A maximum intensity projection (MIP) may be used for a manual pre-alignment. Further software tools for the analysis of void kinetics of AlSiC provide linked views for 2D slices, 3D isosurfaces, volume rendering, as well as overlaid contours (isolines) of multiple images on an arbitrarily positionable cutting plane, and the calculation of a shape factor. By registering the volumetric datasets, we were able to visualize the shrinkage and expansion of a specific void during thermal cycling between room temperature and 400℃ and to analyze the change of quantitative properties of specific voids–including volume, surface, shape factor—resulting from temperature changes.

2

# Kurzfassung

Das thermische Ausdehnungsverhalten von AlSiC, einem SiC-partikelverstärkten Aluminium-Matrix Verbundwerkstoff, zeigt Anomalien bei Temperaturen über 250°C. Diese erklären sich durch reversible elasto-plastische Verformung der Matrix in Kombination mit Porosität. Volumetrische Bilddaten wurden durch Synchrotron-Tomographie während eines thermischen Zyklus zwischen 30°C und 400°C aufgenommen, um das Verhalten der Poren zu analysieren. Diese hochauflösenden $3D + T$ Bilder sind nicht aufeinander ausgerichtet, und erlauben dadurch nur einen quantitativen Vergleich über einen größeren Bereich mit einer Vielzahl an Poren. Der direkte Vergleich spezifischer Poren bei bestimmten Temperaturen war dadurch nicht möglich.

Diese Arbeit präsentiert einen Ansatz zur 3D Bildregistrierung mittels Mutual Information, der subpixel-genaue Registrierung ermöglicht. Unterstützend ist eine manuelle Vorausrichtung basierend auf einer Maximum-Intensitäts-Projektion (MIP) möglich. Die entwickelte Software ermöglicht außerdem die Untersuchung der Porenkinetik durch gekoppelte Ansichten von 2D Schichten, 3D Oberflächen, Volumenrendering, sowie den Vergleich von Poren mittels überlagerter Konturen auf einer frei positionierbaren Schnittfläche und die Berechnung eines Formfaktors. Durch die Registrierung konnten wir das Wachstum und das Schrumpfen einzelner Poren während eines Temperaturzyklus zwischen Raumtemperatur und 400°C visualisieren, sowie die Veränderung quantitativer Eigenschaften einzelner Poren bei Temperaturveränderungen anhand von Volumen, Oberfläche und Formfaktor zeigen.

# 1 Introduction

> The purpose of computing is insight,
> not numbers.
>
> _____
>
> RICHARD HAMMING

This thesis emerged from a cooperation of the *work group for fundamentals of materials science* of the *Institute of Materials Science and Technology* and the *computer graphics group* of the *Institute of Computer Graphics and Algorithms* at the *Vienna University of Technology.* Computer science has only a short history as a formal academic discipline, but exhibits a growing influence on other sciences. Cutting edge technology in materials science nowadays mostly involves the application of computer systems for data acquisition and analysis. Three dimensional (*3D*) imaging techniques including X-ray and synchrotron tomography lead to new insights on the micro-structural features of materials, but also produce vast amounts of data. The field of computer graphics provides the means for adequate preparation and visualization of these datasets.

The aim of this work is to provide the materials scientists with a set of tools for the analysis of a series of 3D images of *AlSi7Mg/SiC/70p* (AlSiC). AlSiC is a 70 vol.-% SiC particle reinforced aluminum-silicon alloy matrix composite material used for heatsinks and as base plates in components for high power electronics like *Insulated Gate Bipolar Transistors* (IGBT). It combines a high thermal conductivity with a low *coefficient of thermal expansion* (CTE) which reduces the thermally induced stresses compared to copper when soldered to a ceramic IGBT substrate. It shall ease the detection of micro-structural changes during a heating cycle, in particular the behavior of voids as the porosity changes.

## 1.1 Motivation

The *Institute of Materials Science and Technology* evaluated the residual stresses and micro voids in AlSiC by subjecting it to multiple thermal cycles between +30°C and

+400°C. At several temperatures 3D images were acquired by means of synchrotron tomography. This revealed the existence of microscopic voids of about 0.14 vol.-% at room temperature [SRD06]. During heating a change of porosity was observed. The thermal expansion curves show an anomaly that can not be explained by thermo-elastic models [Hub03]. The reason for this is expected to be the closure of the microvoids by diffusion of the aluminum matrix above 200°C, and opening of the voids during cooling.

The acquired high resolution 3D images have a voxel size of 1.6 $\mu m$. The void volume fraction has been analyzed over a gauge volume of $\approx 0.5 mm^3$ during heating and cooling. As the sample is shifted (a translative movement combined with a small rotatory movement) between consecutive images from a single heating cycle, the comparison of exactly the same microscopic structure in consecutive images was impossible. Manually aligning the datasets was not feasible, as it needs a tremendous amount of work to find and align corresponding particles in consecutive images due to the low contrast and repetitive indistinguishable patterns.

By comparing the images on a macroscopic scale it was not possible to observe the behavior of individual voids during temperature cycling. Therefore it was not clear where exactly at the interface between SiC and Al voids open: if they occur (predominantly) at previous locations of voids, or if voids open particularly at angles of adjacent SiC particles when the aluminum retreats.

## 1.2 Objectives

A major purpose of this work is to provide a software that is able to align the 3D images. This task is called 3D image registration. By registering the volumetric images it will be possible to find the same void in various images. When the images are resampled using the transformation obtained by the registration it will be possible to find the location of a specific void by looking at the same coordinates in each image.

Further software applications will help to analyze the behavior of a void in multiple images. Linked views enable a simple one-to-one comparison of registered slices and 3D renderings of a void. The possibility to superimpose the contour of a void at various temperatures will give further insight into the void kinetics during a thermal cycle.

## 1.3 Structure of the Thesis

The thesis is structured as follows:

A detailed description of the composition of the sample material and information about the image acquisition can be found in *Chapter 2*. *Chapter 3* provides an overview in established methods which will be used in this work. The problem we are trying to solve in this work is elaborated in more detail in *Chapter 4*.

*Chapter 5* presents the concepts used, such as details on the registration method and parameters of its actual implementation. The achievements of this work are presented in *Chapter 6* and discussed in *Chapter 7*. A conclusion follows in *Chapter 8*, summarizing the key results of the presented work and giving an outlook on what can be expected from future research in this area. Finally, the thesis ends with a comprehensive summary in *Chapter 9*.

# 2 Data Acquisition

> Don't ask what it means, but rather
> how it is used.
>
> ———————————————
> LUDWIG WITTGENSTEIN

## 2.1 Material of the Sample: AlSiC

*Silicon carbide particulate reinforced aluminum matrix composites* have a high potential
for advanced engineering applications. AlSiC has particularly improved properties for
heatsinks compared to copper regarding the application as base plates in high power elec-
tronics. Its thermal expansion behavior is very similar to that of the ceramic substrate
which it is mounted on, thus reducing the stresses on the solder joints.

### 2.1.1 Metal Matrix Composite

The composite approach enables significant and tailored alterations in the physical prop-
erties of metallic materials [Mor]. *Metal Matrix Composites* (MMC) consist of a con-
tiguous metallic matrix in which an additional material is embedded. The *reinforcement
material* may be continuous fibers, short fibers, or particles of rigid material, generally
ceramic [CW93]. In the investigated AlSiC, *silicon carbide*(SiC) particles with a tri-
modal particle size distribution are embedded into an *AlSi7Mg* matrix. Aluminum is a
light metal with a good thermal conductivity. Silicon carbide is a ceramic compound of
silicon and carbon with a low *coefficient of thermal expansion* (CTE) and a high thermal
conductivity. The combination of these constituents leads to a material with a low CTE
and a high thermal conductivity [Hub03]. AlSiC is produced by infiltrating SiC particle
preforms of 30% porosity with the liquid Al-alloy under high pressure (Figure 2.1 and
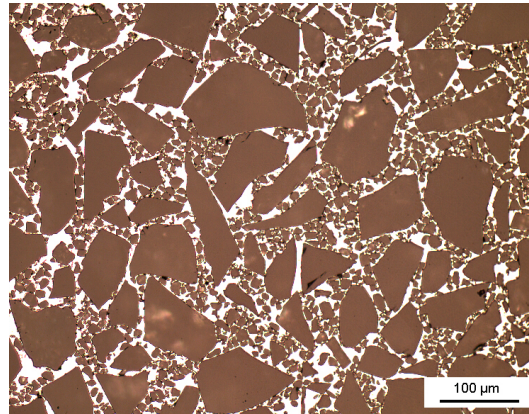2.2).

Figure 2.1: Light optical micrograph of AlSiC shows the trimodal particle distribution with a few dark voids in the bright matrix [SRD$^+$07b] (image from Huber [Hub03]).
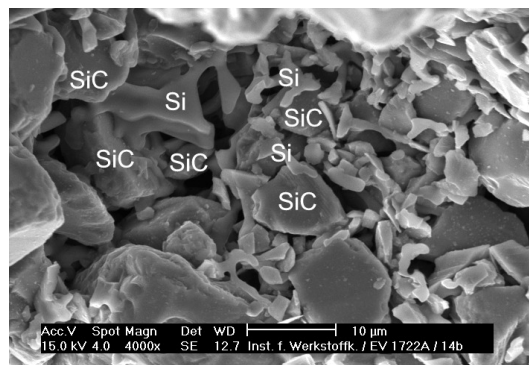


Figure 2.2: Scanning Electron Microscopy (SEM) micrograph with partly dissolved Al-matrix of the AlSiC-composite showing Si lamella connecting SiC particles (image from Huber [Hub03]).

## 2.1.2 Production

The full designation of the MMC is AlSi7Mg/SiC/70p. It has been produced by gas pressure infiltration with 70 vol.-% trimodal SiC-particle preforms by Electrovac GmbH, Klosterneuburg, Austria. 55 vol.-% of the particles are of the FEPA [FoEPoA] F100 size class (149–105 $\mu m$), the others F600 and smaller than 5 $\mu m$. See Huber [Hub03] for a detailed analysis of the thermal expansion behavior of various aluminum-matrix composites. The shrinkage of Al during cooling after infiltration leads to plastic deformations in the matrix. 'Small voids in the MMC could be observed which are formed during cooling after the infiltration process' [SRD$^+$07b].

Figure 2.3: A detail from a single slice of an AlSiC tomogram (voxel size: $(1.6\mu m)^3$).

## 2.1.3 Application

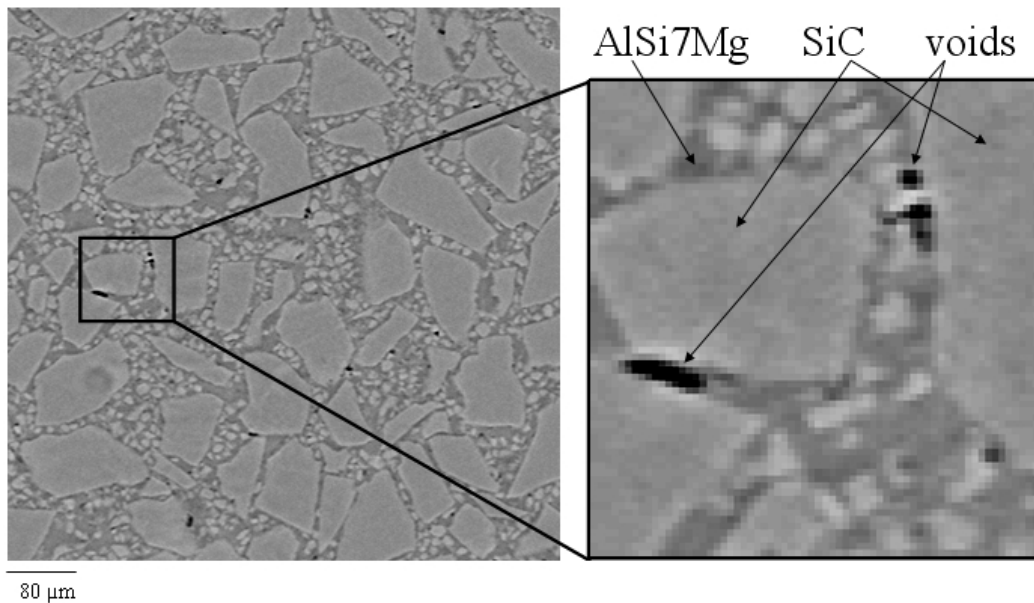Schöbel et al. [SRD$^+$07a] describe the application of AlSiC as follows:

> 'AlSi7Mg/SiC/70p (AlSiC) is used for heat sinks because of its good thermal conductivity combined with a low coefficient of thermal expansion (CTE). These properties are important for power electronic devices where heat sinks have to provide efficient heat transfer to a cooling device. A low CTE is essential for a good surface bonding of the heat sink material to the isolating ceramics. Otherwise mismatch in thermal expansion would lead to damage of the bonding degrading the thermal contact within the electronic package. Therefore AlSiC replaces increasingly copper heat sinks. The CTE mismatch between isolation and a conventional metallic heat sink is transferred into the MMC heat sink. The stability of the interface bonding within a MMC is critical for its thermal properties.'

*Silicon carbide particulate reinforced aluminum matrix composites* have a high potential as advanced engineering materials being developed for use in traction, automotive, aviation and space applications [GLD03]. Figures 2.4 and 2.5 show examples for the application of AlSiC as base plates in IGBTs.
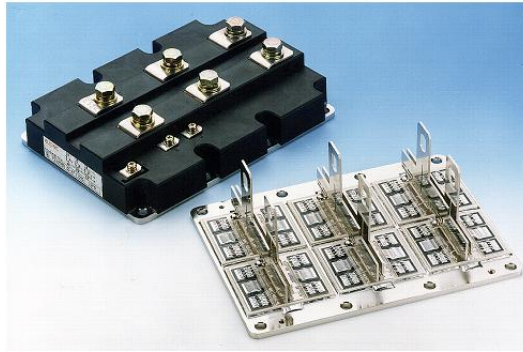
Figure 2.4: Ready-packed and semi-finished module consisting of baseplate onto which the ceramic substrate chips and terminals have been soldered [LLS+00] (image from Huber [Hub03]).
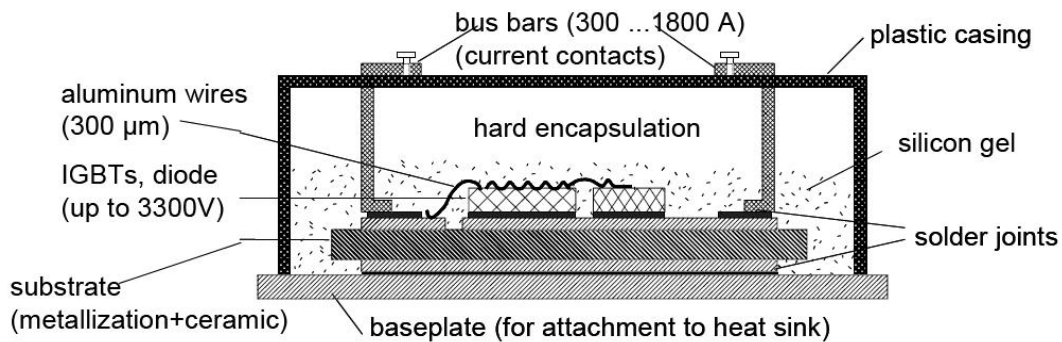


Figure 2.5: The design of IGBT power modules and the main functions of the components [LLS+00] (image from Huber [Hub03]). Baseplates are made of AlSiC.

## 2.2 Image Acquisition

3D images of the sample were acquired prior to this work by the *Institute of Materials Science and Technology* by means of *synchrotron tomography* on beamline ID15A at the ESRF Grenoble. The basic principle of synchrotron X-ray micro-tomography is similar to *computed tomography* (CT) in medical imaging, but *synchrotron tomography* is able to provide a finer resolution—in the range of micrometers—by using a brighter and more focused monochromatic X-ray beam with advanced high resolution CCD cameras. A slice of a tomogram is shown in Figure 2.3.

### 2.2.1 Synchrotron Tomography

Computed tomography is an imaging method that reconstructs a 3D volume from a series of 2D projections. Contrary to the medical imaging CT scanners, where the X-ray

tube rotates around the patient and generates a series of projections, for synchrotron tomography the sample itself is rotated. By using filtered back-projection a volumetric dataset is reconstructed from the measured X-ray projections. The resulting 3D dataset consists of a series of 2D greyscale images, commonly referred to as *slices*. The intensity of a *volumetric pixel* (voxel) denotes the attenuation of the X-ray beam at this position.
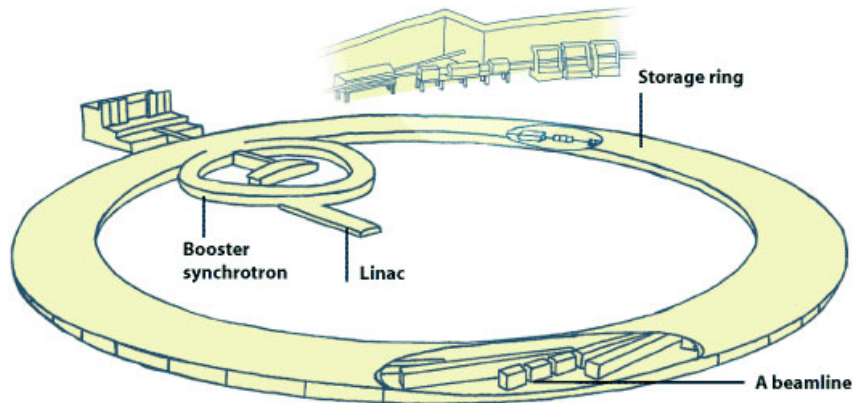


Figure 2.6: Electrons emitted by an electron gun are first accelerated in a linear accelerator (linac) and then transmitted to a circular accelerator (booster synchrotron) where they are accelerated to reach an energy level of 6 billion electron-volts (6 GeV). These high-energy electrons are then injected into a large storage ring—844 meters in circumference—where they circulate in a vacuum environment, at a constant energy, for many hours [ESR] (image from ESRF [ESR]).

Synchrotron tomography can be compared to 'supermicroscopes'. 'An X-ray beam produced by a synchrotron is a thousand billion times brighter than the beam produced by a hospital X-ray machine' [ESR]. There are about 50 synchrotrons in use worldwide, ESRF is one of the brightest at high energies.

## 2.2.2 Tomography at ESRF

Synchrotron tomography was done at the *European Synchrotron Radiation Facility* (ESRF) in Grenoble, France. The ESRF is a joint facility supported and shared by 18 European countries and operates the most powerful synchrotron radiation source in Europe, built in 1988. Figure 2.6 provides a schematic overview of the ESRF. The tomography setup is depicted in Figure 2.7 and Figure 2.8. A schematic of the high resolution imaging

detector is depicted in Figure 2.9. Experiments were carried out on the *ID15A High Energy Scattering Beamline*. The High Energy Scattering Beamlines ID15 offer a high flux of highly collimated photons in the energy range from about 30 keV to 1000 keV.
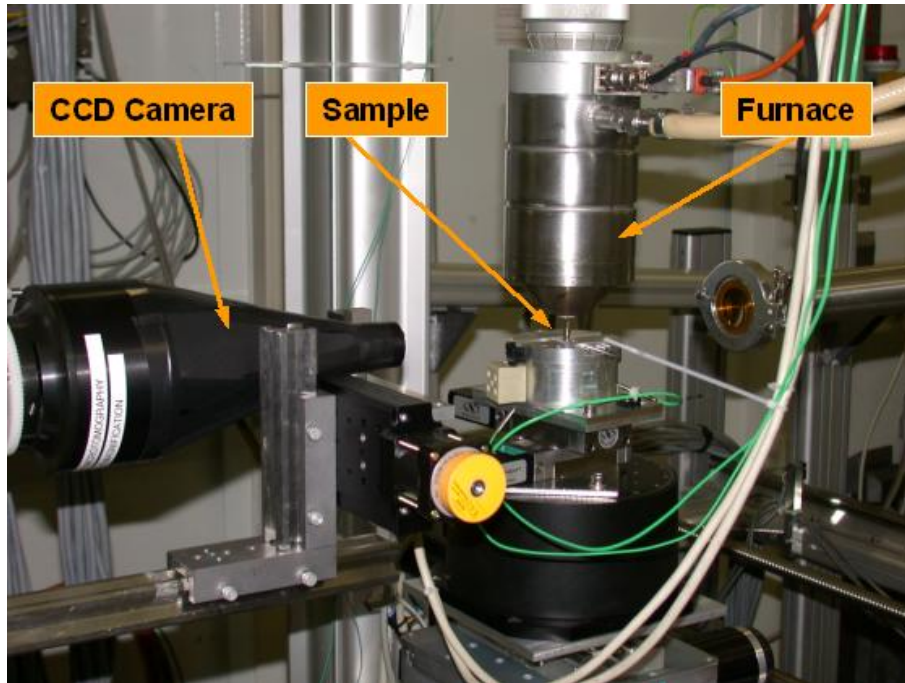


Figure 2.7: Synchrotron tomography setup with a resistance heating of the sample.

**Defects in reconstructed images**

From the series of 2D projections the 3D volumetric dataset is reconstructed. The resulting image exhibits noise and artefacts (Figure 2.10). An artefact is 'a feature or appearance that is seen on an image, which does not actually exist' [JT04]. Errors in the projection propagate back into the reconstructed image and can be seen as streaking (beam-hardening), shading, rings, etc.

**Tomogram Data**

Reconstructed tomogram data was received from the ID15A staff of the *ESRF*. The volume data is stored as one slice per file. One volume consists of 750 slices of approximately 1024 x 1024 pixels. The voxel size is $1.6^3 \mu m$. Actual dimensions vary slightly between tomograms. The file is a binary file, it has 1024 Bytes header, and is saved as

Figure 2.8: Schematic view of the x-ray micro-tomography experiment setup at the ESRF ID15A beamline. The radiation source is a 7-pole asymmetric wiggler. The x-ray white spectrum is adapted to the different samples by changing the wiggler magnetic field as well as by means of filters (image from Reimann et al.[RPMF05]).

32 bit float values (little endian byte order). One slice needs about 4 MB of diskspace, which accumulates to about 2.8 GB per volume.

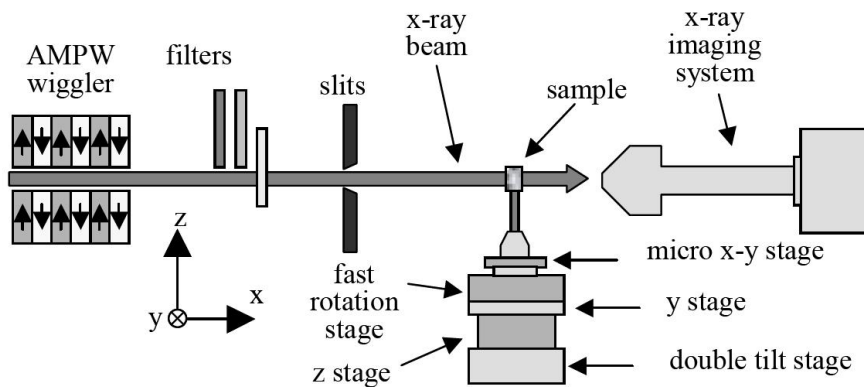The gray level histogram of the source files has a range of $-26$ to $+42$, with a mode at approximately 1.0, the intensity where most of the material is found (Aluminum and Silicon-Carbide). The voids reside in the low intensity region. The volume fraction of the voids is about 0.14% in as-cast condition. The high intensity region represents high absorbing Fe containing impurities, which are the heavier elements with an higher absorption coefficient.

