# Adaptive 3D algorithm to detect bridging ligaments during intergranular stress corrosion cracking of stainless steel

M. Janaszewski[1,3], TJ. Marrow[2] and L. Babout[1]

[1] Department of Computer Engineering, Technical University of Lodz, Poland, lbabout@kis.p.lodz.pl
[2] Materials Performance Centre, School of Materials, University of Manchester, UK
[3] The College of Computer Science in Lodz, Poland

## ABSTRACT

*This paper presents an adaptive algorithm to identify holes in 3D objects which cannot be labelled with standard techniques such as 6 or 26-connectivity based methods. The algorithm consists of two stages. The first one lies in the application of a voxel coding technique to make a transformation of an object into a set of clusters, where a cluster is a set of connected voxels. The set of clusters gives important information about the object topology. In the second step the algorithm extracts and analyses clusters to detect holes. The algorithm was satisfyingly tested on several 3D crack images from a stress corrosion cracking experiment on a stainless steel sample. In this case, the "holes" in the crack are bridging ligaments of solid material which connect the opposing faces of the crack. The discussion of obtained results and propositions for future works are also included in the paper.*

**Keywords** X-ray microtomography, image analysis, voxel coding, hole detection

## 1 INTRODUCTION

Prediction of fracture phenomena is an important issue in materials and mechanical science. For example, the lifetime to crack nucleation in safety critical equipment in the aerospace and power generation industries can be dominated by the early stages of crack development, as the crack growth rate is slowest in this stage. An important example with relevance to power generation is stress corrosion cracking of stainless steel (SCOTT, 2000). X-ray microtomographic studies have observed the initiation and propagation of such cracks along the grain boundaries (BABOUT, 2006). These observations reveal local ligaments of material which span the developed crack. These ligaments are called bridges, and generally correspond to special grain boundaries with higher resistance to stress corrosion. The mechanical requirement for ductile failure of these bridges as the crack extends and widens is predicted to have a direct effect on the rate to stress corrosion crack nucleation and the component lifetime (MARROW, 2006). Measurement of the spatial distribution and size of these bridges during in-situ observations of cracking is essential for the verification of 3D models (JIVKOV, 2006) for this mode of intergranular stress corrosion cracking, and its sensitvity to Grain Boundary Engineering (GBE) to optimise resistance (MARROW, 2006)

For an image processing point of view, the quantification of the bridges is made difficult by the fact that the present case can be schematized by two continuous phases -the crack and the material- crossing each other. Therefore, the bridges cannot be segmented easily by standard tools like binary segmentation or region growth (RUSS, 1995).

A significant number of image processing algorithms have been conceived and developed to simplify the complexity of 2D objects and detect features without affecting their topology. The most popular algorithms are based on skeletonization using thinning or distance transform methods (NIBLACK, 1992; RUSS, 1995). In the past 20 years, skeletonization methods have started to generalise to volumetric objects (ZHOU, 1999; PALÁGYI, 2002) in response to the increased popularity of 3D images, not only from medical applications, but also from materials or palaeontology science (BARUCHEL, 2000; TAFFOREAU, 2006). However, the extension to the 3D skeletonization has faced problems linked with time-consuming processes, lack of connectivity preservation, and also extraction of a skeleton which is as complicated as the original object (TRAN, 2005).

In this study, the main goal has been the detection of the bridge present along the crack path rather than the extraction of the crack skeleton. To do so, the algorithm initially proposed by ZHOU and TOGA (ZHOU, 1999), based on voxel coding, has been reformulated to isolate the bridges from the material and the crack. This paper presents in detail our approach and its application to bridge detection in an intergranular stress corrosion crack in stainless steel.

## 2 VOLUMETRIC OBJECT VISUALISATION AND PROCESSING

### 2.1 3D images of the stress corrosion cracks

Figure 1 shows a typical series of longitudinal images of a stress corrosion crack inside austenitic stainless steel. The images were obtained with the X-ray microtomography system at ID19 beam line, ESRF (Grenoble, France). The spatial resolution was 0.7 µm and the energy of the X-ray beam set to 30 keV. The white fringes surrounding the crack are generated by the difference in optical indices between the crack (air) and the material (iron). This effect is commonly named phase contrast (CLOETENS, 1997). More details about the set-up of the experiment can be found elsewhere (BABOUT, 2006).The crack, which is continuous from the edge of the sample to the interior of the sample in Figure 1a, is discontinuous in the plane of Figure 1b, then continuous again in Figure 1c. These discontinuities corresponds to bridging ligaments (denoted "BL"). A bridging ligament can be defined as a small section of material which locally has higher resistance to the propagation of a crack. This small portion acts as a bond or ligament between two surfaces of the crack. (BABOUT, 2006; MARROW, 2006).