**Clipping and Conversion** As the original files with 2.8 GB could not be loaded into the RAM of our 32-bit computers running MS Windows XP the volumes were clipped and converted to 8 bit for easier handling. From the reconstructed binary images the non-sample areas and the border of the sample were removed, and the resulting image stored as an 8 bit raw-file. Working with raw-files is error-prone as they do not have an image header containing information about the data and manual input is required. A 3D-file format containing additional information—the VTK file format [Gui]—was therefore preferred.
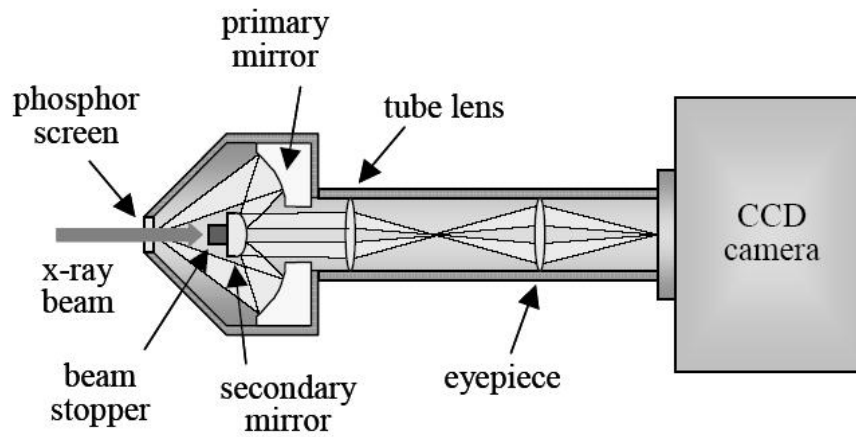
Figure 2.9: The high resolution imaging detector on beamline ID15A (image from Reimann et al. [RPMF05]).
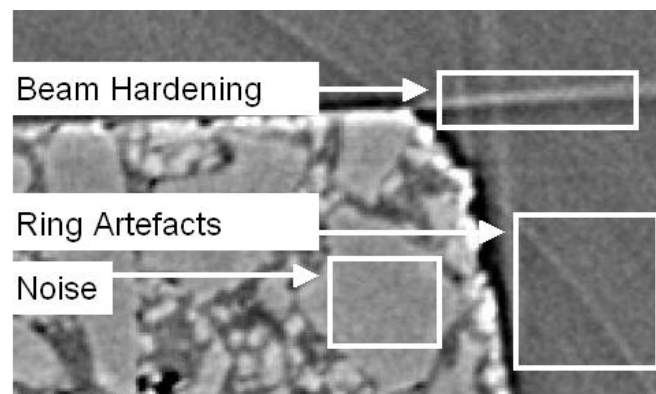


Figure 2.10: Visual reconstruction artefacts of an AlSiC sample (contrast enhanced).

# 3 State of the Art in Volume Registration and Visualization

## 3.1 Microtomography

Microtomography is a noninvasive imaging technique, based on the invention of X-ray computed tomography (CT) by G.N. Hounsfield [Hou73] [HA73]. Besides its application in medical imaging [Amb73] [LCLT74], it has been used extensively 'in nondestructive testing (NDT) for many industrial applications such as mechanical part manufacturing, production of composite materials, waste container inspection and even hardwood lumber processing' [HMMW03], as well as for the study of art, archeology and cultural heritage [BC06].

## 3.2 Visualization: Techniques and Libraries

Common to 3D imaging techniques is that they produce vast amounts of data to process and visualize, which demands appropriate methods. There are two types of visualization mappings for volume data: isosurfaces and direct volume rendering [Grö02]. An *isosurface* represents points of a constant value within a volume. A common algorithm for constructing the isosurface that creates triangle models of constant density surfaces is the *marching cubes* algorithm by Lorensen and Cline [LC87]. A different method for computing a 2D image from a 3D dataset is *volume ray casting* as described by Drebin

et al. in 1988 [DCH88]. Kaufman and Mueller [KM04] give an overview of volume rendering. The *maximum intensity projection* (MIP) is a parallel projection that shows the maximum intensity value along each ray through the image. It was invented for use in Nuclear Medicine by Jerold Wallis in 1988 [WMLK89].

Various libraries and toolkits exist that implement algorithms for visualization and registration, a lot of them are open source software. In particular the field of medical imaging provides a wide range of software to choose from. Software for medical imaging is valuable for materials science, as the image modality is similar. The most notable requirement is support for the rendering of volume data (voxels). The following list provides toolkits that support volume rendering:

**The Visualization ToolKit (VTK)**  VTK is an open source software for 3D computer graphics, image processing, and visualization. It is written in C++ and available for Windows, Unix-based platforms and Mac OS X. VTK provides visualization algorithms for scalar, vector, tensor, texture, and volumetric data and advanced modeling techniques like implicit modeling, polygon reduction, mesh smoothing, cutting, contouring, and Delaunay triangulation. It enables mixing 2D imaging / 3D graphics algorithms and data. It has an extensive set of 3D widgets including point, line, plane, implicit plane, box, sphere, scalar bar, image plane, and spline widgets and supports texture-based volume rendering, VolumePRO volume rendering hardware, mixing opaque surface geometry and volume rendering [VTK].

**Volume Graphics VGL**  *VGL* by *Volume Graphics* is a commercial cross-platform C++ class graphics library dedicated to volume rendering. It provides a real-time volume ray-tracer, isosurface rendering and MIP, as well as hardware accelerated volume renderers; ambient, diffuse and specular shading, material and light settings; 3D texture, 2D texture rendering with shading support); parallel, perspective, and stereo rendering support [Vol06].

**SIM Voleon (for Systems in Motion's Coin)**  *SIM Voleon* is a free cross-platform 3D graphics library with a C++ application programming interface. It is an add-on library to *Systems in Motion's* visualization platform technology, known as *Coin*. The support includes 2D texture mapping, hardware assisted 3D texture mapping, Maximum Intensity Projection and Sum Intensity Projection through hardware, and picking support [Sys08].

## 3.3 Image Registration

*Image registration* is 'the process of determining the spatial transform that maps points from one image to homologous points on an object in the second image' [ISNC05]. An extensive review of image registration methods can be found in [ZF03].

Volume registration by maximization of mutual information (MI) was introduced by Collignon et al. [CVSM95] and Viola and Wells [VWMW97]. While Viola and Wells propose Parzen-window probability density function (PDF) estimation, the Mattes et al. implementation [MHV$^+$01] is based on the joint histogram of the images using B-Spline kernels.

**Insight Segmentation and Registration Toolkit (ITK)**   The *Insight Segmentation and Registration Toolkit* (ITK) is a cross-platform (Unix, Windows and MacOS X) open source software system that was designed for performing registration and segmentation. The focus of the toolkit is mainly on medical applications, but not limited to these. It provides data representation and algorithms, but does not address visualization. It supports multi-threaded (shared memory) parallel processing. A pipeline is used to connect data objects with filters. ITK is written in C++ using generic programming and 'many advanced design patterns' [ITK].

**Medical Imaging Interaction Toolkit (MITK)**   The *Medical Imaging Interaction Toolkit (MITK)* combines VTK, ITK and the pic-based libraries of the *Division of Medical and Biological Informatics* of the *German Cancer Research Center (DKFZ)* [Ger] and adds features that are covered neither by VTK nor ITK like multiple, consistent views on the same data, undo/redo of interactions, organization of data objects in a tree at run-time, visualization and interaction with 3D+T data, structured combination of modules (so-called functionalities), e.g., for combining and switching between one functionality for segmentation and another one for registration [WVW$^+$05].

**Automated Image Registration (AIR)**   *Automated Image Registration* (AIR) is a program suite for volume-based image registration. Its source code written in C is available to the research community free of charge. *AIR* is especially designed for neuroimaging applications and has primarily been used in research-oriented functional neuroimaging with brain scans from positron emission tomography and magnetic resonance scanners.

The suite provides a number of programs for image registration with different transformation models, such as rigid-body, affine and nonlinear warping [Wik07].

**FLIRT** *FLIRT* is an automated linear (affine) registration tool based around a multi-start, multi-resolution global optimization method. It can be used for inter- and intra-modal registration with 2D or 3D images. In addition, it can be run with a number of different transformation models (degrees of freedom) and it implements a general cost function weighting scheme for all cost functions [JBBS02].

## 3.4 Voids in AlSiC

Research on changes of residual stresses and micro voids during thermal cycling of AlSiC MMCs is an ongoing project. The *Institute of Materials Science and Technology* of the *Vienna University of Technology* documented the current research in AlSiC in several publications ([SRD$^+$07b] [SRD06] [HDLS06]).

To characterize reasons for failure, non-destructive material investigations including *Acoustic Scanning Microscopy*, *X-Ray*, and *Neutron Diffraction* were done. Important for the understanding of the deformation behavior and thermal fatigue is the investigation of the residual stress and voids in the composite and their change under operating conditions respectively under dynamic thermal load [GLD03]. *Acoustic Scanning Microscopy* delivers information about lateral density differences respectively material inhomogeneities integrated over the sample thickness. Diffraction methods are used for in situ evaluation of macro and micro stresses, Synchrotron X-Ray tomography provides three-dimensional structural information as well as the residual stress state that helps to understand the internal behavior of MMCs under thermal load.

# 4 Hypothesis

## 4.1 Voids in MMCs and their Effects on Thermal Expansion
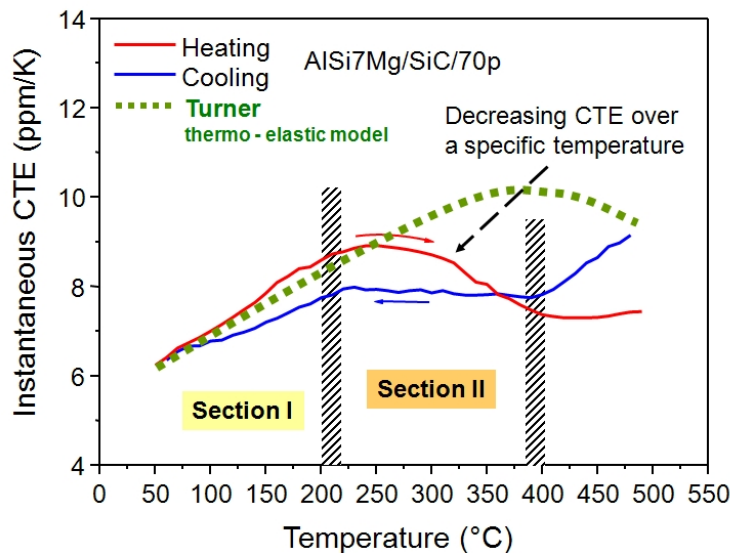


Figure 4.1: The *coefficient of thermal expansion*(CTE) can be separated into two sections where the CTE shows linear behavior in the first section. In the second section above a characteristic transition temperature the CTE decreases at heating and stays constant at cooling down the system [SRD06] (image from Huber [Hub03])

The following excerpt about the expansion behavior of AlSiC MMCs depicts the consistency of the sample, the procedure of the measurement, and presents the hypothesis explaining the anomaly in the CTE(T) compared to thermoelastic models (see Figure 4.1). The paragraph is an excerpt from a progress report on high temperature heat sinks titled 'Voids in MMCs and their effects on thermal expansion' by M. Schöbel, G. Requena and H.P. Degischer of the *Institute of Materials Science and Technology* [SRD06]:

19

'*AlSiC usually consists of an AlSi7Mg matrix which infiltrated SiC particulate preforms of about 70 vol.-% thus denominated AlSi7Mg/SiC/70p. This type of MMC provides good thermal conductivity and a low thermal expansion coefficient (CTE). The mismatch in thermal expansion coefficients between the MMC constituents amounts around 20 ppm/K. Therefore significant internal stresses are building up during changes in temperature. When the elastic stresses in the matrix relax by plastification, the thermo-elastic models are not valid any more for the prediction of the CTE(T) of the composite. When AlSiC is heated from room temperature to more than 200°C, the growth of CTE(T) slows down and it passes a maximum value of 9 ppm/K at 250°C from which it drops to 7 ppm/K at 400°C. The CTE(T) remains almost constant 7–8 ppm/K during cooling down to 200°C.*

*Such an anomaly in the temperature dependence of the CTE is explained by corresponding changes in the volume fraction of micro-voids within the MMC. These are generated not only by incomplete infiltration during processing, but as well during cooling after solidification due to the mismatch in thermal expansion between SiC and the Al-matrix. The volume fraction of voids is measured by synchrotron tomography while the AlSiC sample is heated to 400°C in average at 5K/min and cooled down to room temperature again at similar rate. The decreasing volume fraction of voids with increasing temperature and its increase during cooling are correlated to the CTE(T) curve. The relative reduction of the CTE(T) above 200°C is explained by filling of voids by the viscous flow of the matrix subjected to compressive stresses. During cooling compression is inverted into increasing tensile stresses in the matrix which open voids again resulting in an almost constant CTE(T).*

*The integrity of the MMC during multiple thermal cycles is essential for the conservation of high thermal conductivity and dimensional stability of heat sink materials. The AlSiC sample was subjected to 25 thermal cycles between −100°C and 400°C. Synchrotron tomography revealed the increase of porosity resulting from local debonding at the interfaces between matrix and bigger SiC-particles.*' [SRD06]

Figure 4.2 visualizes the change in void volume of a $420^3 \mu m$ cube during a thermal cycle. Our aim is to develop software that enhances the existing analysis capabilities regarding void kinetics. By aligning the existing datasets it will be possible to study the behavior of a specific void and gain more insight into the structural changes. We present a 3D registration approach based on mutual information that allows for sub-pixel accurate registration. A maximum intensity projection (MIP) may be used for a manual pre-alignment.
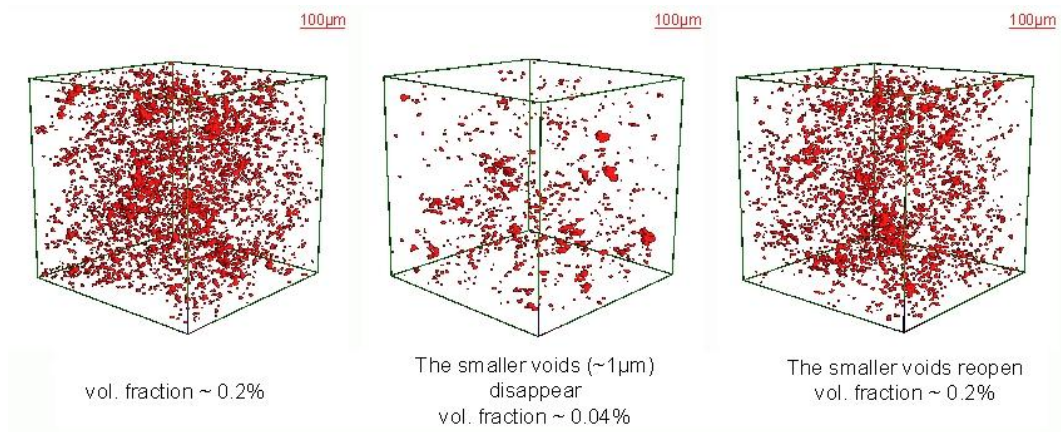
Figure 4.2: Opening and closing of voids during a thermal cycle: room temperature (RT) - 400°C - RT. [SRD+07b]

Further software tools for the analysis of void kinetics of AlSiC provide linked views for 2D slices, 3D isosurfaces, volume rendering, as well as overlaid contours (isolines) of multiple images on an arbitrary positionable cutting plane, and the calculation of a shape factor. After registering the volumetric datasets, we visualize the shrinkage and expansion of a specific void during a heating cycle between room temperature and 400°C and analyze the change of quantitative properties of specific voids–including volume, surface, shape factor–resulting from temperature changes.

# 5 Methodology

> Any sufficiently advanced technology
> is indistinguishable from magic.
>
> ――――――――――――――――――――――
> ARTHUR C. CLARKE

In this chapter we present the basic principles of the registration (Section 5.1) and give an overview of the visualization software developed to analyze the registered datasets (Sections 5.2 and 5.3). Details on the implementation can be found in Section 5.4.

## 5.1 3D Image Registration

The movement of the sample between imaging needs to be corrected in order to be able to observe the same void over multiple images. Ibanez et al. [ISNC05] define image registration as 'the process of determining the spatial transform that maps points from one image to homologous points on an object in the second image'.

We expect the sample to expand uniformly due to the isotropic material distribution. The transformations therefore needed are translation, rotation and global scaling, which are linear transformations, more precisely affine transformations. Registration where only an affine transformation is needed is called *rigid* registration—contrarily to an elastic *non-rigid* registration which can cope with local deformations.

The data are produced by the same imaging source, thus the registration can be categorized as a *single modal* registration. An image similarity based method—mutual information—measures the matching of the images. The following chapter presents details about the *rigid single-modal 3D registration*.

### 5.1.1 Overview of the Registration Process

For a successful registration the following steps are required (see Figure 5.1):

1. Converting the data into 8-bit gray levels and saving in the VTK file format (see Section 5.4.1).

2. Loading the data into the registration application and optionally removing the major components of misalignment as described in Section 5.4.2.

3. Adjusting the parameters of the registration, as explained in Section 5.4.2.

4. Registering the datasets.

5. By watching the metric value (see Section 5.4.2) and visually evaluating the checkerboard comparison (see Section 7.1.1) one can decide whether to accept the result or start a new registration run using optimized parameters.
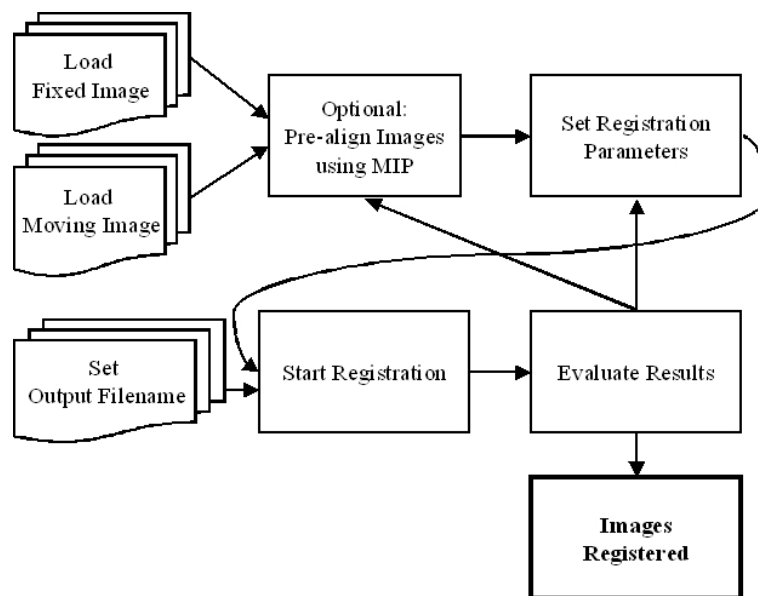


Figure 5.1: Registering two 3D datasets involves optionally correcting the initial displacement, setting the registration parameters, and evaluating the result. Using the knowledge from a first registration may lead to an improved parameter set for further registrations.

## 5.1.2 Registration Components Overview

Ibanez et al. [ISNC05] describe the basic components of the ITK registration framework (see Figure 5.2 as follows:

> 'The *transform* component [. . . ] represents the spatial mapping of points from the fixed image space to points in the moving image space. The *interpolator* is used to evaluate moving image intensities at non-grid positions.

Figure 5.2: The basic components of the ITK registration framework are two input images, a transform, a metric, an interpolator and an optimizer [Kit05].
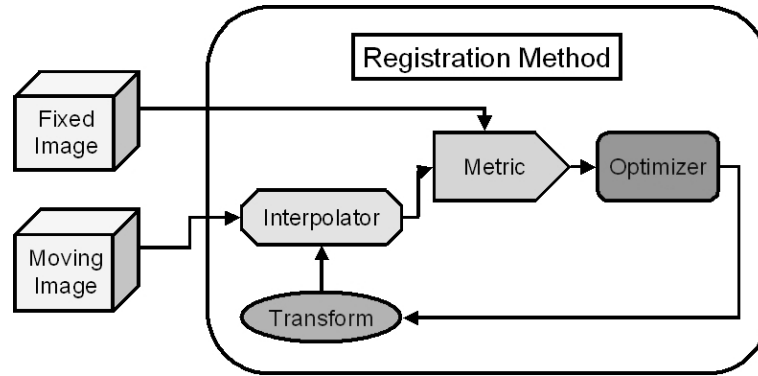
The *metric* component [...] provides a measure of how well the fixed image is matched by the transformed moving image. This measure forms the quantitative criterion to be optimized by the *optimizer* over the search space defined by the parameters of the *transform*.'

## 5.1.3 Transform and Initialization

The *affine transform* handles rotation, scaling, shearing and translation. Ibanez et al. [ISNC05] propose to remove the major components of misalignment before applying the affine transform to reduce the computational costs. In our data translation is the major form of misalignment. Rotation and scaling are minimal compared to the translation.

We provide the user with an adaptable overlaid *maximum intensity projection* (MIP) of highly absorbing iron precipitates of the two images to assist a manual initialization of the translation. This helps to reduce the initial misalignment improving the registration performance by a factor of two or more (see Section 5.4.2). The iron precipitates in Al-alloys originated in the production process.

## 5.1.4 Metric

An image similarity based measure is used to assess the progress of the registration. The notion of *mutual information* (MI) is based on probability theory and information theory. 'Mutual information (MI) measures how much information one random variable (image intensity in one image) tells about another random variable (image intensity in the other image)' [ISNC05].

Mutual information is defined in terms of entropy. Let

$$H(A) = - \int p_A(a) \log p_A(a) \, da \tag{5.1}$$

be the entropy of random variable $A$, $H(B)$ the entropy of random variable $B$ and

$$H(A, B) = \int p_{AB}(a, b) \log p_{AB}(a, b) \, da \, db \tag{5.2}$$

be the joint entropy of $A$ and $B$. If $A$ and $B$ are independent, then

$$p_{AB}(a, b) = p_A(a) p_B(b) \tag{5.3}$$

and

$$H(A, B) = H(A) + H(B). \tag{5.4}$$

where $p_{AB}(a, b)$ is the joint probability distribution function of $A$ and $B$ while $p_A(a)$ and $p_B(b)$ are the marginal probability density functions.

However, if there is any dependency, then

$$H(A, B) < H(A) + H(B). \tag{5.5}$$

The difference is called Mutual Information : $I(A, B)$

$$I(A, B) = H(A) + H(B) - H(A, B) \tag{5.6}$$

The joint probability function $p_{AB}(a, b)$ is evaluated at discrete positions or bins uniformly spread within the dynamic range of the images. Entropy values are then computed by summing over the bins [ISNC05]. The number of bins has a great influence on the progress of registration, as the optimizer may get trapped at local minima. Prenormalization of the intensity values of the image is not necessary. As the metric rescales internally when building up the discrete density functions it is contrast insensitive.

Registration was done using the ITK implementation of the *Mattes Mutual Information* algorithm [MHV+01], which addresses registration as an optimization problem. The algorithm tries to maximize the mutual information between the two images by traversing the parameter space of the spacial transformation [OIK+07]. The mutual information registration algorithm described by Mattes et al. [MHV+01] uses only one set

of intensity samples which 'results in a much smoother cost function and hence allows the use of more intelligent optimizers' [ISNC05] compared to the initial implementation proposed by Viola and Wells [VWMW97].

## 5.1.5 Optimizer

A *regular step gradient descent* optimizer is used. It advances the parameters of the transform in the direction of the gradient where a bipartition scheme is used to compute the step size $L$ [ISNC05]. The user sets the maximum step size $L_{max}$ and the minimum step size $L_{min}$. Iterations end when the computed step size is smaller than the minimum step size($l < L_{min}$).

## 5.1.6 Multi-Resolution Registration

Best results could be achieved using multi-resolution registration. In general multi-resolution improves registration speed, accuracy and robustness of the image registration by eliminating local optima at coarser scales [ISNC05]. Registration is first done using down-sampled images. These are created by applying a Gaussian smoothing followed by subsampling. Once the convergence criterion defined by the *minimum step length* is reached, registration continues on the next level. The *minimum step length* is reduced by a factor of ten every level. The resulting transformation is applied to the next finer image as the initial transformation. The number of iterations per level is limited by the *number of maximum iterations*. Multiple stages of down-sampling can be combined to form an *image-pyramid*. Using the multi-resolution approach allows for registering images with more displacement than it would be possible without. On the downside, registration takes longer due to the involved down-sampling and needs additional RAM for the down-sampled images.

## 5.2 Overlaid Void Contours

A tool for comparison of multiple void contours has been developed (see Figure 5.3). Isolines of a void are plotted along an arbitrary positioned cutting plane. These contours can be overlaid for multiple volumetric datasets and displayed in the same image.
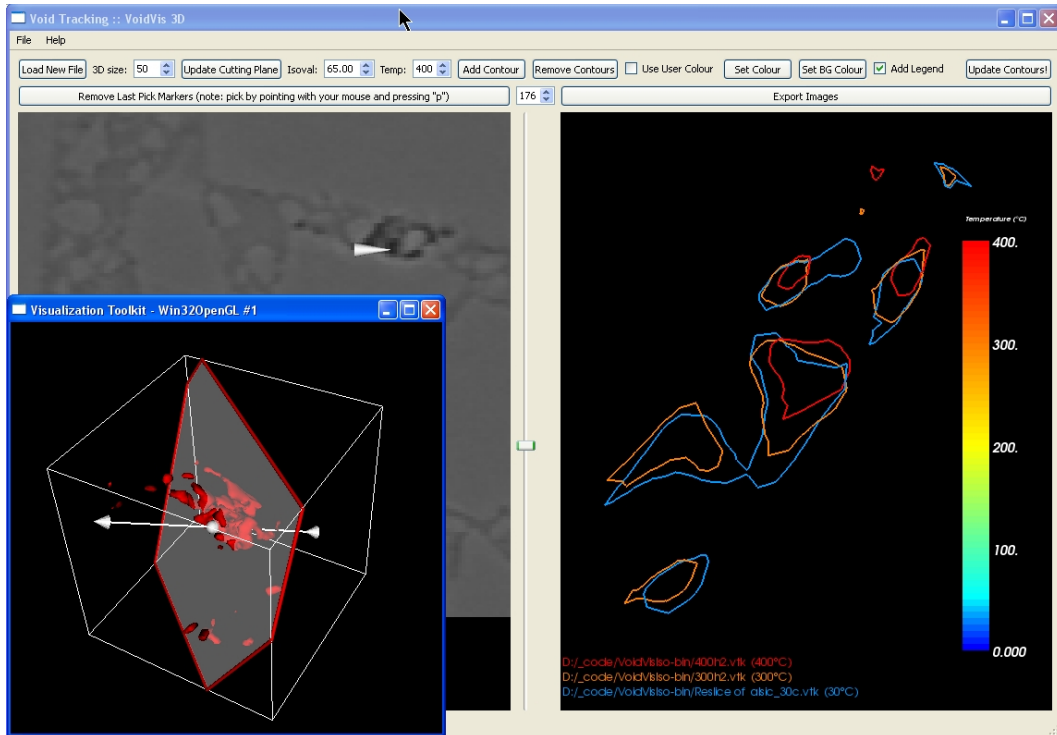


Figure 5.3: Overlaid void contours for images at different temperatures along an arbitrary cutting plane.



Figure 5.4: Comparing multiple void contours by selecting a void, positioning the cutting plane in space, and adding contours.

First a dataset is loaded and presented in a 2D slice view, that supports continuous infinite zooming and panning. After picking a point in or near a void, the three dimensional cutting plane through an isosurface rendering of that void can be adjusted.

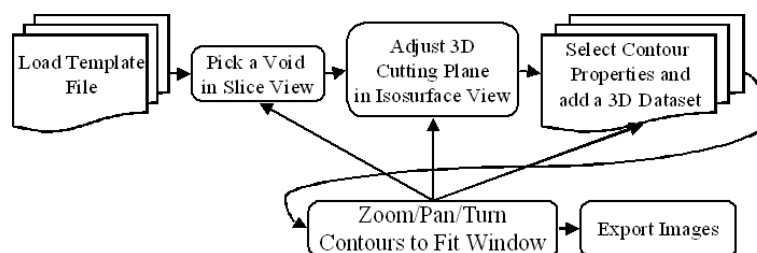Registered volumetric datasets can then be added, and for every volume an isoline of the voids on the cutting plane is rendered. Adding multiple datasets superimposes the void contours. When changing the orientation of the cutting plane the void contours update accordingly. Figure 5.4 shows the work flow.

The main window of the application consists of a 2D slice view to pick the void position, and the render window into which the contours are plotted. An additional window is used to position the 3D cutting plane. The translucent plane is rendered together with a marching cubes isosurface of the void. It can be adjusted arbitrarily by grabbing it with the mouse.

## 5.3  Linked 2D/3D Views for Void Comparison

After the volumetric images have been registered to each other, it is possible to compare a specific feature in images acquired at different temperatures. We developed an application for comparison of two 3D images (see Figure 5.5). Linked views for 2D and 3D comparison help to analyze the alterations occurring during a thermal cycle.
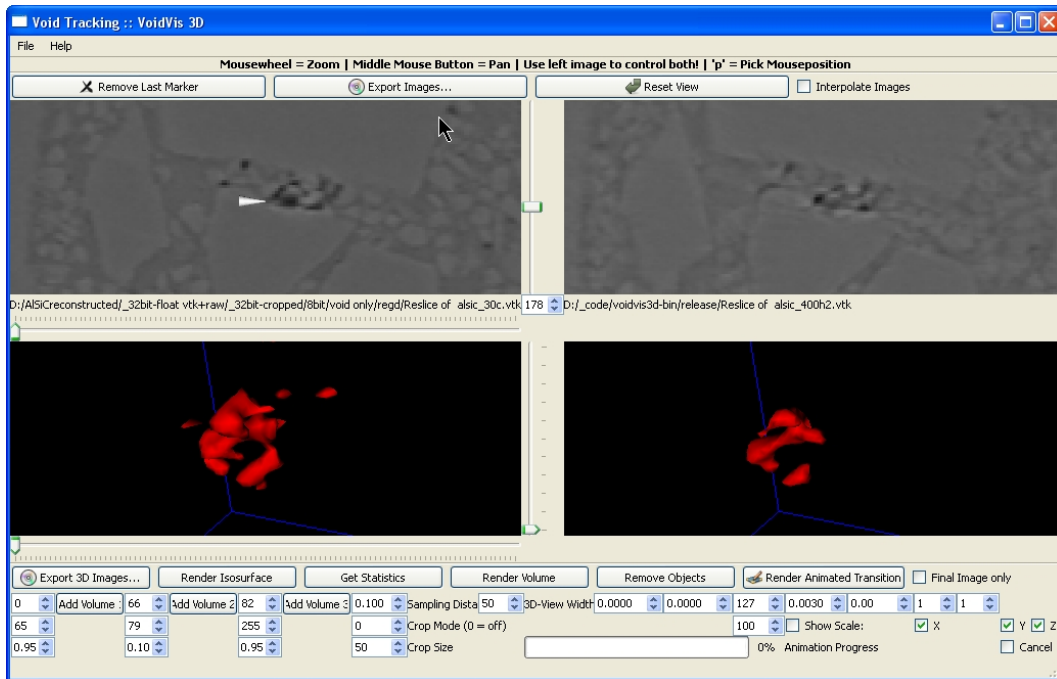


Figure 5.5: 3D void comparison using linked views (examples at RT and 400°C).

### Linked 2D Slice View

Linked views allow for a direct one-to-one comparison of two slices. Functionality for continuous infinite zooming and panning supports the analysis of alterations in 2D. A linear interpolation for zooming is optionally available. Linking the side-by-side slice views together ensures that the same features are always visible in both render windows. We implemented the linked views by using the same camera object on both datasets.

### Distance Measurement and Picking

When a point is picked in the slice view its 3D position and the Euclidean distance to the previously picked point is displayed. The picked point is used as the center point

for the 3D visualization. An arrow marks the picked positions. The Euclidean distance between the point $P = (p_x, p_y, p_z)$ and $Q = (q_x, q_y, q_z)$ is defined as:

$$\sqrt{(p_x - q_x)^2 + (p_y - q_y)^2 + (p_z - q_z)^2}$$

**Linked 3D Rendering**

The 3D render windows are also linked together, thus when moving one camera the other one follows. *Volume rendering* and *isosurface rendering* modes are available.

**Volume Rendering**   A volumetric raytracing render mode is implemented. A cube centered at the picked position with variable side length is volume rendered. The transfer function is divided into three linear sections: one for the voids, one for the silicon-carbide and one for the aluminum. The sampling distance may be adjusted to trade-off speed for accuracy. A good rendering quality was achieved using a sampling distance of 0.1 voxel. Values greater than 1 result in far faster rendering. Various crop modes are supported. By cropping it is also possible to cut out parts of the cube. It is possible to select subvolume, fence, inverted fence, cross, and inverted cross crop modes.

**Isosurface Rendering**   The isosurface rendering provides a more responsive interaction than the volume rendering, as the rendering is faster once the surface is extracted. The *marching cubes* algorithm is used to generate the isosurface for a cube centered at the picked position. The global isovalue can be set by the user (see Section 7.2 for considerations on the selection of the isovalue).

## 5.4 Implementation Details

The following chapter provides details on the implementation such as the parameters used to initialize the registration.

### 5.4.1 Putting the data into the VTK file format

The applications developed for this thesis can read input files in the VTK file format. The VTK file format stores 3D data together with a header containing information about the data type and dimension. We used the open source image processing application *ImageJ* [Ras07] for converting RAW files to the VTK format. A plug-in [Ima] enables *ImageJ* to read and write VTK files. The image type was set to 8-bit greyscale. A multitude of plug-ins is available for *ImageJ* to read various file formats.

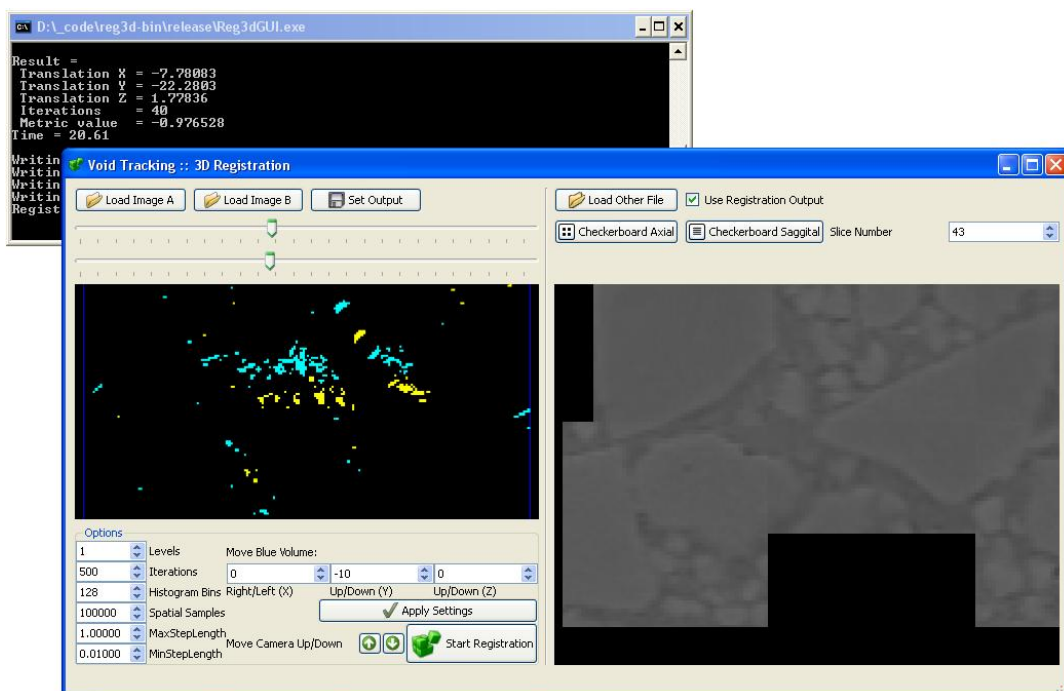### 5.4.2 Registration Parameters



Figure 5.6: 3D registration application for manual pre-alignment using MIP and adjusting the registration parameters. Checkerboard comparison helps to visually evaluate the results while the console provides detailed information about the process.

An application for registering the tomograms using mutual information was developed (Figure 5.6), incorporating specific features like a manual maximum intensity projection (MIP) based linear pre-alignment. Several parameters regarding the registration can be adjusted:

- Initial translation

- Number of levels (multi-resolution image pyramid)

- Number of maximum iterations (per level)

- Number of histogram bins

- Number of spatial samples

- Maximum step length $L_{max}$

- Minimum step length $L_{min}$

These are set to initial values which are likely to give a satisfactory registration result. Once it is known that the datasets register properly, it is possible to speed up the registration process by tuning the parameters, e.g., reducing the number of levels, the maximum step length, etc. Figure 5.7 shows the registration pipeline with its components.

**Initial translation**

By using MIP to visualize only highly absorbing iron precipitates in the fixed and moving image it is possible to easily reduce the initial offset by aligning clearly distinguishable patterns. In the initial view it is possible to align on the x-y axis, by tilting the camera 90° upward this can be done for the x-z axis. The resulting 3D translation is fed as the initial translation into the registration algorithm, reducing the number of required iterations. It is possible to avoid the use of multi-resolution image-pyramids by pre-aligning the datasets to an accuracy within the capture radius of the registration algorithm, greatly reducing the overall time, as this avoids the costly rescaling. The capture radius is the maximum displacement that the algorithm can correct. For a single layer registration this is about 15 to 25 voxels in our case.

**Number of levels (multi-resolution image pyramid)**

For the multi-resolution registration the number of levels can be set. Registration then starts on a coarser, down-sampled image and continues on a higher resolution image using the previous results. Each additional level reduces the image by a factor of two.
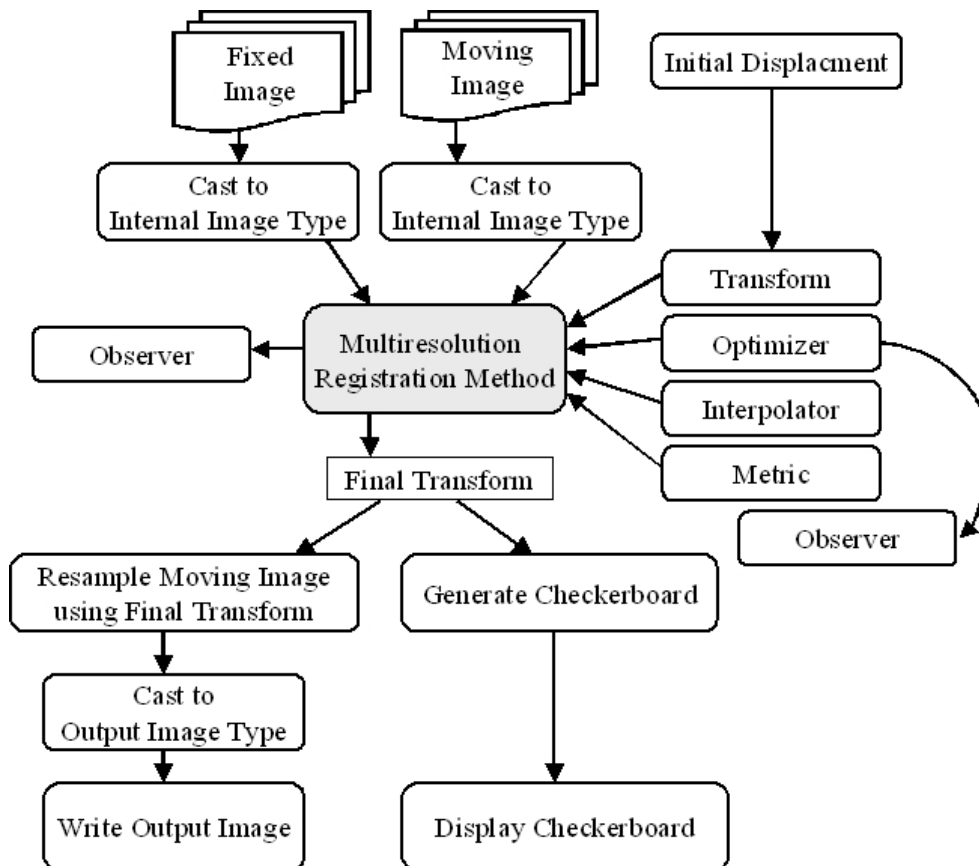
Figure 5.7: Overview of the registration pipeline.

Good results were achieved without multi-resolution registration ($levels = 1$) when the initial misalignment was in the range of 15 to 25 voxels for a cube of $200^3$ voxels. Multi-resolution registration allows for registration of volumes with a greater initial misalignment without the need for a manual pre-alignment. Note that the down-sampled images might get too small to be useful when the number of levels is too high (e.g., $levels = 4$ for a cube of $200^3$ voxels).

**Number of maximum iterations**

The number of maximum iterations specifies when the algorithm should stop at the latest. It might stop earlier when no further optimization is possible using the defined minimum step length. As a rule of thumb, we start using a low number (e.g., 50 iterations), we watch the output of the registration for the metric value as it should gradually get lower as the optimizer treats the mutual information as a cost function and tries to minimize it. When the metric value decreases consistently a good set of parameters

is found. We then increase the number of maximum iterations in order to allow the optimizer reaching the desired precision tolerance. 500 – 3000 iterations proved to be a sane upper limit for a registration with a repeat accuracy in the sup-pixel range. If the desired precision can not be reached (e.g., due to the optimizer being trapped in a local minimum) the algorithm will stop after the number of maximum iterations.

### Number of histogram bins

Using less bins for the histogram results in faster calculation but the quantizing reduces the available information. As the input data is expected to be 8-bit per pixel, a value of 256 $(= 2^8)$ bins gives the best results, while smaller values (128 bins, 64 bins, ...) result in faster registration.

### Number of spatial samples

The Mattes et al. [MHV$^+$01] mutual information algorithm uses an arbitrary number of samples for the calculation of mutual information. While a larger number improves the smoothness of the metric from one iteration to the next this results in a longer calculation time per evaluation of the metric. According to Ibanez et al. [ISNC05] 'the number of spatial samples can usually be as low as 1% of the total number of pixels in the fixed image'. We were able to successfully register 3D images using 1.00% of the pixels as spatial samples.

### Maximum step length

The maximum step length sets the first step size. Setting this too small results in a larger number of iterations, increasing computation time. Setting it to 1 voxel is a good starting point, but it can be reduced (e.g., to 0.1 voxel) once the rough initial misalignment is known and considered in the initial translation.

### Minimum step length

The minimum step length represents the convergence criterion. 'This is equivalent to defining the precision with which the final transform should be known' [ISNC05]. We suggest to use a value of 0.1 if only a rough alignment is required, and 0.01–0.0001 for a precise alignment with no visible offset in the resulting checkerboard image. As long as

| Number of voxels | 400*400*400 | 400*400*400 | 400*400*400 | 400*400*400 |
|---|---|---|---|---|
| Initial translation | (0, 10, 0) | (0, 10, 0) | (10, 23, 4) | (12, 25, 5) |
| Levels | 1 | 1 | 1 | 1 |
| Max. iterations | 500 | 500 | 500 | 5000 |
| Histogram bins | 256 | 128 | 256 | 256 |
| Spatial samples | 1.10% | 1.10% | 1.10% | 1.10% |
| Max. step length | 1 | 1 | 0.1 | 0.1 |
| Min. step length | 0.1 | 0.1 | 0.001 | 0.000001 |
| Translation | (10.9, 23.2, 4.4) | (11.0, 23.3, 4.5) | (12.40, 24.81, 5.25) | (12.47, 24.83, 5.31) |
| Iterations | 23 | 25 | 185 | 540 |
| Final metric value | -0.76 | -0.72 | -1.07 | -1.07 |
| Time | 87 sec. | 90 sec. | 365 sec. | 932 sec. |

Table 5.1: Registration results using manual pre-alignment. A lower negative metric value denotes a better correlation.

the metric value during registration is gradually getting lower, using a smaller minimum step length will result in a better alignment.

**Optimizer parameter scaling**

The scale of units in translation is different from rotation: the rotation is measured in radians, the translation is millimeters. The units need to be rescaled for the optimizer to work properly. The scales for translation is set to 0.000001, the other matrix coefficients are set to 1. These values are specific to our datasets and account for the fact that the major form of misalignment is a linear displacement with only a minimal rotation.

**Example Parameter Set**

The following set of parameters is from a successful registration. It shows how roughly aligned datasets quickly converge. The manual alignment was performed as described in Section 5.4.2 by aligning the projected iron precipitates. It took only 23 iterations to reach an alignment where no major dislocation is easily visible. Registration applying a stricter convergence (e.g., setting the minimum step length to 0.001) criterion takes longer, but produces a visually "perfect" result (no visible dislocation at the borders of the checkerboard comparison).

For the same two datasets registration without manual pre-alignment by using multi-resolution image-pyramids results in the following:

| | |
|---|---:|
| Number of voxels | 400*400*400 |
| Initial translation | (none) |
| Levels | 3 |
| Maximum iterations | 500 |
| Histogram bins | 128 |
| Spatial samples | 1.10% (700,000) |
| Maximum step length | 1 |
| Minimum step length | 0.1 |
| Final translation | (12.40, 24.79, 5.28) |
| Actual iterations | 334 (in the last level) |
| Final metric value | -0.98 |
| Time | 1416 sec. |

Table 5.2: Results for registration without manual pre-alignment.