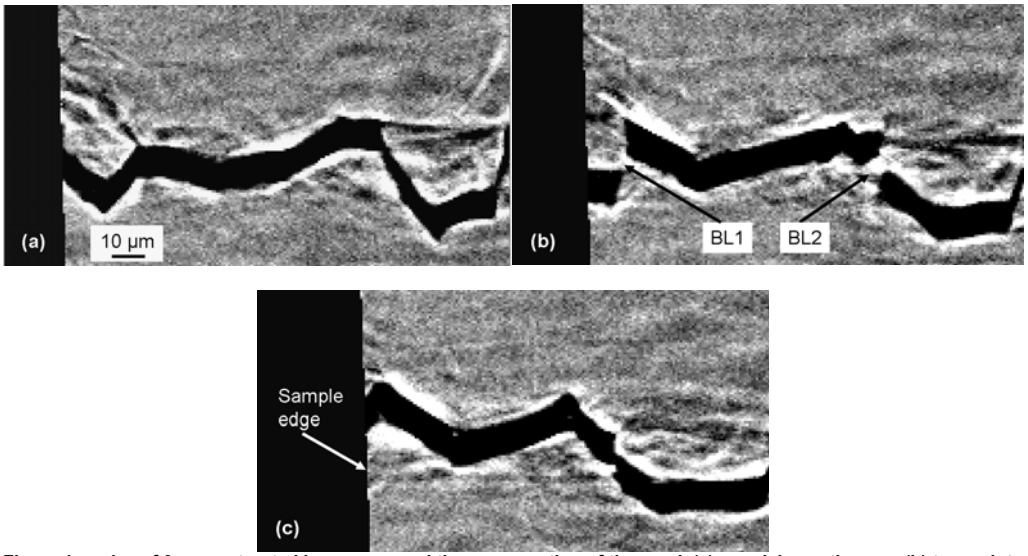



**Figure 1: series of 3 reconstructed images around the same portion of the crack (a): crack is continuous, (b) two points of discontinuity BL1 and BL2, (c) crack is once again continuous. The distance between 2 successive images is 7 µm.**

3D renderings of the portion of the crack shown in Figure 1 are displayed in Figure 2. The isosurface reveals satisfactorily the topography of the grain boundaries followed by the crack, i.e. flat faces and sharp edges. One can also see clearly bridge BL2, also shown with an enlarged view on Figure 2b, and also another bridge BL3 located along another grain boundary. Bridge BL1 is hidden but inspection of the volume shows that it is located at the edge of the sample. One can also notice, on the bottom right hand side of Figure 2a, portions of the crack where it is divided into two branches which do not subsequently merge. Hence, they do not surround a bridge ligament.

Quantification of the bridges located along the crack path can be done manually using both 2D images and 3D rendering of the crack, as shown above, but such a strategy is time consuming and unreliable. Image quantification via image processing is an alternative, but computational detection is very challenging in the particular issue of this work. The main difficulty resides in the bridges themselves. Topological connectivity exists between the bridge and two opposite sides of the material and classical methods based on voxel labelling or region rowing are therefore inappropriate. A particular strategy to select the voxels of the bridges is then necessary. Detection of bridges based on a voxel coding algorithms has been used because the technique, as it is described below, enables detection of locations where cracks divide and merge, hence allowing the identification of bridges.

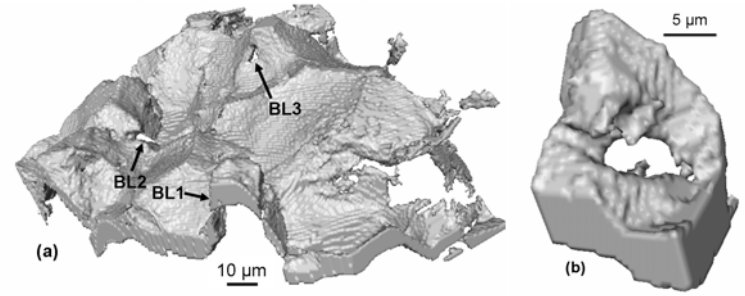



**Figure 2: (a) 3D isosurface of the crack showing positions of bridges BL1 and BL2 defined in Figure 1. (b) Enlarged view of bridge BL2.**

### 2.2 Basic concepts of 3D volumetric images

Basic concepts of a volumetric image and volumetric image processing were presented in detail elsewhere e.g. (ZHOU, 1999). In this subsection only the notions necessary to understand the following parts of the paper are introduced.