**Reproducibility**

The spatial samples are selected randomly by the Mattes mutual information metric. The pseudo-random generator used for this is always initialized using the same constant seed, ensuring that the same set of input images and parameters always yields the same result.

**Interpreting the Mutual Information Metric Value**

Calculating the mutual information metric value for the fixed image with itself gives $-2.35$, the value before registration of two distinct images is $> -0.001$. These values differ only slightly between the various AlSiC images.

The metric value is negative, as the optimizer treats it as cost function and tries to minimize it. Small metric values denote a better fit. When a registration fails due to the optimizer getting trapped in a local minimum (because of a large misalignment) typical metric values are larger than $-0.01$ ($-0.01 < metricvalue < 0$).

A registration that was able to align the images results in metric values considerably smaller than $-0.2$ to $-0.4$.

**Memory issues**

3D registration is a memory intensive task. Registering two cubes of $400^3 = 64,000,000$ voxels (8-bit data) uses $> 900MB$ of RAM. One reason is that registration internally uses data types of higher precision, typically floating point variables, another one that

| | | |
|---|---|---|
| Filesize | 15.2 MB | 61 MB |
| Resolution (voxels) | 200*200*200 | 400*400*400 |
| Memory requirement | 250 MB | 950 MB |

Table 5.3: Memory requirement for registration.

the image is held in memory multiple times by the processing pipeline. We use short integers as the internal data type for registration, which greatly reduces the memory footprint, and enables registering volumes of $400^3$ voxels on a computer running 32-bit Windows XP.

**32-bit Restrictions**    Currently the maximum size of datasets is limited due to memory restrictions imposed by the 32-bit operating system. For 32-bit systems the maximum addressable memory is 4 GB. Windows XP itself reserves 2 GB for its own use, leaving 2 GB of (virtual) memory for other applications [Micc]. The */3GB switch* in the file *boot.ini* [Micb] allows applications to address 1 GB of additional virtual address space above 2 GB, if they define IMAGE_FILE_LARGE_ADDRESS_AWARE in the process header. Our applications do make use of this in the header.

The limiting factor indeed seems to be fragmentation of virtual address space—not being able to allocate enough contiguous memory blocks although using well below 2 GB / 3 GB of virtual memory. We did not try to solve the issue of memory fragmentation, but further explanations of the Win32 memory handling and possible solutions can be found on the Microsoft website [Mica].

Porting the applications to a 64-bit OS would void the current memory limits and allow for registration and visualization of bigger 3D images, thus reducing the need for manual cropping of the datasets and enabling processing of the whole sample dataset at once.

**Large Data Sets in ITK / VTK**    Hints on how to handle large data sets in VTK can be found on the VTK website [Wik]. In memory-critical parts of the applications, we use the proposed *ReleaseDataFlag* flag to avoid that intermediate results are kept between the filters in the pipeline. This forces filters to release the memory after data objects have been processed, and reduces the memory footprint of the application. The downside is, that the filters have to re-execute if a parameter has changed, taking additional CPU cycles.

### 5.4.3 Calculation of Surface, Volume, and Shape Factor

The VTK class *vtkMassProperties* provides the means for calculation of surface, volume, and shape factor. As the class processes only triangles, we use the *vtkTriangleFilter* to convert the isosurface to triangle polygons.

**Surface and Volume**   The surface area is measured by using the *maximum unit normal component* (MUNC) method. The volume is measured using the *divergence theorem algorithm* (DTA). A description as well as an evaluation of accuracy and precision of these methods can be found in Alyassin et al. [ALIF94].

**Shape Factor**   The shape factor characterizes the deviation of the shape of an object from a sphere. The shape factor of a sphere is one. For every other shape it is greater than one. The shape factor $SF$ is defined as

$$SF = \frac{\sqrt{A}}{\sqrt[3]{V}} \Big/ N$$

with $A$ being the area of the void, and $V$ being the volume. The normalization factor $N$ is defined as the square root of the area of a sphere divided by the cube root of the volume of a sphere:

$$N = \frac{\sqrt{A_{sphere}}}{\sqrt[3]{V_{sphere}}} = \frac{\sqrt{4\pi r^2}}{\sqrt[3]{\frac{4}{3}\pi r^3}} = 2.199085233$$

### 5.4.4 Segmentation

To segment the voids for calculation of the surface, volume, and shape factor, we first generate an isosurface with the marching cubes algorithm using a global isovalue. The resulting contour is then processed as described in Section 5.4.3, i.e., converted to triangle polygons and the area calculated using the *maximum unit normal component* (MUNC) method. The volume is measured using the *divergence theorem algorithm* DTA) [ALIF94].

### 5.4.5 The Coding Environment

The target audience is currently using Microsoft Windows XP. Therefore all programs were written for MS Windows in C++ using Microsoft Visual Studio 2005 (a.k.a Visual Studio 8.0). The operating system is Microsoft Windows XP SP2 running on an AMD Athlon 64 XP Dual (2.41 GHz) equipped with 2.00 GB of RAM. Some libraries were statically linked and merged into the executable. While this results in bigger executables (being 5-7 MB) it was favored due to the ease of the installation and distribution. C++ was chosen because of the vast amount of freely available software libraries, its speed and efficiency.

### 5.4.6 External Libraries

Intensive use was made of external libraries for visualization, segmentation, registration, and for the graphical user interface. All libraries are open source software and available for multiple operating systems. Although portability was no major design decision it may therefore be possible to port the applications to other platforms without major redesign. The main components on which the software is based are Qt for the GUI, ITK (see Section 3.3) for the registration components, and VTK (see Section 3.2) for the visualization.

#### Trolltech Qt

*Qt* [Tro] is a cross-platform (Windows, X11 (Linux, Unix), Mac OS X) application development framework developed by Trolltech. We use it for the graphical user interface, as it integrates seamlessly with *The Visualization ToolKit* (VTK) and supports C++. Qt is available in two licenses, commercial and open source. The open source edition will not compile with MS Visual Studio due to restrictions imposed by Trolltech. A patched Qt 4.2.3 GPL source is however available at the paraview-website [QT a] and a patch including instructions for Qt 4.3 can be found at qtnode.net [QT b].

#### Boost

*Boost* [Boo] is a large collection of portable C++ libraries, parts of which have been accepted for incorporation into the Technical Report 1 of C++0x [ISO03]. The boost::lexical_cast function template offers a convenient way for conversion of numbers to strings.

## 5.4.7 Design Patterns

Design patterns—development paradigms that offer a general solution for re-occurring design issues—are heavily used by ITK and VTK. This includes the use of pluggable factories, command/observer design pattern, smart pointers, and generic programming.

**Smart Pointers**

Traditional ways of memory allocation using pointers in C++ easily lead to memory leaks when the programmer mistakenly forgets to free the referenced memory. The use of reference-counting smart pointers avoids these pitfalls by automatically counting the number of references and deallocating the memory when the last pointer to the object is destroyed. Therefore it acts as a *garbage collection.*

Both VTK and ITK offer build-in templated smart pointers for transparent reference-counting. The switch from traditional pointers to these smart pointers helped to avoid memory leaks and greatly reduced debugging time.

# 6 Results

> Computers are useless. They can
> only give you answers.
>
> ———————————————
>
> PABLO PICASSO

## 6.1 Registration

Figure 6.1 shows a 2D checkerboard comparison of two image details acquired at different temperatures before and after registration. One part of each image shows the sample at 30°C while the other part is the same sample at 200°C. The black lines have been inserted to emphasize the border. The misalignment before the images were registered is visible on the left side, especially when looking at the border of the brighter SiC-particles. The result of the registration is depicted in the right checkerboard comparison. The images are well aligned and no visible offset is left.



(a)           (b)
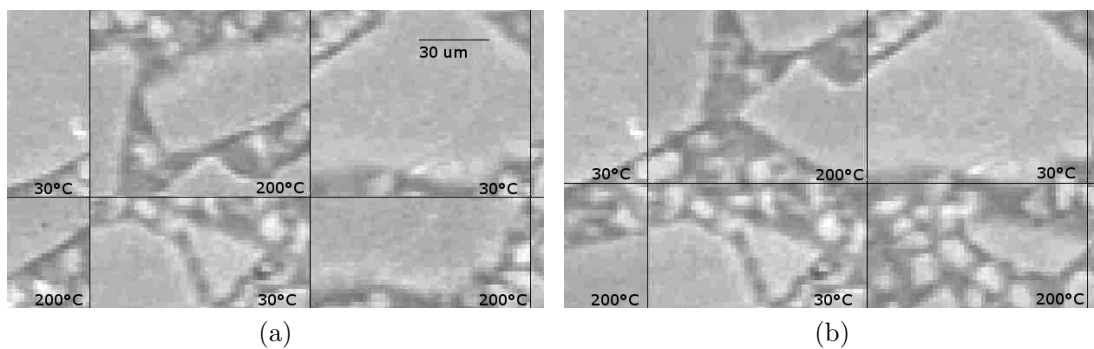
Figure 6.1: Checkerboard image of a slice at 30°C and 200°C (a) before and (b) after registration (2D detail). The displacement before registration can be seen at the borderlines. Image (b) is well-aligned.

## 6.2 Visualization of a Single Void during a Thermal Cycle

By first aligning the datasets it was possible to track a specific void over a thermal cycle. The thermal cycle is depicted in Figure 6.2. Figure 6.3 shows the position of a void between two SiC particles, restricting any expansion of the void to the space between the particles. Figure 6.4 visualizes the same void located between two SiC-particles (not pictured) during this thermal cycle:

$$30°C \rightarrow 200°C \rightarrow 300°C \rightarrow 400°C \rightarrow 300°C \rightarrow 190°C \rightarrow 50°C$$

The void shrinks and splits due to the thermal expansion of the surrounding aluminum matrix. When the sample is cooled, the void re-opens and recaptures its original shape with some minor spheroidization. Most changes take place between 300°C and 400°C.
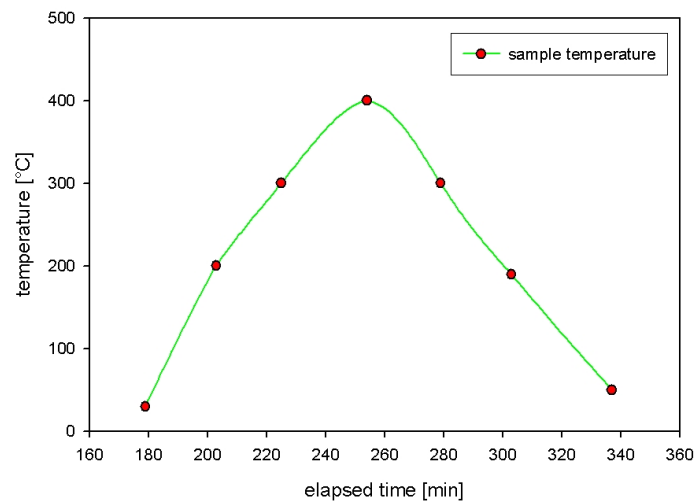


Figure 6.2: The sample was subjected to a heating cycle from room temperature (RT) to 400°C and back. Tomograms were acquired at 30°C, 200°C, 300°C, 400°C, 300°C, 190°C, and 50°C.
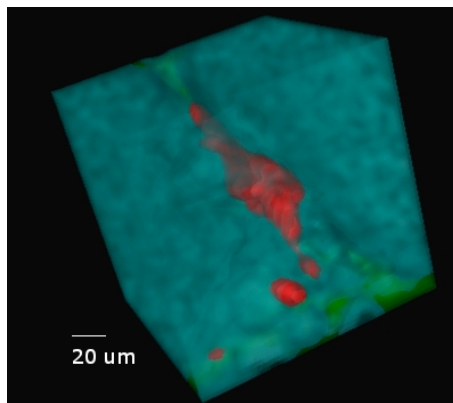
Figure 6.3: 3D volume rendering of a specific void embedded between SiC-particles. The shrinkage and expansion of this void during the heating cycle is depicted in Figure 6.4
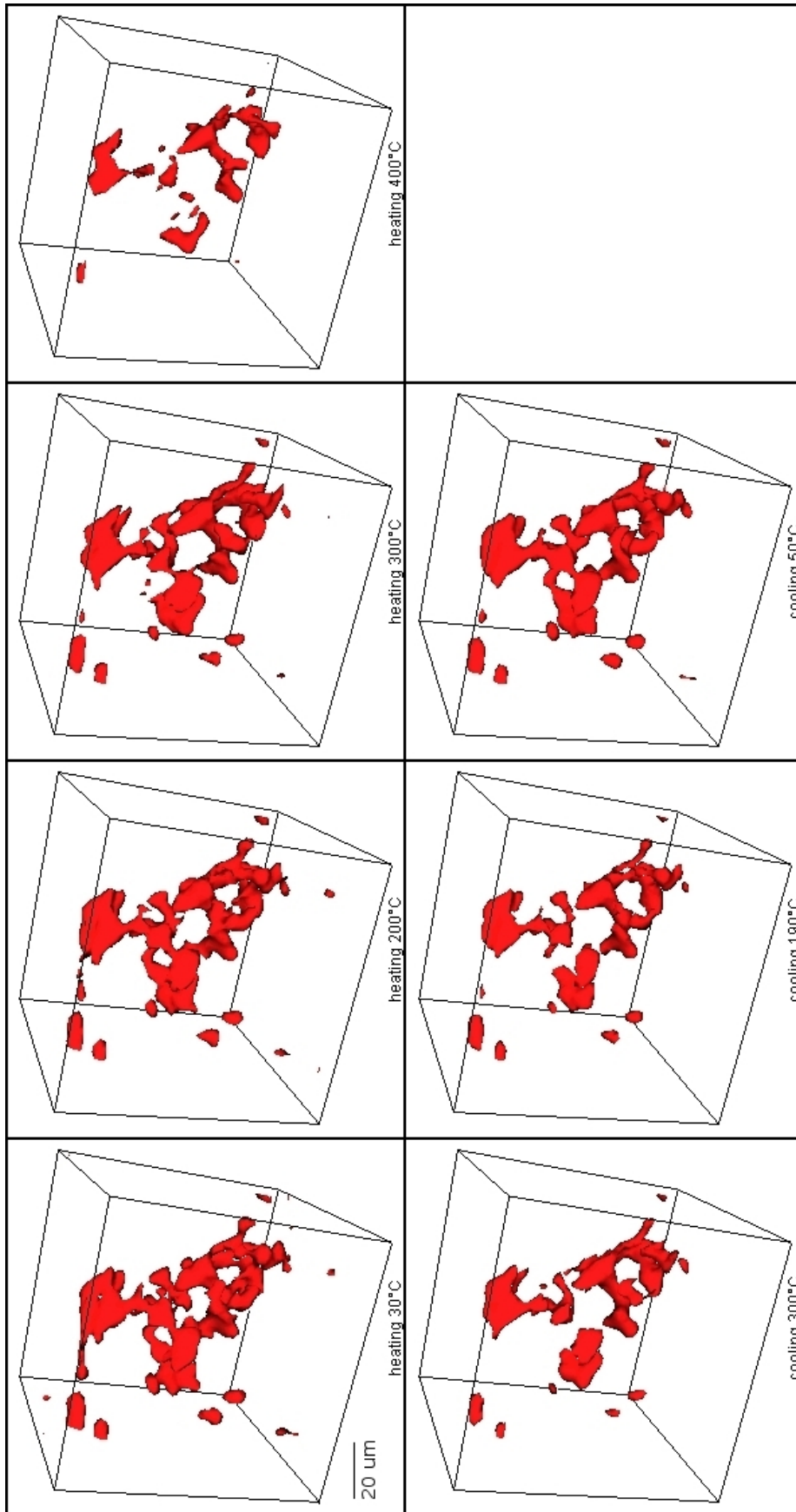
Figure 6.4: The change of a specific void during the thermal cycle $30°C \rightarrow 200°C \rightarrow 300°C \rightarrow 400°C \rightarrow 300°C \rightarrow 190°C \rightarrow 50°C$ (image order: left to right, top to bottom).

## 6.3 Quantitative Results: Microscopic vs. Macroscopic Thermal Behavior

It was shown by Schöbel et al. [SRD$^+$07b] how the void volume fraction changes under thermal load and this is assumed to cause the anomaly in the *coefficient of thermal expansion*. Figure 6.5 plots the volume fraction of a $(480\mu m)^3$ cube over time during a heating cycle. The datasets had not registered for this measurement, thus a representative volume had to be used.
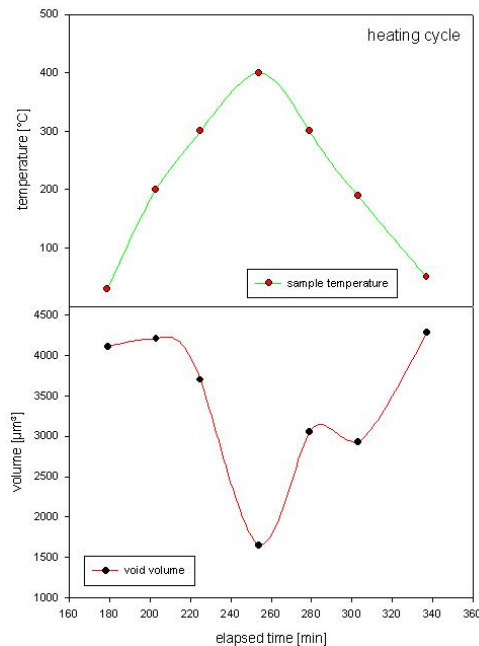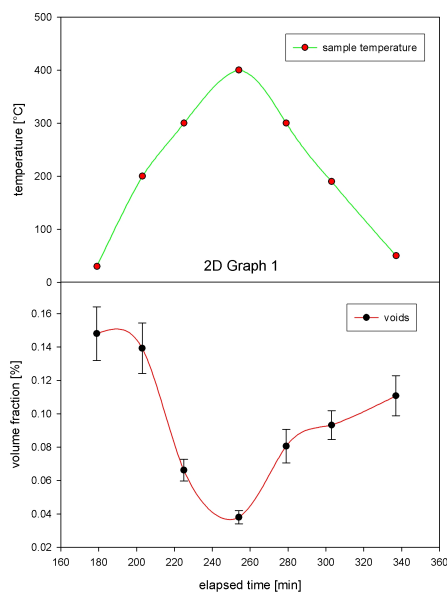


Figure 6.5: Change of void volume fraction of cube (side=480 $\mu m$).

Figure 6.6: Change of volume of a single void.

By aligning the datasets using the 3D image registration approach presented in this thesis, it is for the first time possible to compare the volume of a single void at different temperatures and visually observe the influence of the expanding matrix on the void shape. The plot of the volume of an individual void over time during a heating cycle can be found in Figure 6.6 and Figure 6.7. The behavior of this randomly selected individual void resembles the behavior of the whole cube. The biggest transformation is apparent above 200℃ during heating.

It is now possible for us to analyze the change in surface area and calculate a shape factor for a specific void, as depicted in Figure 6.7. A shape factor of one denotes
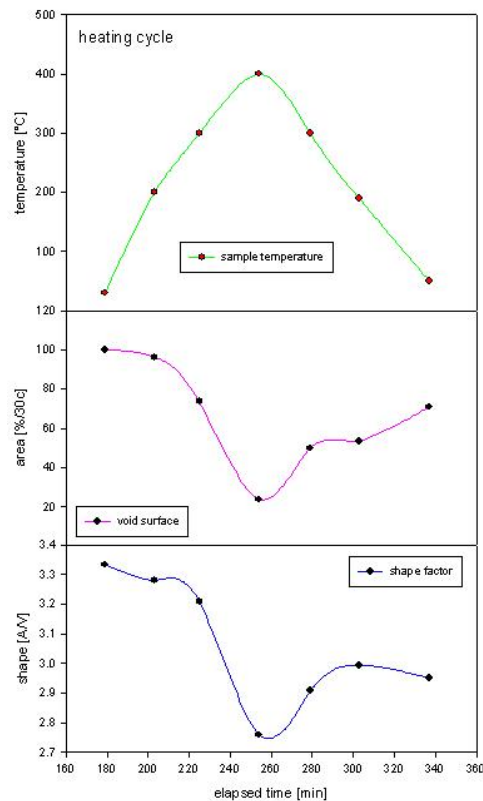
Figure 6.7: Change of area and shape of a single void.

a perfect sphere, whereas higher numbers indicate the deviation from a sphere (see Section 5.4.3 for the calculation of the shape-factor). That means that during heating some spheroidization takes place which is only partly reversed during cooling.

## 6.4 Overlaid Contours

The shape change during a heating cycle was also visualized by overlaying a cross section of the void at different temperatures. The position of the slice plane can be freely adjusted, and the resultant contours are color-coded according to their temperature.

Figure 6.8 shows that the void closes during the heating process. The contours of a void at 30℃, 300℃, and 400℃ are depicted. The shape change between 300℃ and 400℃ is notably bigger than between 30℃ and 300℃, which supports the theory that 'at higher temperatures (section II) inversion of internal stresses takes place submitting the matrix to compressive stresses. Thermally activated diffusion closes the voids successively producing a relative reduction in CTE(T)' [SRD06].

The statement of the 'reversible closing and opening of voids by plastic flow'[SRD06] is backed by Figure 6.9. The shape of the void prior to the heating cycle to 400°C matches the shape afterwards. The void almost returns to its prior shape at room-temperature after a substantial size reduction at 400°C. The contours depicted in Figure 6.9 show the void at 30°C before the heating and at 50°C afterwards. Little rounding of edges, i.e., some spheroidization, is observed.
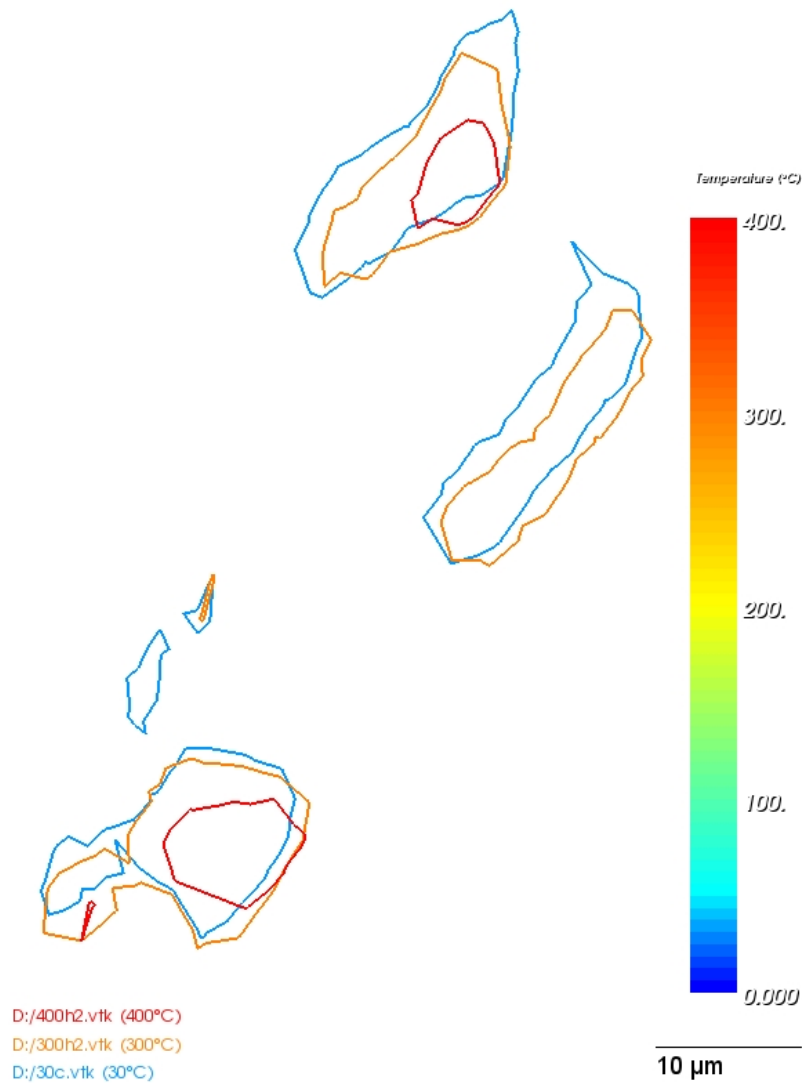


Figure 6.8: Void contours at 30°C, 300°C, and 400°C. Void 1 shrinks. Voids 2–4 disappear below resolution above 300°C. Void 5 shrinks and splits into two small sections close to 400°C.
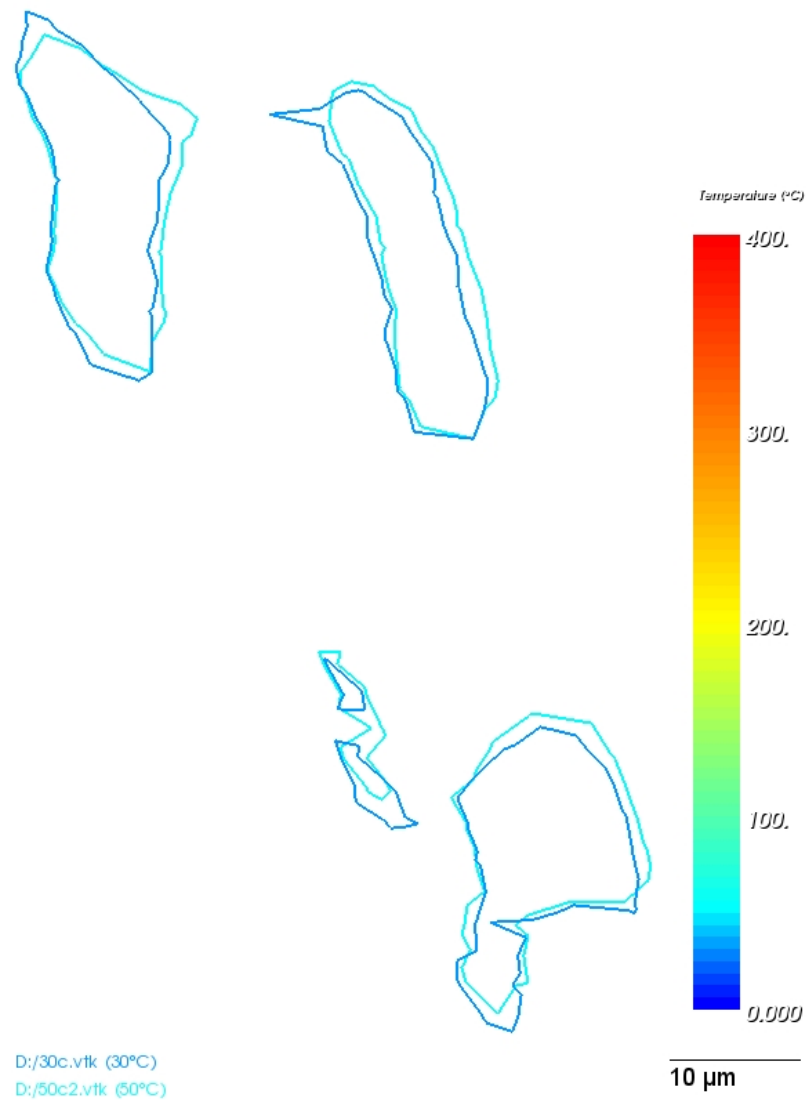
Figure 6.9: Void contours before (30°C) and after (50°C) thermal cycling. The void reopens to its initial size with little spheroidization.

# 7 Discussion

> When shall we three meet again In thunder, lightning, or in rain?
> When the hurlyburly 's done, when the battle 's lost and won.
>
> *MacBeth*
> WILLIAM SHAKESPEARE

## 7.1 Registration Accuracy

### 7.1.1 Checkerboard-based comparison

To evaluate the result of the image registration a checkerboard comparison tool is part of the registration application. A checkerboard comparison displays one image as the black squares and the other image on the positions of the white squares, thus providing borders at which the interleaved pictures are adjacent and their transitions can be compared. Visual assessment of the registration outcome is feasible, as misalignments of the SiC-particles can be spotted at the borders and corners of the resulting checkerboard. Panning, zooming, and changing slices facilitates the inspection. Figure 7.1 (right) is an example of a checkerboard comparison of a successful registration, while Figure 7.1 (left) shows the two interleaved datasets before registration.

### 7.1.2 Difference Images

Difference images show the deviation of two registered images by calculating the absolute difference between the values of pixel pairs. As our datasets are noisy and only a small range of intensity values is dominant it is difficult to show meaningful difference images. The source images in Figure 7.2 have been contrast enhanced using ImageJ [Ras07] before calculating the difference image. We also tried to reduce the noise using Gaussian smoothing and median filtering, but this did not give noteworthy better results.
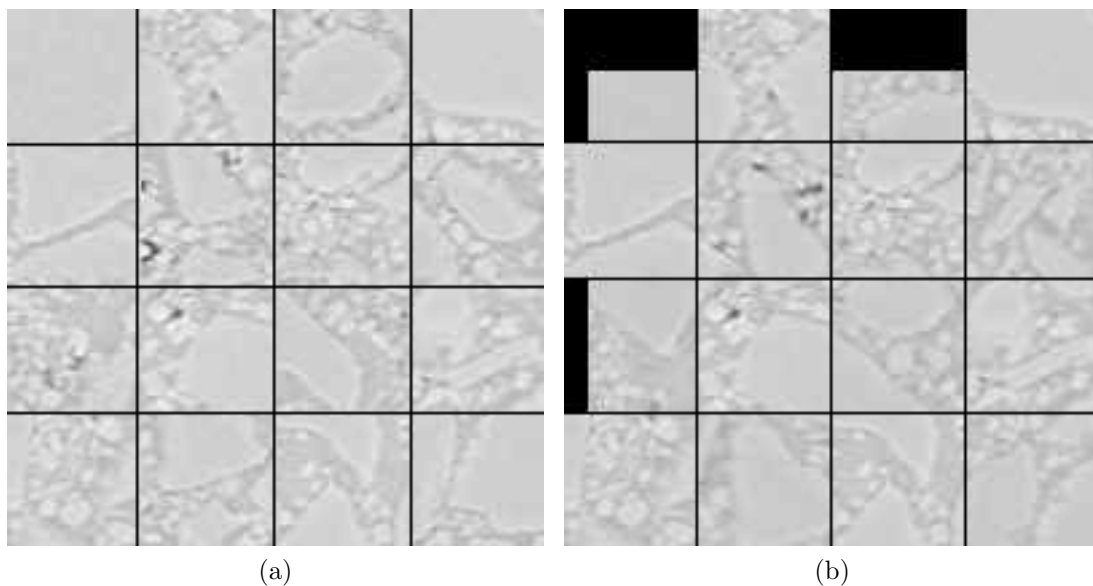
(a)       (b)

Figure 7.1: Checkerboard comparison: 30°C and 200°C before (a) and after (b) registration.

### 7.1.3 Joint Entropy Histogram

Figure 7.3 displays the *joint entropy histogram* of two images and shows that the mutual information increased by registering the images. The joint entropy histogram is given by

$$f_{ij} = -p_{ij} \log_2(p_{ij}) \tag{7.1}$$

where the indices $i$ and $j$ identify the location of a bin in the joint entropy histogram of the two images and are in the ranges $i \in [0 : N - 1]$ and $j \in [0 : M - 1]$. The image $f$ representing the joint entropy histogram has $NxM$ pixels because the intensities of image $A$ are quantized into $N$ histogram bins and the intensities of image $B$ are quantized into $M$ histogram bins. The probability value $p_{ij}$ is computed from the frequency count of the histogram bins.

$$p_{ij} = \frac{q_{ij}}{\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} q_{ij}} \tag{7.2}$$

The value $q_{ij}$ is the frequency of a bin in the histogram and it is computed as the number of pixels where image $A$ has intensities in the range of bin $i$ and image $B$ has intensities in the range of bin $j$. The value $p_{ij}$ is therefore the probability of the occurrence of the measurement vector centered in the bin $ij$ [ISNC05].

Figure 7.3 shows the joint entropy histogram before and after registration. 256 bins
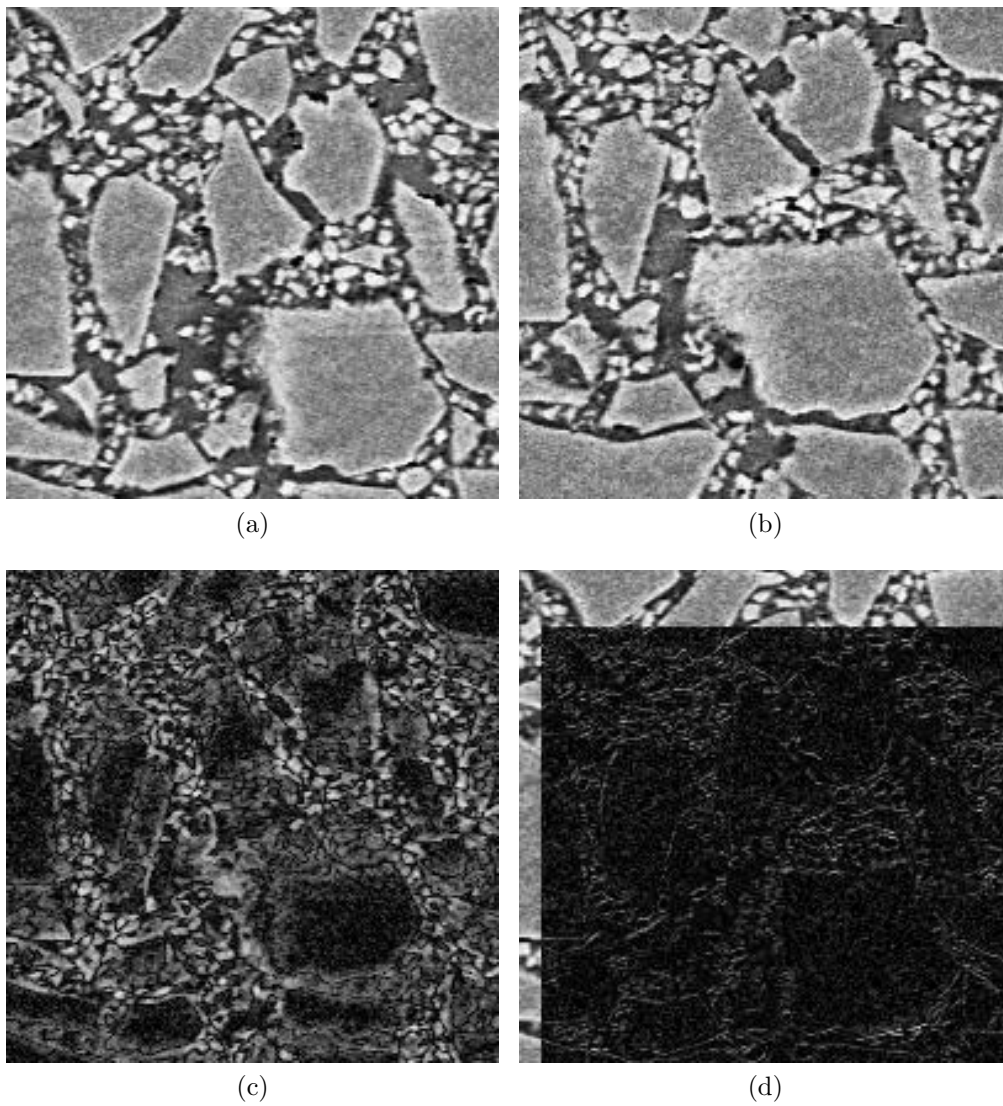
Figure 7.2: Registration results: (a) displays a slice from AlSiC at 30°C, (b) displays a slice from AlSiC at 200°C, (c) shows the difference image before registration, (d) shows the difference image after registration. The contrast of the source images was enhanced using ImageJ's "Enhance Contrast" command, to make the difference visible

| Temperature in °C | **30** | **200** | **300** | **400** | **300** | **190** | **50** |
|---|---|---|---|---|---|---|---|
| *Voxels* | $16x10^6$ | $16x10^6$ | $16x10^6$ | $16x10^6$ | $16x10^6$ | $16x10^6$ | $16x10^6$ |
| *Mean* | 88.87 | 87.57 | 88.00 | 94.24 | 87.47 | 90.90 | 83.40 |
| *StdDev* | 4.05 | 4.22 | 3.93 | 3.92 | 3.90 | 4.14 | 3.66 |
| *Min* | 13 | 7 | 13 | 21 | 18 | 12 | 14 |
| *Max* | 127 | 124 | 123 | 129 | 120 | 126 | 116 |
| *Mode* | 89 | 88 | 88 | 95 | 88 | 91 | 84 |

Table 7.1: Mean, standard deviation, minimum and maximum values, and mode of the intensities for the datasets of one thermal cycle.

were used, one for each intensity. As stated by Ibanez et al. [ISNC05]:

> 'Mutual information attempts to re-group the joint entropy histograms into a more 'meaningful' formation. An optimizer that minimizes the joint entropy seeks a transform that produces a small number of high value bins and a large majority of almost zero bins. Multi-modality registration seeks such a transform while also attempting to maximize the information contribution by the fixed and the moving images in the overall region of the metric.'

This re-grouping effect can be observed in Figure 7.3. Before registration the bins are ordered circularly while after registration they form several, clearly separated, ellipses.

## 7.2 Isovalue Error Determination

The 3D images acquired by synchrotron tomography at different temperatures exhibit slight variations in their intensities. Table 7.1 lists the mean, the standard deviation, minimum and maximum values, and the most occurring value, the mode, of the gray levels, together with the number of voxels for which these values were measured. A cuboid of $200x200x400$ voxels from non-registered images was used for this measurement.

This intensity variation regards the analysis, as one must choose which intensity levels to account for a void. In previous work [SRD06][SRD$^+$07b] an upper bound of $UB = 65$ for 8-bit encoded data was chosen for voxels related to a void ($0 \leq x \leq 65$). To calculate the influence of the upper bound on the resulting void volume we plotted the intensity vs. the volume (see Figure 7.5). The border at which the volume starts to include the matrix can not be decided. A better plot is the intensity $i$ vs. $P'(i)$, the difference quotient of the percentage volume change. The percentage change $P$ is
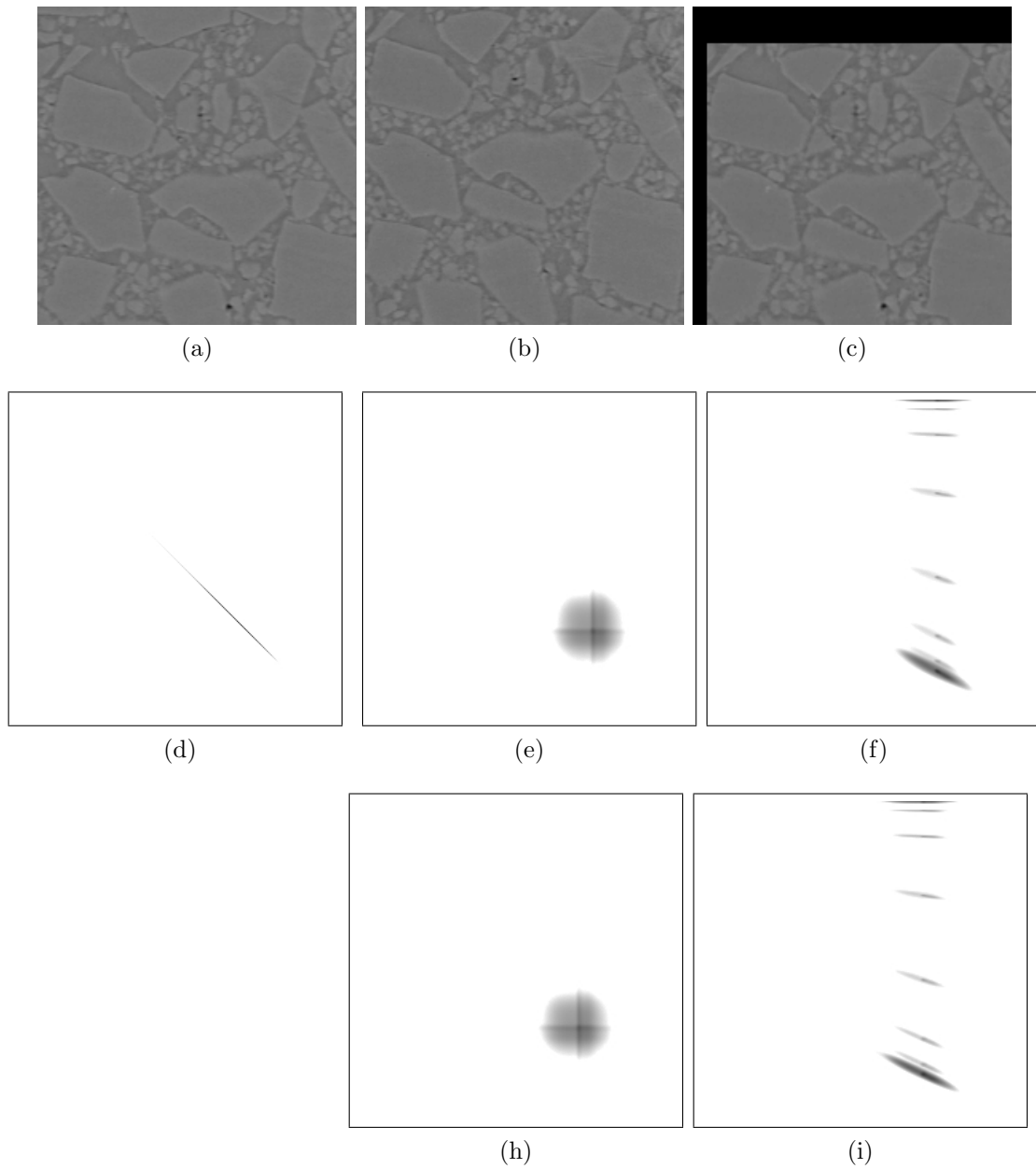
Figure 7.3: Joint entropy histogram: The first row shows from left to right a slice at 30°C (a), at 200°C unregistered (b) and at 200°C registered (c).
The second row shows the joint entropy histogram of the 30°C dataset with itself (d), with the 200°C (unregistered) data (e), and with the registered 200°C data (f).
The third row shows the joint entropy histogram of the 30°C dataset with the unregistered 50°C dataset (h) and with the registered 50°C dataset (i).

$$P = (\frac{V_{i+1} - V_i}{V_i}) * 100$$

with $V_i$ as the void volume at an intensity $i$. The central difference quotient $P'(i)$ is given by

$$P'(i) \approx \frac{P(i + h) - P(i - h)}{2h}$$

with $P$ being the relative void volume percentage change measured at a intensity $i$, and the difference $h$. Fig 7.4 shows the resulting curve. In the region of lower intensity values (x-axis, UB < 45) the volume percentage increases stochastically with rising intensities. The middle section ($\approx 45 \leq UB \leq 65$) shows a quite linear increase of volume percentage for every additionally included intensity value, while in the upper section (UB > 65) the volume percentage rises disproportionately with increasing upper bound threshold.
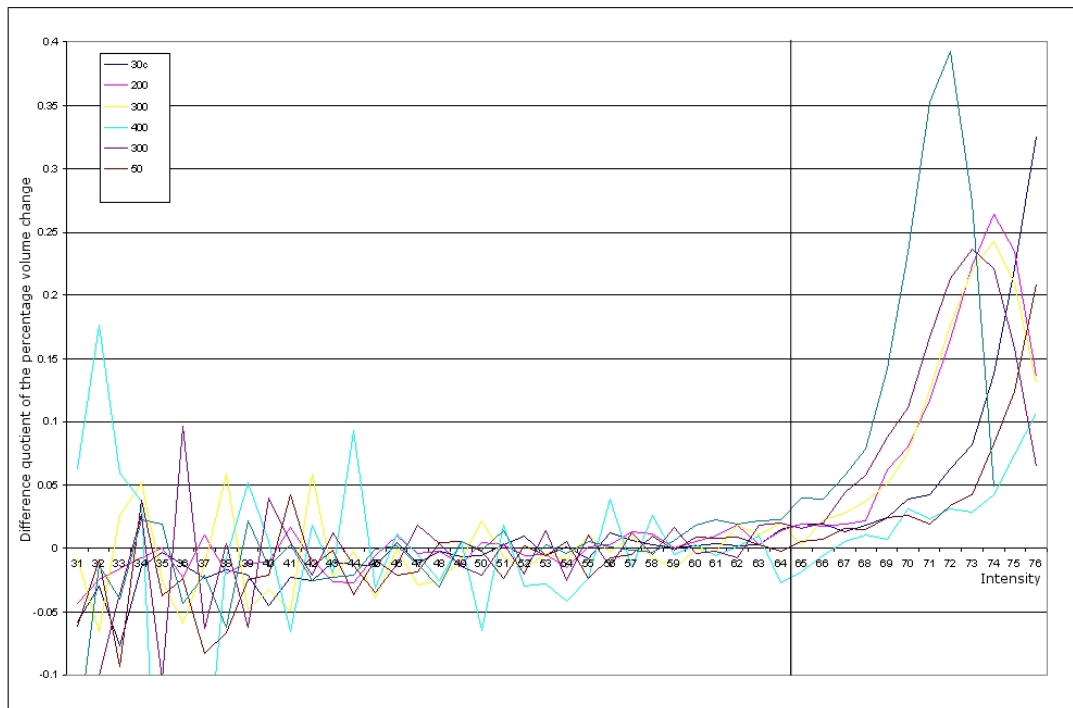


Figure 7.4: Upper threshold of void intensity vs. average rate of volume change. When the upper intensity threshold for voids is too high, the volume calculation includes the surrounding matrix and the resulting volume rises rapidly.

The linear rise of the relative volume percentage in the middle section shows that an

increase in the upper bound of the void threshold results in a proportional change of the void volume. At the intensity values at which the curve starts to rise disproportionately the calculated volume successively includes not only the voids but also the adjacent phases. A value of 65—as used in previous work [SRD06][SRD+07b]—can be considered sound. It is reasonably close to the beginning of the rise but still at the top of the middle section that exhibits linear grow.

Care should be taken at the imaging source that the scale for radiodensity stays constant between contiguous tomograms of the same sample. This is the only way to ensure the possibility of meaningful quantitative comparisons of images. As a workaround intensity rescaling could be used to match the image intensities, e.g., based on the mode, mean and standard deviation, utilizing the *a priori* knowledge of structural change. This needs further exploration.
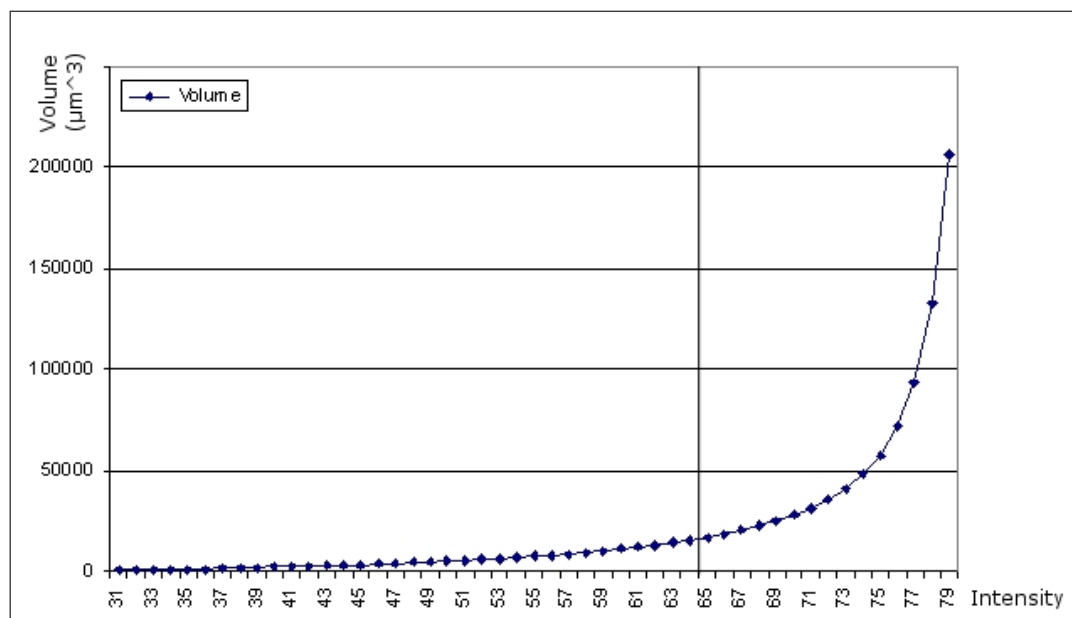


Figure 7.5: Upper threshold of void intensity vs. calculated void volume. While the volume at lower intensity thresholds incorporates only the voids, at higher thresholds parts of the matrix are included. Identifying a good threshold from this graph is not feasible, Figure 7.4 provides a better method for this.

# 8 Conclusion and Further Work

We presented a method for automated 3D image registration of *AlSiC* datasets acquired by synchrotron tomography. AlSiC is a *metal matrix composite* (MMC) that consists of aluminum reinforced with 70 vol.-% silicon-carbide particles. The volumetric datasets were registered using a similarity measure based on *mutual information* and a regular step gradient descent optimizer. Registration results were assessed visually. The registered images were used to analyze the shape change of a specific void during a thermal cycle between room temperature and 400°C. The results support the hypothesis of Schöbel et al. [SRD06] that the relative reduction of the CTE due to plastic flow closes the voids successively and that during cooling these voids re-open again keeping the CTE at a constant level. This behavior in void kinetics could be shown by visualizing the images of a void during one thermal cycle and analyzing quantitative results. Tools were developed for simple one-to-one comparison of two 3D datasets including 2D slice view, 3D adjustable volume rendering, 3D surface rendering, as well as overlaying 2D contours of an arbitrarily positioned cutting plane.

**Porting to 64-bit to avoid memory restrictions**    The current images have a spatial resolution of $1.6\mu m$, but higher resolution tomograms are needed for the study of similar materials. Currently the maximum size of datasets is limited due to memory restrictions imposed by the 32-bit architecture and the operating system. Porting the applications to a 64-bit architecture and OS would allow for registration and visualization of bigger 3D datasets, thus reducing the need for manual cropping of the datasets and enabling the processing of the whole sample at once.

**More robust registration**    The image registration metric is sensitive to noise. Enhancing the image prior to registration (e.g., by applying an edge-preserving smoothing filter) would possibly speed-up the registration process and reduce the chance that the registration process gets trapped in a local minimum before finding the correct transfor-

mation. Registration of *phase contrast tomograms* [PMea00] could enhance the state of phase and pore contours.

Further refinement may be achieved by evaluating the use of alternative components of the registration framework, e.g., replacement of the regular step gradient descent optimizer by the one-plus-one evolutionary optimizer. Ibanez et al. [ISNC05] describe it as 'naturally well-suited for optimizing the mutual information metric given its random and noisy behavior'.

# 9 Summary

This chapter shall serve as a concept for a scientific publication.

## 9.1 Introduction

AlSiC is a metal matrix composite where an aluminum matrix is reinforced with silicon-carbide particles. It is used for heatsinks and as base plates in components for high power electronics like *Insulated Gate Bipolar Transistors* (IGBT). It combines a high thermal conductivity with a low *coefficient of thermal expansion* (CTE). This reduces the thermally induced stresses compared to copper when soldered to a ceramic IGBT substrate. Synchrotron tomography revealed the existence of microscopic voids in AlSiC. These microvoids cause an anomaly in the thermal expansion curve that could not be explained by thermoelastic models [Hub03]. This is explained by closing of the voids by diffusion of the aluminum matrix above 200°C and opening during cooling. Volumetric images were acquired at several temperatures between room temperature and 400°C, but due to displacements between the tomograms comparisons were only possible at a macroscopic scale comprising many voids. It was not possible to observe the behavior of a single void in multiple images.

A major purpose of this work is to provide a software that is able to align the 3D images. This task is called 3D image registration. By registering the volumetric images it will be possible to find the exactly same void in various images. When the images are resampled using the transformation obtained by the registration it will be possible to find the location of a specific void by looking at the same coordinates in each image. Further software applications shall enable the user to easily analyze the behavior of a void in multiple images. Linked views enable a simple one-to-one comparison of registered slices and 3D renderings of a void. The possibility to superimpose the contour of a void at various temperatures will give further insight into the void kinetics during thermal

cycling and help to localize the debonding at the interfaces between matrix and the reinforcement.

## 9.2 Methods and Materials

### 9.2.1 Production and Application of AlSiC

The full designation of the MMC is AlSi7Mg/SiC/70p. It has been produced by gas pressure infiltration with 70 vol.-% trimodal SiC-particle preforms by Electrovac GmbH, Klosterneuburg, Austria. 55 vol.-% of the particles are of the FEPA [FoEPoA] F100 size class (149–105 $\mu m$), the others F600 and smaller than 5 $\mu m$. See Huber [Hub03] for a detailed analysis of the thermal expansion behavior of various aluminum-matrix composites. The shrinkage of Al during cooling after infiltration leads to plastic deformations in the matrix. 'Small voids in the MMC could be observed which are formed during cooling after the infiltration process' [SRD+07b].

AlSiC has superior properties compared to copper regarding the application as base plates in high power electronics. Its thermal expansion behavior is very similar to that of the substrate which it is mounted on, thus reducing the stresses on the solder joints. It has a low *coefficient of thermal expansion* (CTE) and a high thermal conductivity [Hub03].
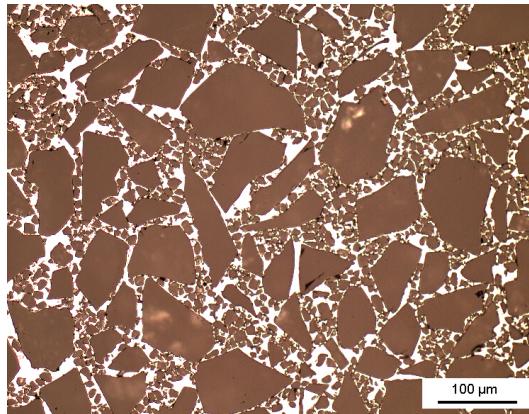


Figure 9.1: Light optical micrograph of AlSiC shows the trimodal particle distribution with a few dark voids in the bright matrix [SRD+07b] (image from Huber [Hub03])

## 9.2.2 Image Acquisition

3D images of the sample were acquired prior to this work by the *Institute of Materials Science and Technology* by means of *synchrotron tomography.*

Synchrotron tomography was done at the *European Synchrotron Radiation Facility* (ESRF) in Grenoble, France. Experiments were carried out on the *ID15A High Energy Scattering Beamline.*

Each volume consists of 750 slices of approximately 1024 x 1024 pixels. Actual resolutions vary slightly between tomograms. The pixel type is 32 bit float. One slice needs about 4 MB of diskspace, which accumulates to about 2.8 GB per volume (see Figure 9.2.

The voids reside in the low intensity region. The volume fraction of the voids is about 0.14% in as-cast condition. The high intensity region represents high absorbing Fe containing impurities, the heavier elements with an higher absorption coefficient.
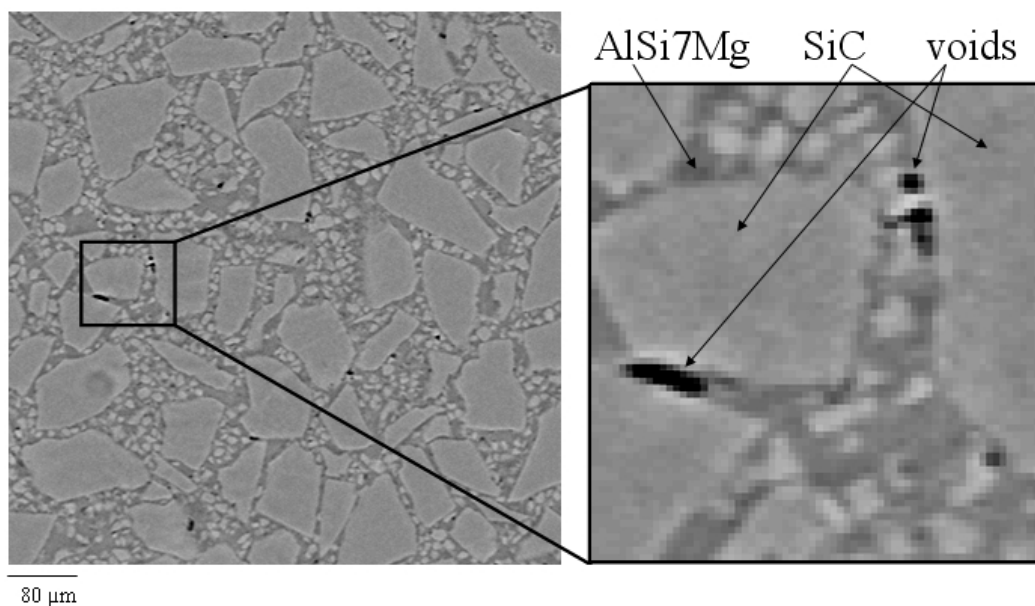


Figure 9.2: A detail from a slice acquired by synchrotron tomography (voxel size: $(1.6\mu m)^3$)

## 9.2.3 Registration

The movement of the sample between imaging needs to be corrected in order to be able to observe the same void over multiple images. Ibanez et al. [ISNC05] define image

registration as 'the process of determining the spatial transform that maps points from one image to homologous points on an object in the second image'.

We expect the sample to expand uniformly due to the isotropic material distribution, therefore opt for a *rigid* registration comprising only affine transformations. The *affine transformation* handles rotation, scaling, shearing and translation. The data are produced by the same imaging source, thus the registration can be categorized as a *single modal* registration. An image similarity based method—mutual information—measures the matching of the images. Ibanez et al. [ISNC05] describe the basic components of the ITK registration framework (see Figure 9.3) as follows:

'The *transform* component represents the spatial mapping of points from the fixed image space to points in the moving image space. The *interpolator* is used to evaluate moving image intensities at non-grid positions. The *metric* component provides a measure of how well the fixed image is matched by the transformed moving image. This measure forms the quantitative criterion to be optimized by the *optimizer* over the search space defined by the parameters of the *transform*.'
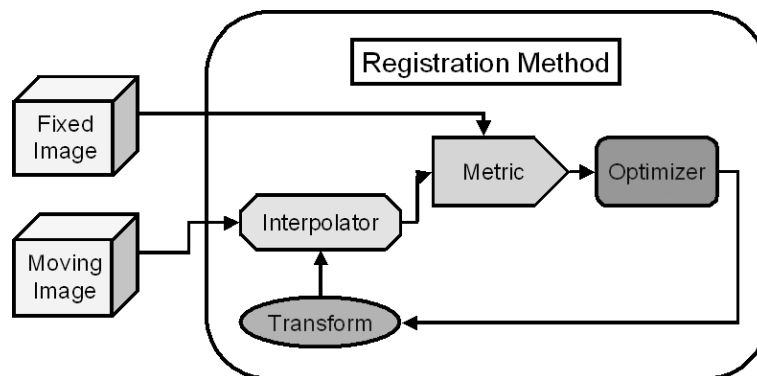


Figure 9.3: The basic components of the ITK registration framework are two input images, a transform, a metric, an interpolator and an optimizer [Kit05].

An image similarity based measure is used to assess the progress of the registration. The notion of *mutual information* (MI) is based on probability theory and information theory. Ibanez et al. [ISNC05] state: '*Mutual information (MI) measures how much information one random variable (image intensity in one image) tells about another random variable (image intensity in the other image).*'

Mutual information is defined in terms of entropy. Let

$$H(A) = - \int p_A(a) \log p_A(a) \, da \qquad (9.1)$$

be the entropy of random variable $A$, $H(B)$ the entropy of random variable $B$ and

$$H(A, B) = \int p_{AB}(a, b) \log p_{AB}(a, b) \, da \, db \tag{9.2}$$

be the joint entropy of $A$ and $B$. If $A$ and $B$ are independent, then

$$p_{AB}(a, b) = p_A(a) p_B(b) \tag{9.3}$$

and

$$H(A, B) = H(A) + H(B). \tag{9.4}$$

where $p_{AB}(a, b)$ is the joint probability distribution function of A and B while $p_A(a)$ and $p_B(b)$ are the marginal probability density functions.

However, if there is any dependency, then

$$H(A, B) < H(A) + H(B). \tag{9.5}$$

The difference is called Mutual Information : $I(A, B)$

$$I(A, B) = H(A) + H(B) - H(A, B) \tag{9.6}$$

The joint probability function $p_{AB}(a, b)$ is evaluated at discrete positions or bins uniformly spread within the dynamic range of the images. Entropy values are then computed by summing over the bins [ISNC05]. The number of bins has a great influence on the progress of registration, as the optimizer may get trapped at local minima. Prenormalization of the intensity values of the image is not necessary, as the metric rescales internally when building up the discrete density functions therefore being contrast insensitive.

Registration was done using the ITK implementation of the *Mattes Mutual Information* algorithm [MHV+01], which addresses registration as an optimization problem. The algorithm tries to maximize the mutual information between the two images by traversing the parameter space of the spacial transformation [OIK+07]. The mutual information registration algorithm described by Mattes [MHV+01] uses only one set of intensity samples which 'results in a much smoother cost function and hence allows the use of more intelligent optimizers' [ISNC05] compared to the initial implementation proposed by Viola and Wells [VWMW97]. Ibanez et al. [ISNC05] propose to remove the

major components of misalignment before applying the affine transform to reduce the computational costs. In our data translation is the major form of misalignment. Rotation and scaling are minimal compared to the translation. We provide the user with an adaptable overlaid *maximum intensity projection* (MIP) of highly absorbing iron precipitates of the two images to assist a manual initialization of the translation. This greatly reduces the initial misalignment. The iron precipitates in Al-alloys originated in the production process.

Best results could be achieved using multi-resolution registration. In general multi-resolution improves registration speed, accuracy and robustness of the image registration by eliminating local optima at coarser scales [ISNC05]. Registration is first done using down-sampled images. These are created by applying a Gaussian smoothing followed by subsampling. Once the convergence criterion defined by the *minimum step length* is reached, registration continues on the next level. The *minimum step length* is reduced by a factor of ten every level. The resulting transformation is applied to the next finer image as the initial transformation. The number of iterations per level is limited by the *number of maximum iterations*. Multiple stages of down-sampling can be combined to form an *image-pyramid*. Using the multi-resolution approach allows for registering images with more displacement than it would be possible without. On the downside, registration takes longer due to the involved down-sampling and needs additional RAM for the down-sampled images.

### 9.2.4 Registration Implementation

**The Coding Environment**

The target audience is currently using Microsoft Windows XP. Therefore all programs were written for MS Windows in C++ using Microsoft Visual Studio 2005. The operating system is Microsoft Windows XP SP2 running on an AMD Athlon 64 XP Dual (2.41 GHz) equipped with 2.00 GB of RAM. Some libraries were statically linked and merged into the executable. While this results in bigger executables (being 5-7 MB) it was favored due to the ease of the installation and distribution. C++ was chosen because of the vast amount of freely available software libraries, its speed and efficiency.

**External Libraries**   Intensive use was made of external libraries for visualization, segmentation, registration, and for the graphical user interface. All libraries are open source

software and available for multiple operating systems. Although portability was no major design decision it may be possible to port the applications to other platforms without major redesign. The main components on which the software is based are Qt for the GUI, ITK for the registration components, and VTK for the visualization.

The *Insight Segmentation and Registration Toolkit (ITK)* [ITK] was used for registration. ITK is a cross-platform (Unix, Windows and MacOS X) open source software system that was designed for performing registration and segmentation. The focus of the toolkit is mainly on medical applications, but not limited to these. It provides data representation and algorithms, but does not address visualization. It supports multithreaded (shared memory) parallel processing. A pipeline is used to connect data objects with filters. ITK is written in C++ using generic programming and 'many advanced design patterns'[ITK].

The *Visualization ToolKit (VTK)* [VTK] was used for visualization. VTK is an open source software for 3D computer graphics, image processing, and visualization. It is written in C++ and available for Windows, Unix-based platforms and Mac OS X. VTK includes visualization algorithms for scalar, vector, tensor, texture, and volumetric data and advanced modeling techniques like implicit modeling, polygon reduction, mesh smoothing, cutting, contouring, and Delaunay triangulation [VTK].

**Implementation Details**

An application for registering the tomograms using mutual information was developed, incorporating specific features like a manual *maximum intensity projection* (MIP) based linear pre-alignment. Several parameters regarding the registration can be adjusted:

- Initial translation
- Number of levels (multi-resolution image pyramid)
- Number of maximum iterations (per level)
- Number of histogram bins
- Number of spatial samples
- Maximum step length
- Minimum step length

These are set to initial values which are likely to give a satisfactory registration result. Once it is known that the datasets register properly, it is possible to speed up the

registration process by tuning the parameters, e.g., reducing the number of levels, the maximum step length, etc.

### Initial translation

By using the MIP to visualize only highly absorbing iron precipitates in the fixed and moving image it is possible to easily reduce the initial offset by aligning clearly distinguishable patterns. In the initial view it is possible to align on the x-y axis, by tilting the camera 90° upward this can be done for the x-z axis. The resulting 3D translation is fed as the initial translation into the registration algorithm, reducing the number of required iterations. It is possible to avoid the use of multi-resolution image-pyramids by pre-aligning the datasets to an accuracy within the capture radius of the registration system, greatly reducing the overall time, as this avoids the costly rescaling. The maximum displacement for a single layer registration to align without using multi-resolution registration is about 15 to 25 voxels in our case.

### Number of levels (multi-resolution image pyramid)

For the multi-resolution registration the number of levels can be set. Registration then starts on a coarser, down-sampled image and continues on a higher resolution image using the previous results. Each additional level reduces the image by a factor of two. Good results were achieved without multi-resolution registration ($levels = 1$) when the initial misalignment was in the range of 15 to 25 voxels for a cube of $200^3$ voxels. Multi-resolution registration allows for registration of volumes with a greater initial misalignment without the need for a manual pre-alignment. Note that the down-sampled images might get too small to be useful when the number of levels is too high (e.g., $levels = 4$ for a cube of $200^3$ voxels).

### Number of maximum iterations

The number of maximum iterations specifies when the algorithm should stop at the latest. It might stop earlier when no further optimization is possible using the defined minimum step length. As a rule of thumb, we start using a low number (e.g., 50 iterations), we watch the output of the registration for the metric value as it should gradually get lower as the optimizer treats the mutual information as a cost function and tries to minimize it. When the metric value decreases consistently a good set of parameters

is found. We then increase the number of maximum iterations in order to allow the optimizer reaching the desired precision tolerance. 500 – 3000 iterations proved to be a sane upper limit for a registration with a repeat accuracy in the sup-pixel range. If the desired precision can not be reached (e.g., due to the optimizer being trapped in a local minimum) the algorithm will stop after the number of maximum iterations.

### Number of histogram bins

Using less bins for the histogram results in faster calculation but the quantizing reduces the available information. As the input data is expected to be 8-bit per pixel, a value of 256 ($= 2^8$) bins gives the best results, while smaller values (128 bins, 64 bins, ...) result in faster registration.

### Number of spatial samples

The Mattes et al. [MHV$^+$01] mutual information algorithm uses an arbitrary number of samples for the calculation of mutual information. While a larger number improves the smoothness of the metric from one iteration to the next this results in a longer calculation time per evaluation of the metric. According to Ibanez et al. [ISNC05] 'the number of spatial samples can usually be as low as 1% of the total number of pixels in the fixed image'. We were able to register 3D images using about 1% (or more) of the pixels as spatial samples.

### Maximum step length

The maximum step length sets the first step size. Setting this too small results in a larger number of iterations, increasing computation time. For the initial alignment setting it to 1 is a good starting point, but it can be reduced (e.g., to 0.1) once the rough initial misalignment is known and considered in the initial translation.

### Minimum step length

The minimum step length represents the convergence criterion. 'This is equivalent to defining the precision with which the final transform should be known' [ISNC05].

| Number of voxels | 400*400*400 | 400*400*400 | 400*400*400 |
| --- | --- | --- | --- |
| Initial translation (voxels) | (0, 10, 0) | (10, 23, 4) | (12, 25, 5) |
| Levels | 1 | 1 | 1 |
| Maximum iterations | 500 | 500 | 5000 |
| Histogram bins | 256 | 256 | 256 |
| Spatial samples | 1.10% (700,000) | 1.10% | 1.10% |
| Maximum step length | 1 | 0.1 | 0.1 |
| Minimum step length | **0.1** | **0.001** | 0.000001 |
| Final translation (voxels) | (10.9, 23.2, 4.4) | (12.40, 24.81, 5.25) | (12.47, 24.83, 5.31) |
| Actual iterations | 23 | 185 | 540 |
| Final metric value | **-0.76** | **-1.07** | -1.07 |
| Time | 87 sec. | 365 sec. | 932 sec. |

Table 9.1: Registration results using manual pre-alignment. A lower negative metric value denotes a better correlation.

**Optimizer parameter scaling**

The scale of units in translation is different from rotation: the rotation is measured in radians, the translation is millimeters. The units need to be rescaled for the optimizer to work properly. The scales for translation is set to 0.000001, the other matrix coefficients are set to 1. These values are specific to our datasets and account for the fact that the major form of misalignment is a linear displacement with only a minimal rotation.

**Example Parameter Set and Registration Results**

The following set of parameters is from a successful registration. It shows how roughly aligned datasets quickly converge. The manual alignment was performed as described in Section 5.4.2 by aligning the projected iron precipitates. It took only 23 iterations to reach an alignment where no major dislocation is easily visible. Registration applying a stricter convergence criterion (e.g., setting the minimum step length to 0.001) takes longer, but produces a visually "perfect" result (no visible dislocation at the borders of the checkerboard comparison).

Registration without manual pre-alignment also gives very good results, while taking substantially longer.

**Interpreting the Mutual Information Metric Value**

Calculating the mutual information metric value for the fixed image with itself gives $-2.35$, the value before registration of two distinct images is $> -0.001$. These values

| Number of voxels | 400*400*400 |
|---|---|
| Initial translation (voxels) | (none) |
| Levels | 3 |
| Maximum iterations | 500 |
| Histogram bins | 128 |
| Spatial samples | 1.1% (700,000) |
| Maximum step length | 1 |
| Minimum step length | 0.1 |
| Final translation (voxels) | (12.40, 24.79, 5.28) |
| Actual iterations | 334 (in the last level) |
| Final metric value | -0.98 |
| Time | 1416 sec. |

Table 9.2: Results for registration without manual pre-alignment.

| Filesize | 15.2 MB | 61 MB |
|---|---|---|
| Resolution (voxels) | 200*200*200 | 400*400*400 |
| Memory requirement | 250 MB | 950 MB |

Table 9.3: Memory requirement for registration.

differ only slightly between the various AlSiC images.

The metric value is negative, as the optimizer treats it as cost function and tries to minimize it. Small metric values denote a better fit. When a registration fails due to the optimizer getting trapped in a local minimum (because of a large misalignment) typical metric values are larger than $-0.01$ ($-0.01 < metricvalue < 0$).

A registration that was able to align the images results in metric values considerably smaller than $-0.2$ to $-0.4$.

**Memory issues**

3D Registration is a memory intensive task. Registering two cubes of $400^3 = 64,000,000$ voxels (8-bit data) uses $> 900MB$ of RAM. One reason is that registration internally uses data types of higher precision, typically floating point variables, another one that the image is held in memory multiple times by the processing pipeline. We use short integers as the internal data type for registration, which greatly reduces the memory footprint, and enables registering volumes of $400^3$ voxels on a computer running 32-bit Windows XP.

### 9.2.5 Limitations

**32-bit Restrictions**   Currently the maximum size of datasets is limited due to memory restrictions imposed by the 32-bit operating system. For 32-bit systems the maximum addressable memory is 4 GB. Windows XP itself reserves 2 GB for its own use, leaving 2 GB of (virtual) memory for other applications [Micc]. The */3GB switch* in the file *boot.ini* [Micb] allows applications to address 1 GB of additional virtual address space above 2 GB, if they define IMAGE_FILE_LARGE_ADDRESS_AWARE in the process header. Our applications do make use of this in the header.

The limiting factor indeed seems to be fragmentation of virtual address space—not being able to allocate enough contiguous memory blocks although using well below 2 GB / 3 GB of virtual memory. We did not try to solve the issue of memory fragmentation, but further explanations of the Win32 memory handling and possible solutions can be found on the Microsoft website [Mica].

Porting the applications to a 64-bit OS would void the current memory limits and allow for registration and visualization of bigger 3D images, thus reducing the need for manual cropping of the datasets and enabling processing of the whole sample dataset at once.

**Large Data Sets in ITK / VTK**   Hints on how to handle large data sets in VTK can be found on the VTK website [Wik]. In memory-critical parts of the applications, we use the proposed *ReleaseDataFlag* flag to avoid that intermediate results are kept between the filters in the pipeline. This forces filters to release the memory after data objects have been processed, and reduces the memory footprint of the application. The downside is, that the filters have to re-execute if a parameter has changed, costing CPU cycles.

### 9.2.6 Visualization Applications

After the volumetric images have been registered to each other, it is possible to compare a specific feature in images acquired at different temperatures. We developed an application for comparison of two 3D images. Linked views for 2D and 3D comparison help to analyze the alterations occurring during a thermal cycle. A tool for comparison of multiple void contours by overlaying has been developed. Isolines are plotted along

an arbitrarily positioned cutting plane. Contours of multiple volumetric datasets can be overlaid.

## 9.3 Results

We were able to visualize the shape change of a single void during one heating cycle. Figure 9.4 shows the isosurface rendering of a void during cycling $30°C \rightarrow 200°C \rightarrow 300°C \rightarrow 400°C \rightarrow 300°C \rightarrow 190°C \rightarrow 50°C$. The void shrinks and splits due to the thermal expansion of the surrounding aluminum matrix. When the sample is cooled, the void re-opens and recaptures its original shape with some minor spheroidization. Most changes take place between 300°C and 400°C.
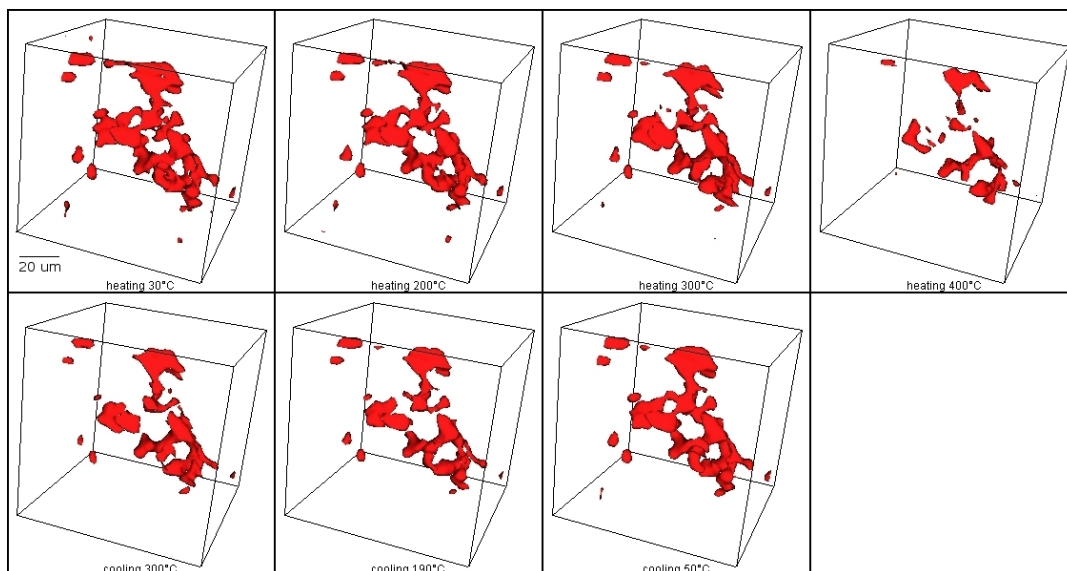


Figure 9.4: The change of a void during the thermal cycle $30°C \rightarrow 200°C \rightarrow 300°C \rightarrow 400°C \rightarrow 300°C \rightarrow 190°C \rightarrow 50°C$ (image order: left to right, top to bottom).

It was shown by Schöbel et al. [SRD⁺07b] how the void volume-fraction changes under thermal load and this is assumed to cause the anomaly in the *coefficient of thermal expansion*. Figure 9.5 plots the volume fraction of a $(480\mu m)^3$ cube over time during a heating cycle. The datasets had not registered for this measurement, thus a representative volume had to be used.

By aligning the datasets using the 3D image registration approach presented in this thesis, it is for the first time possible to compare the volume of a single void at different temperatures and visually observe the influence of the expanding matrix on the void
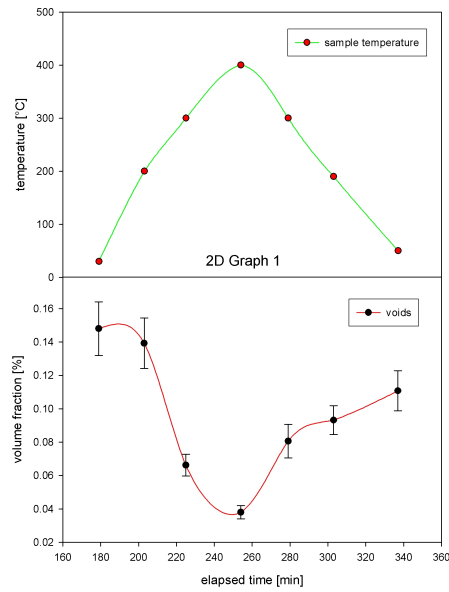
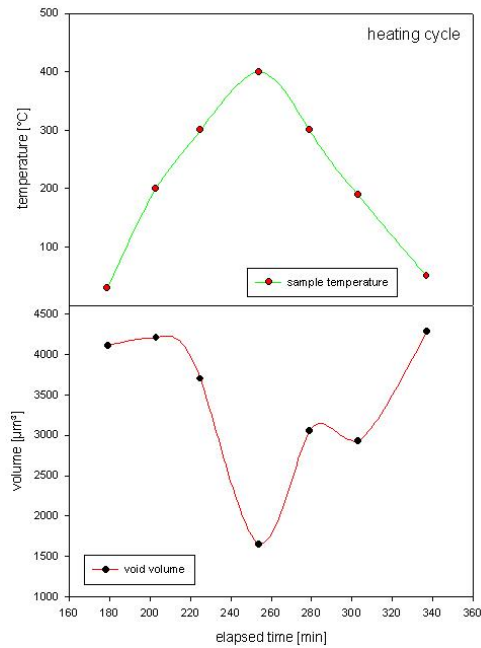Figure 9.5: Change of void volume-fraction of cube (side=480 $\mu m$).

Figure 9.6: Change of volume of a single void.

shape. The plot of the volume of an individual void over time during a heating cycle can be found in Figure 9.6 and Figure 9.7. The behavior of this randomly selected individual void resembles the behavior of the whole cube. The biggest transformation is apparent above 200°C during heating.

It is now possible for us to analyze the change in surface area and calculate a shape factor for a specific void, as depicted in Figure 9.7. A shape factor of one denotes a perfect sphere, whereas higher numbers indicate the deviation from a sphere (see Section 5.4.3 for the calculation of the shape-factor). That means that during heating some spheroidization takes place which is only partly reversed during cooling.

Analysis is further supported by superimposing multiple void contours acquired at different temperatures. Figures 9.8 and 9.9 provide examples for such plots. It is shown that the void shrinks and splits into two small sections when heated to 400°C.
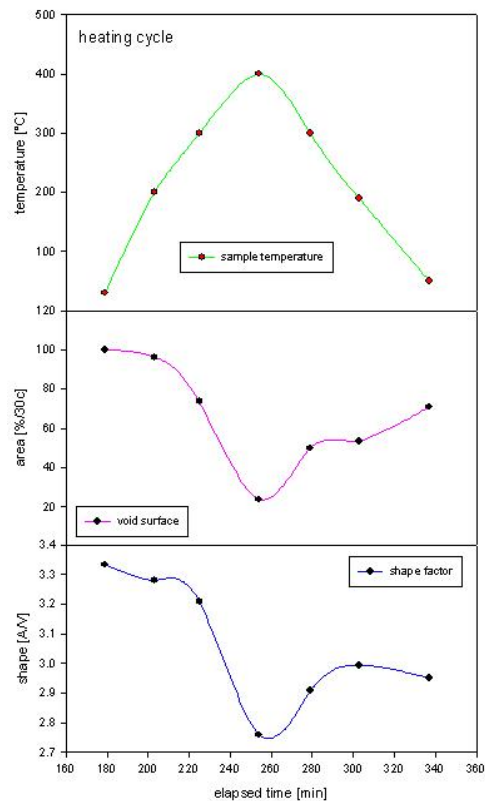
Figure 9.7: Change of area and shape of a single void.

## 9.4 Evaluation

### 9.4.1 Checkerboard-based comparison

To evaluate the result of the image registration a checkerboard comparison tool is part of the registration application. The checkerboard comparison provides borders at which the interleaved pictures are adjacent and their transitions can be compared. Visual assessment of the registration outcome is feasible, as misalignments of the SiC-particles can be spotted at the borders and corners of the resulting checkerboard. Panning, zooming, and changing slices facilitates the inspection. Figure 9.10 (right) is an example of a checkerboard comparison of a successful registration, while Figure 9.10 (left) shows the two interleaved datasets before registration.

### 9.4.2 Difference Images

Difference images show the deviation of two registered images by calculating the absolute difference between the values of pixel pairs. As our datasets are noisy and only a small range of intensity values is dominant it is difficult to show meaningful difference images. The source images in Figure 9.11 have been contrast enhanced using ImageJ [Ras07] before calculating the difference image. We also tried to reduce the noise using Gaussian smoothing and median filtering, but this did not give noteworthy better results.

### 9.4.3 Joint Entropy Histogram

Figure 9.12 displays the *joint entropy histogram* of two images and shows that the mutual information increased by registering the images. The joint entropy histogram is given by

$$f_{ij} = -p_{ij} \log_2(p_{ij}) \tag{9.7}$$

where the indices $i$ and $j$ identify the location of a bin in the joint entropy histogram of the two images and are in the ranges $i \in [0 : N - 1]$ and $j \in [0 : M - 1]$. The image $f$ representing the joint entropy histogram has $NxM$ pixels because the intensities of image $A$ are quantized into $N$ histogram bins and the intensities of image $B$ are quantized into $M$ histogram bins. The probability value $p_{ij}$ is computed from the frequency count of the histogram bins.

$$p_{ij} = \frac{q_{ij}}{\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} q_{ij}} \tag{9.8}$$

The value $q_{ij}$ is the frequency of a bin in the histogram and it is computed as the number of pixels where image $A$ has intensities in the range of bin $i$ and image $B$ has intensities in the range of bin $j$. The value $p_{ij}$ is therefore the probability of the occurrence of the measurement vector centered in the bin $ij$ [ISNC05]. The joint entropy histograms in Figure 9.12 were computed using 256 bins for each dataset.

## 9.5 Conclusion

We presented a method for automated 3D image registration of *AlSi7Mg/SiC/70p* (AlSiC) datasets acquired by synchrotron tomography. AlSiC is a metal matrix composite (MMC) that consists of aluminum reinforced with 70 vol.-% silicon-carbide particles with a trimodal size distribution. The volumetric datasets were registered to sub-pixel

precision using a similarity measure based on *mutual information* and a regular step gradient descent optimizer. Registration results were assessed visually. The comparison can be performed for voids of any size due to the complete datasets being aligned. The location of the voids can be identified, as well as the direction of the maximal dimensional change (e.g., the interface between SiC and AL matrix). The registered images were used to analyze the shape change of a specific void during a thermal cycle between room temperature and 400°C. The results support the hypothesis of Schöbel et al. [SRD06] that the relative reduction of the CTE due to plastic flow closes the voids successively and that when cooling from a relaxed stress condition at 400°C voids open again keeping the CTE at a constant level. This behavior in void kinetics could be shown by visualizing the image of a void during the thermal cycle and analyzing quantitative results. Tools were developed for simple one-to-one comparisons of two 3D datasets including 2D slice view, 3D adjustable volume rendering, 3D surface rendering, as well as overlaying 2D contours of an arbitrarily positioned cutting plane. Further examples should be evaluated to draw more statistical conclusions on the change of volume and shape, and to assess the reversibility during thermal cycling.
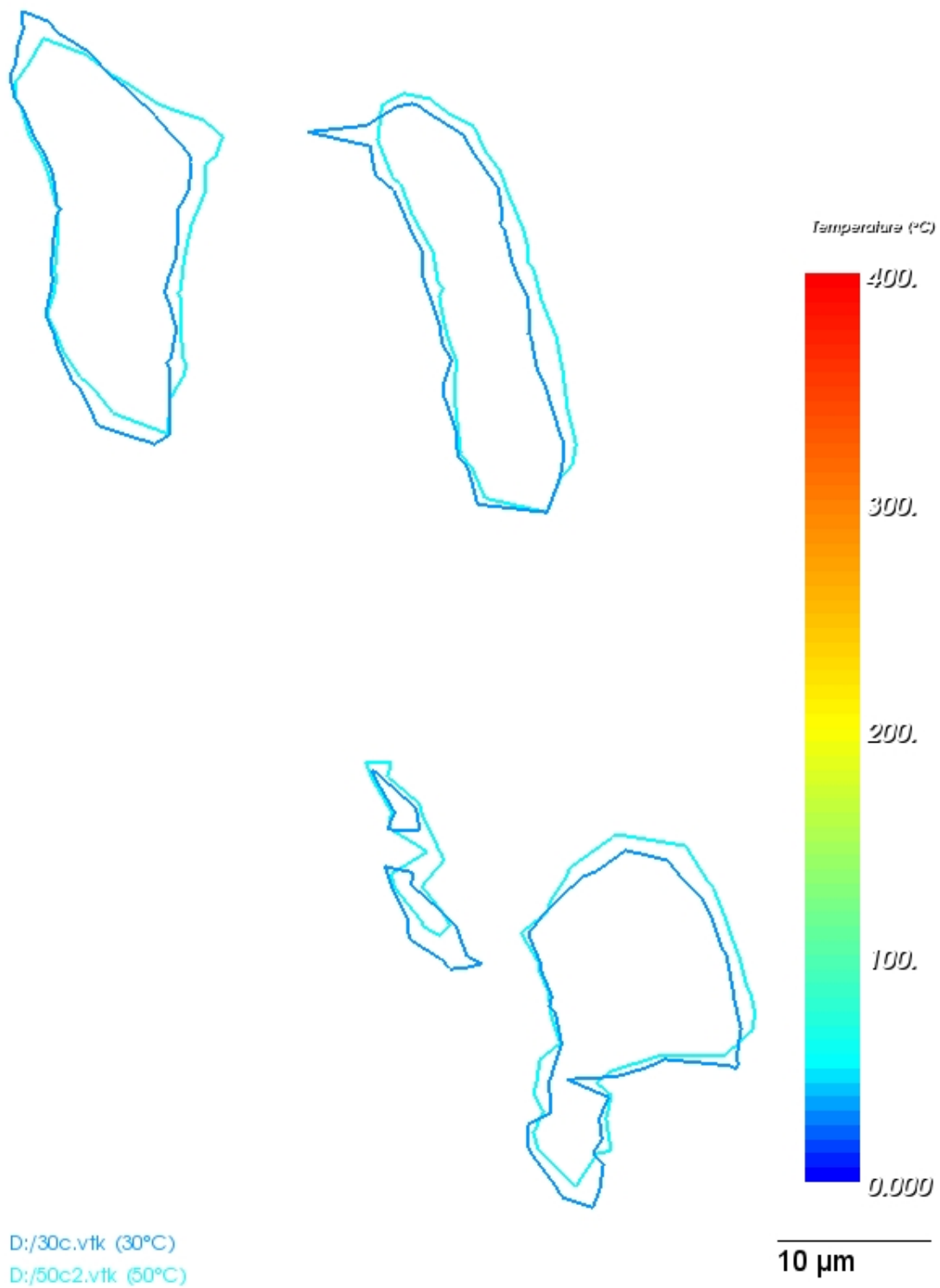
Figure 9.8: Void contours before (30℃) and after (50℃) heating. The void reopens to its initial size with little spheroidization.
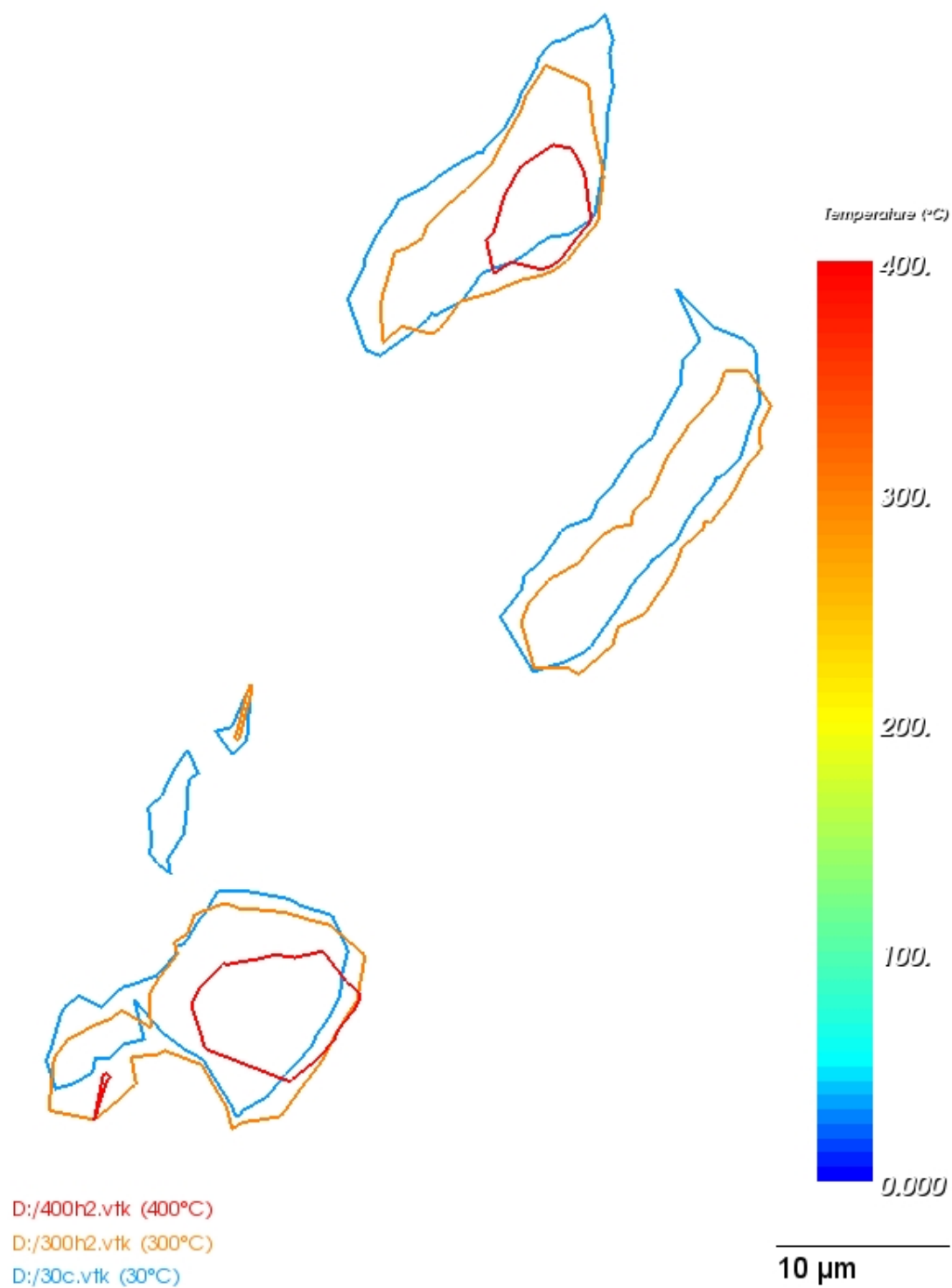
Figure 9.9: Void contours at 30℃, 300℃, and 400℃. Void 1 shrinks. Voids 2–4 disappear below resolution above 300℃. Void 5 shrinks and splits into two small sections close to 400℃.
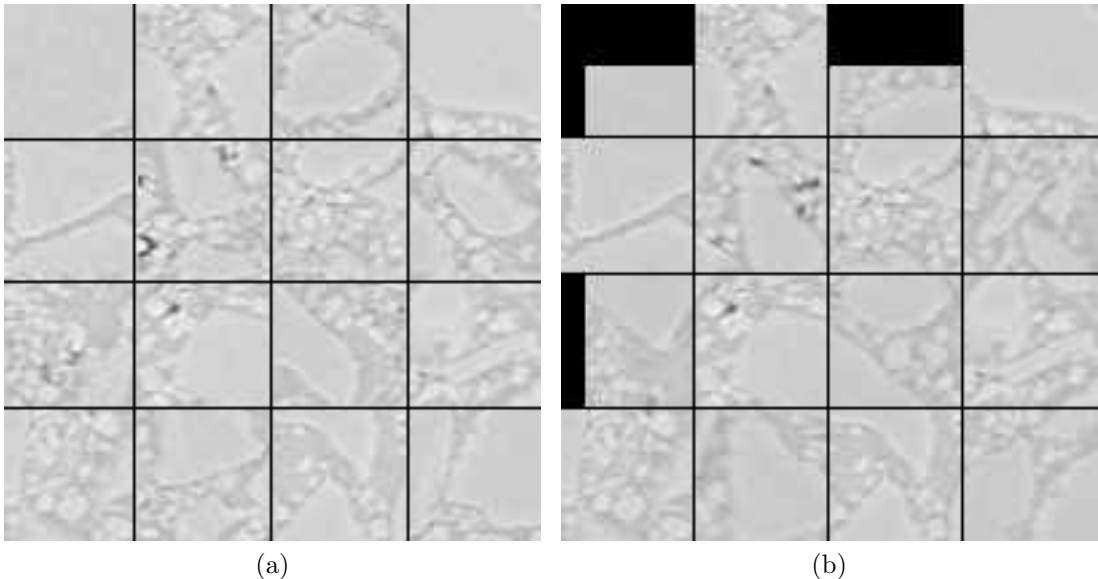
(a) (b)

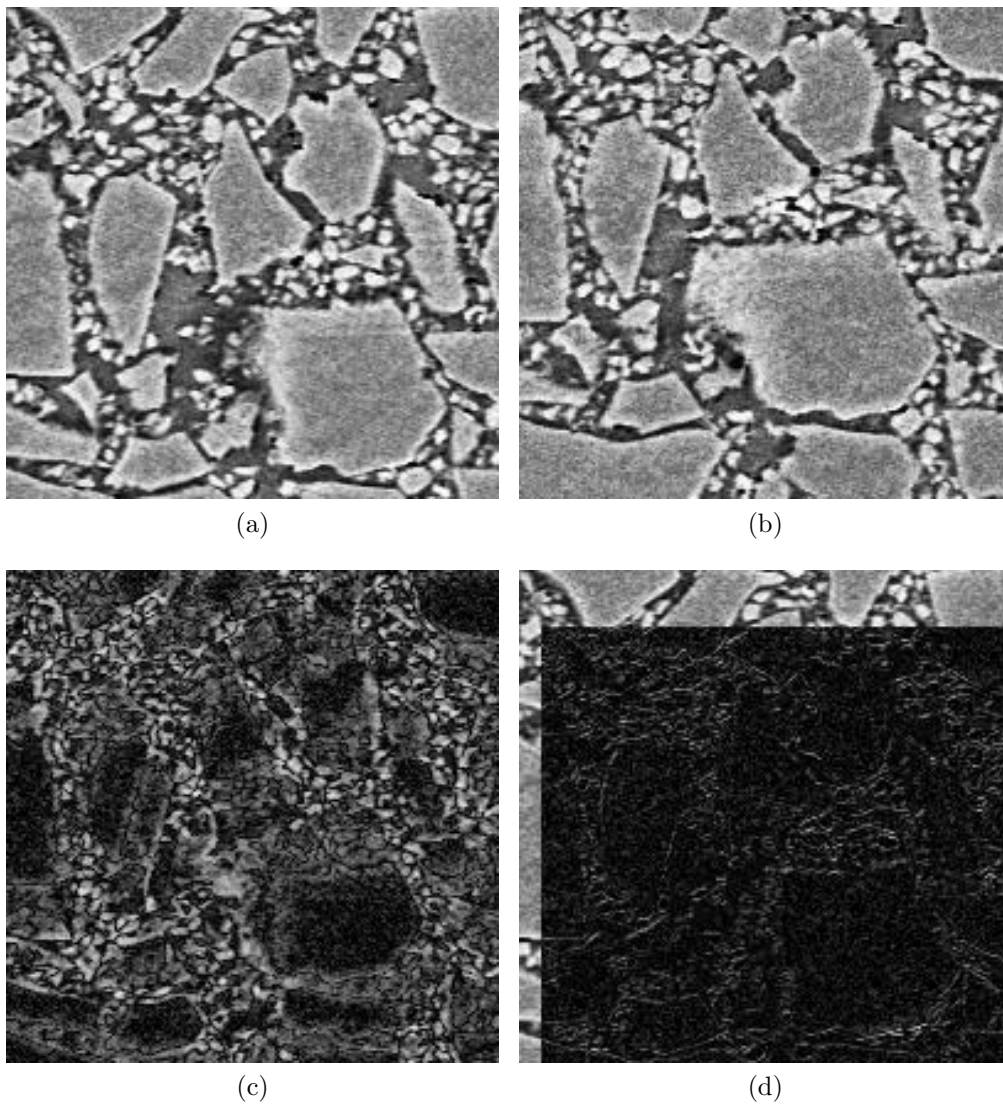Figure 9.10: Checkerboard comparison: 30°C and 200°C before (a) and after (b) registration.

Figure 9.11: Registration results: (a) displays a slice from AlSiC at 30°C, (b) displays a slice from AlSiC at 200°C, (c) shows the difference image before registration, (d) shows the difference image after registration. The contrast of the source images was enhanced using ImageJ's "Enhance Contrast" command, to make the difference visible
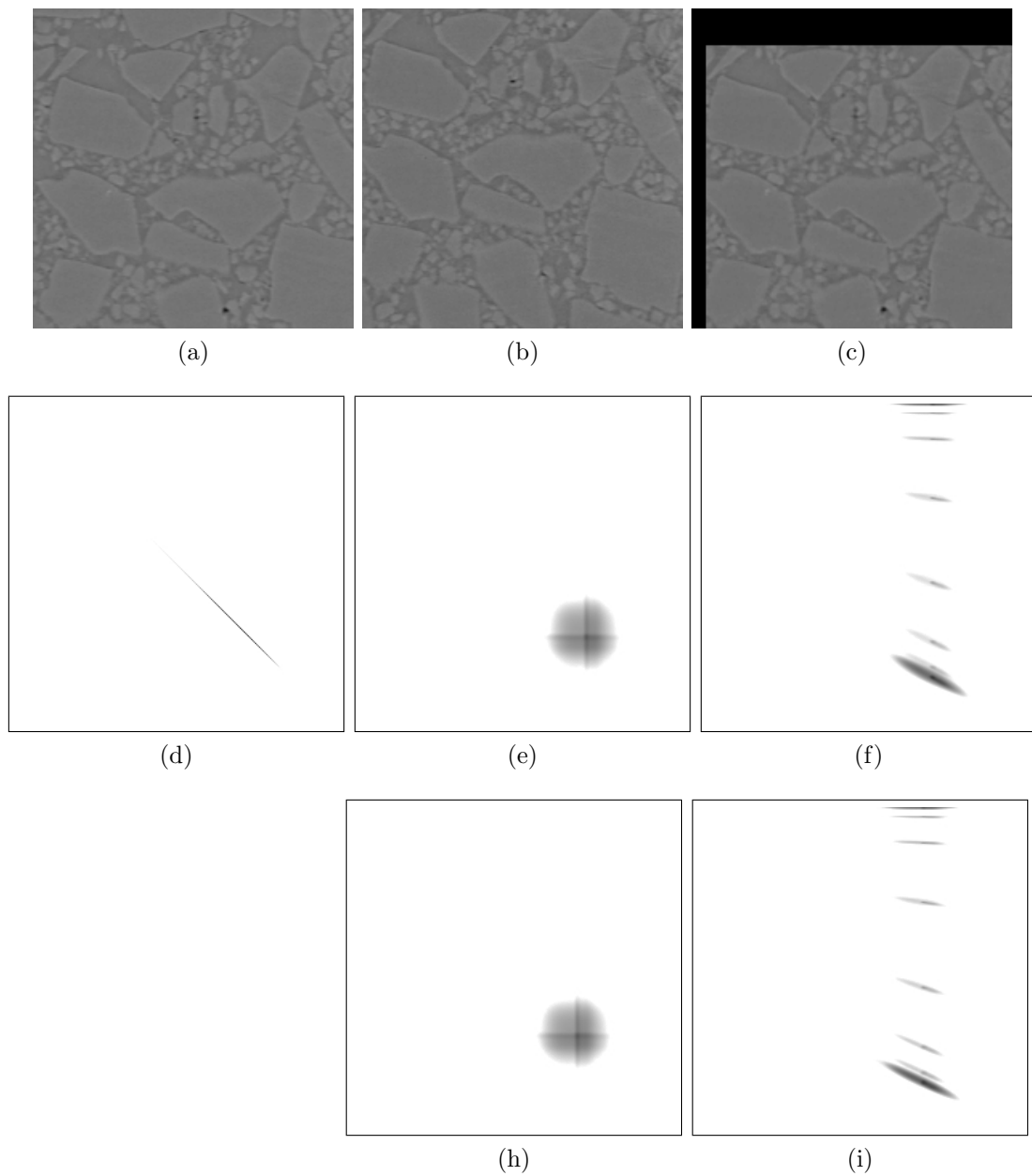
Figure 9.12: Joint entropy histogram. Mutual information attempts to re-group the joint entropy histograms into a more 'meaningful' formation. An optimizer that minimizes the joint entropy seeks a transform that produces a small number of high value bins and a large majority of almost zero bins. Multi-modality registration seeks such a transform while also attempting to maximize the information contribution by the fixed and the moving images in the overall region of the metric [ISNC05]. The first row shows from left to right a slice at 30°C (a), at 200°C unregistered (b) and at 200°C registered (c).
The second row shows the joint entropy histogram of the 30°C dataset with itself (d), with the 200°C (unregistered) data (e), and with the registered 200°C data (f).
The third row shows the joint entropy histogram of the 30°C dataset with the unregistered 50°C dataset (h) and with the registered 50°C dataset (i).

# List of Figures

# List of Tables

# Bibliography

[ALIF94]    Abdalmajeid M. Alyassin, Jack L. Lancaster, J. Hunter Downs III, and Peter T. Fox. Evaluation of new algorithms for the interactive measurement of surface area and volume. *Medical Physics*, 21(6):741–752, 1994.

[Amb73]    James Ambrose. Computerized transverse axial scanning (tomography): 2. Clinical application. *The British Journal of Radiology*, pages 1023–1047, December 1973.

[BC06]    David Bradley and Dudley Creagh. *Archaeology and Cultural Heritage.* Elsevier, 2006. ISBN 0444521313.

[Boo]    Boost C++ Libraries. `http://www.boost.org/`. [Online; accessed 31-January-2008].

[CVSM95]    André Collignon, Dirk Vandermeulen, Paul Suetens, and Guy Marchal. 3D Multi-Modality Medical Image Registration Using Feature Space Clustering. In *CVRMed '95: Proceedings of the First International Conference on Computer Vision, Virtual Reality and Robotics in Medicine*, pages 195–204, London, UK, 1995. Springer-Verlag.

[CW93]    T. W. Clyne and P. J. Withers. *An Introduction to Metal Matrix Composites.* Cambridge University Press, 1993. ISBN 0521483573.

[DCH88]    Robert A. Drebin, Loren Carpenter, and Pat Hanrahan. Volume rendering. In *SIGGRAPH '88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, pages 65–74, New York, NY, USA, 1988. ACM. ISBN 0897912756.

[ESR]    ESRF. European Synchrotron Radiation Facility. `http://www.esrf.eu`. [Online; accessed 31-January-2008].

[FoEPoA]    Federation of European Producers of Abrasives. Aluminium Oxide and Silicon Carbide Grains. `http://www.fepa-abrasives.org/DesktopDefault.aspx?portalname=www.fepa-abrasives.`

org&language=E&folderindex=0&folderid=3&headingindex= 5&headingid=80&tabindex=0&tabid=272. [Online; accessed 14-March-2008].

[Ger]   German Cancer Research Center, Division of Medical and Biological Informatics. `http://www.dkfz.de/en/mbi/index.html`. [Online; accessed 15-March-2008].

[GLD03]   Ulrike Göbel, Guy Lefranc, and Hans-Peter Degischer. Non-destructive Investigation of AlSi7Mg/SiC70p MMCs by means of Acoustic Scanning Microscopy, X-Ray and Neutron Diffraction. *Materialwissenschaft und Werkstofftechnik, WILEY-VCH Verlag GmbH & Co. KGaA, Weinheim*, 34(4):375–380, 2003.

[Grö02]   Eduard Gröller. Insight into Data through Visualization. In *GD '01: Revised Papers from the 9th International Symposium on Graph Drawing*, pages 352–366, London, UK, 2002. Springer-Verlag.

[Gui]   VTK Users's Guide. File Formats for VTK Version 4.2. `http://www.vtk.org/pdf/file-formats.pdf`. [Online; accessed 31-January-2008].

[HA73]   Godfrey Newbold Hounsfield and James Ambrose. Computerized transverse axial tomography. *The British Journal of Radiology*, 46(542):148–149, February 1973.

[HDLS06]   Thomas Huber, Hans-Peter Degischer, Guy Lefranc, and Theodor Schmitt. Thermal expansion studies on aluminium-matrix composites with different reinforcement architecture of SiC particles. *Composites Science and Technology*, 66:2206–2217, 2006.

[HMMW03]   Runzhen Huang, Kwan-Liu Ma, Patrick McCormick, and William Ward. Visualizing Industrial CT Volume Data for Nondestructive Testing Applications. In *VIS '03: Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*, page 72, Washington, DC, USA, 2003. IEEE Computer Society.

[Hou73]   Godfrey Newbold Hounsfield. Computerized transverse axial scanning (tomography): Part I. Description of the system. *The British Journal of Radiology*, 46:1016–1022, December 1973.

[Hub03]   Thomas Huber. *Thermal Expansion of Aluminium Alloys and Composites*. PhD thesis, Vienna University of Technology, 2003.

[Ima]         ImageJ Plugins: 3D Toolkit: 3D IO. `http://ij-plugins.sourceforge.net/3d-toolkit/#ref_3d-io`. [Online; accessed 14-March-2008].

[ISNC05]      Luis Ibáñez, William Schroeder, Lydia Ng, and Joshua Cates. *The ITK Software Guide*. Kitware, Inc., second edition, November 2005. ISBN 1-930934-15-7.

[ISO03]       ISO/IEC JTC1/SC22/WG21 - The C++ Standards Committee. Library Technical Report. `http://www.open-std.org/jtc1/sc22/wg21/docs/library_technical_report.html`, 2003. [Online; accessed 31-January-2008].

[ITK]         ITK. Insight Segmentation and Registration Toolkit. `http://www.itk.org`. [Online; accessed 31-January-2008].

[JBBS02]      Mark Jenkinson, Peter Bannister, Michael Brady, and Stephen Smith. Improved optimization for the robust and accurate linear registration and motion correction of brain images. *Neuroimage*, 17(2):825–841, October 2002.

[JT04]        Simon Jackson and Richard Thomas. *Cross-Sectional Imaging Made Easy*. Churchill Livingstone, August 2004. ISBN 044307187X / 9780443071874.

[Kit05]       Kitware, Inc. ITK Registration Methods. `http://www.itk.org/CourseWare/Training/SPIE2005-RegistrationMethodsOverview.ppt`, 2005. [Online; accessed 31-January-2008].

[KM04]        Arie Kaufman and Klaus Mueller. *The Visualization Handbook*, chapter Overview of Volume Rendering, pages 127–174. Academic Press, December 2004. ISBN 012387582X.

[LC87]        William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. In *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 163–169, New York, NY, USA, August 1987. ACM. ISBN 0897912276.

[LCLT74]      R. S. Lcdley, G. Di Chiro, A. J. Luessenhop, and H. L. Twigg. Computerized Transaxial X-ray Tomography of the Human Body. *Science*, 186 (no. 4160):207–212, October 1974.

[LLS$^+$00]   Guy Lefranc, T. Licht, H.J. Schultz, R. Beinert, and G. Mitic. Reliability

testing of high-power multi-chip IGBT modules. *Microelectronics Reliability*, 40:1659–1663, August 2000.

[MHV+01] David Mattes, David R. Haynor, Hubert Vesselle, Thomas K. Lewellyn, and William Eubank. Nonrigid multimodality image registration. In M. Sonka and K. M. Hanson, editors, *Proc. SPIE Vol. 4322, p. 1609-1620, Medical Imaging 2001: Image Processing, Milan Sonka; Kenneth M. Hanson; Eds.*, volume 4322, pages 1609–1620, July 2001.

[Mica] Microsoft Corporation. How to optimize memory usage in Exchange Server 2003. `http://support.microsoft.com/kb/Q815372`. [Online; accessed 31-January-2008].

[Micb] Microsoft Corporation. How to Set the /3GB Startup Switch in Windows. `http://technet.microsoft.com/en-us/library/bb124810.aspx`. [Online; accessed 31-January-2008].

[Micc] Microsoft Corporation. Memory Support and Windows Operating Systems. `http://www.microsoft.com/whdc/system/platform/server/PAE/PAEmem.mspx`. [Online; accessed 31-January-2008].

[Mor] Andreas Mortensen. Metal Matrix Composites in Industry: an Overview. `http://mmc-assess.tuwien.ac.at/public/mmc_in_ind.pdf`. [Online; accessed 31-January-2008].

[OIK+07] Moriyoshi Ohara, Hiroshi Inoue, Hideaki Komatsu, Hangu Yeo, Frank Savino, Vadim Sheinin, Giridharan Iyengar, Leiguang Gong, Shahrokh Daijavad, and Bradley Erickson. Real-Time Mutual-Information-Based Linear Registration on the Cell Broadband Engine Processor. Technical report, IBM Systems and Technology Group, 2007.

[PMea00] Paul Merle et al. X-Ray Computed Tomography on Metal Matrix Composites. MMC-Assess Consortium, Volume 4. Vienna University of Technology, Institute of Materials Science and Technology. `http://mmc-assess.tuwien.ac.at/public/v4_tomography.pdf`, August 2000. [Online; accessed 15-March-2008].

[QT a] Obtaining GPL'ed Qt for Windows. `http://www.paraview.org/Wiki/Obtaining_GPL'ed_Qt_for_Windows`. [Online; accessed 14-March-2008].

[QT b] Qt4 with Visual Studio. `http://qtnode.net/wiki?title=Qt4_with_Visual_Studio`. [Online; accessed 14-March-2008].

[Ras07]     Wayne Rasband. ImageJ. `http://rsb.info.nih.gov/ij/`, 1997-2007.
            [Online; accessed 31-January-2008].

[RPMF05]    Jörg Reimann, R. A. Pieritz, M. Michiel, and C. Ferrero. Topology
            of compressed pebble beds determined by X-ray tomography. Technical
            report, Wissenschaftliche Berichte FZKA, Forschungszentrum Karlsruhe,
            Karlsruhe, Germany, 2005.

[SRD06]     Michael Schöbel, Guillermo Requena, and Hanss-Peter Degischer. Voids
            in MMCs and their effects on thermal expansion. Extremat IP, Subpro-
            ject 2: High Temperature Heat Sinks, Work Package 2.3., Progress report
            (phase 3/project month 18), Vienna University of Technology, Institute of
            Materials Science and Technology, June 2006.

[SRD+07a]   Michael Schöbel, Guillermo Requena, Hans-Peter Degischer, Thomas Bus-
            laps, and Marco di Michiel. Changes of residual stresses and micro voids
            during thermal cycling of AlSiC MMCs. Technical report, Vienna Univer-
            sity of Technology, Institute of Materials Science and Technology, 2007.

[SRD+07b]   Michael Schöbel, Guillermo Requena, Hans-Peter Degischer, Thomas Bus-
            laps, and Marco di Michiel. Residual stresses and void kinetics in AlSiC
            MMCs under thermal load. Technical report, Vienna University of Tech-
            nology, Institute of Materials Science and Technology, 2007.

[Sys08]     Systems in Motion AS. SIMVoleon Documentation 2.0.2a. `http://doc.`
            `coin3d.org/SIMVoleon/`, 2008. [Online; accessed 15-March-2008].

[Tro]       Trolltech. Qt: Cross-Platform Rich Client Development Framework. `http:`
            `//trolltech.org/products/qt`. [Online; accessed 31-January-2008].

[Vol06]     Volume Graphics GmbH. VGL 3.2. `http://www.volumegraphics.com/`
            `products/vgl/features.html`, 2006. [Online; accessed 15-March-2008].

[VTK]       VTK. The Visualization ToolKit. `http://www.vtk.org`. [Online; accessed
            31-January-2008].

[VWMW97]    Paul Viola and III William M. Wells. Alignment by maximization of mutual
            information. *International Journal of Computer Vision*, 24(2):137–154,
            1997.

[Wik]       VTK Wiki. How to handle large data sets in VTK. `http://www.vtk.`
            `org/Wiki/VTK_FAQ#How_to_handle_large_data_sets_in_VTK`. [Online;
            accessed 31-January-2008].

[Wik07]     Wikipedia.   AIR (program) — Wikipedia, The Free Encyclope-
            dia. `http://en.wikipedia.org/w/index.php?title=AIR_%28program%`
            `29&oldid=136597701`, 2007. [Online; accessed 31-January-2008].

[WMLK89]    J.W. Wallis, T.R. Miller, C.A. Lerner, and E.C. Kleerup.    Three-
            dimensional display in nuclear medicine. *IEEE Transactions on Medical
            Imaging*, 8:297–230, December 1989.

[WVW⁺05]    Ivo Wolf, Marcus Vetter, Ingmar Wegner, Thomas Böttger, Marco Nolden,
            Max Schöbinger, Mark Hastenteufel, Tobias Kunert, and Hans-Peter
            Meinzer. The Medical Imaging Interaction Toolkit. *Medical Image Analy-
            sis*, 9(6):594–604, December 2005.

[ZF03]      Barbara Zitova and Jan Flusser.  Image registration methods: a survey.
            *Image and Vision Computing*, 21(11):977–1000, October 2003.