The 3D crack images described above are 3D binary volume data sets. A 3D binary volume set consists of voxels – the smallest unit cube in the volume. Each voxel is described by a quadruple ($x$, $y$, $z$, $v$) where ($x$, $y$, $z$) represents 3D location of the voxel and value $v$ indicates its membership. $v = 0$ means that a voxel belongs to a background, $v = 1$ indicating that a voxel belongs to an object (in our case a crack). Treating a voxel as a unit cube results in three kinds of voxel neighbourhood. Following the same notations in (ZHOU, 1999), for a voxel $p$ a voxel $q$ is called a *F-neighbour, E-neighbour*, or *V-neighbour*, of $p$ if it shares a face, an edge or a vertex, respectively, with voxel p. Voxels $p$ and $q$ are also called *F-connected*, *E-connected*, *V-connected*. Two voxels are *adjacent* or *neighbours* if they are at least V-connected. A voxel within an object is considered as an *inside voxel,* a *boundary voxel* if it does not have a background voxel ($v = 0$) as a neighbour. A voxel is called an *outside voxel* if all its neighbours are background voxels; otherwise it is regarded as a boundary voxel. A sequence of voxels $p_1, p_2,...,p_n$ is called a *voxel path* if it fulfils the following condition: $p_i$ is adjacent to $p_j$ if and only if $|i - j| = 1$, for $i, j = 1, 2...,n$ and $i \neq j$.

A set of voxels is *connected* if, for any two voxels within it, there is a path within the set connecting them. Two sets of voxels $A$, $B$ are *connected* or *adjacent* if there are at least two voxels, one within set $A$ and the other one within set $B$, which are neighbours.

### 2.3 Concept of a bridge (hole)

In this subsection the concept of a bridge is defined more precisely than was done in section 2. The concept of a bridge introduced in the previous section is equivalent to the well-defined concept of a 3D hole. Contrary to the 2D case, a 3D hole is not simple to define. A 2D hole can be defined as a set of background pixels bounded by a set of object pixels. Unfortunately a 3D hole is not a subset of 3D space. A definition of detection of a 3D hole can be found in literature (KONG, 1989). According to the definition the presence of a hole in an object $O$ is detected whenever there is a closed path of voxels in $O$ that cannot be deformed into a single voxel. For example a hollow ball has no hole, a solid torus has one. For a more formal definition of a hole, based on digital geometry, the interested reader is referred to (KONG, 1989). Unfortunately, the definition is not sufficient for 3D hole detection. It can be only used to check if an object has one or more holes (i.e. an object has no hole if any closed path within it can be deformed to a single voxel). Moreover, according to our knowledge, there is not any paper in the literature which describes an algorithm for 3D hole localization.

### 2.4 Voxel coding technique

According to (ZHOU, 1999) voxel-coding is a recursive voxel-by-voxel propagation and assignment of values to voxels within a volumetric object $O$ starting with the set $S$ of seed voxels ($S \subset O$), using a specific coding scheme until constraint conditions are met. The purpose of voxel-coding technique is to extract geometric features of objects and it was extensively exploited earlier e.g. (BORGEFORS, 1986; NIBLACK, 1992; ZHOU, 1999).

The voxel propagation scheme used in voxel-coding is similar to a discrete minimum distance transform. It uses the coding scheme "$n_f$-$n_e$-$n_v$" which is described with three integer values greater than 0: $n_f$, $n_e$, $n_v$, ($n_f < n_e < n_v$). First, all the voxels in $O$ are initialized with a code (value) of infinity. Than the propagation starts from seed voxels which are given code 0. Then all the F-neighbours, E-neighbours, V-neighbours within an object are given a code of $n_f$, $n_e$, $n_v$, respectively. In the $i^{th}$

iteration, all neighbours of voxels which have been assigned with a code value during the $i^{th}$ -1 iteration are processed. Assume that voxel *p* is assigned with a value of *n* for the $i^{th}$-1 iteration. Thus for the $i^{th}$ iteration all its F-neighbours, E-neighbours, V-neighbours within an object are assigned with value $n + n_f$, $n + n_e$, $n + n_v$, respectively, provided that the new code values are lower than the actual ones (i.e. an infinity value replaced by a code 2, or a code 4 replaced by a code 2). This method prevents voxels coded during an iteration from being coded again in the following one. This coding procedure stops when there is not any voxel to process in the next iteration or the constraint conditions are fulfilled (e.g. a particular voxel is met).
The choice of a particular coding scheme depends on the application. Commonly used schemes are "1-2-3", "2-3-4", "3-4-5". The choice of a set of seed voxels depends on the object feature to be extracted. Commonly used sets are: a set of boundary voxels which leads to voxel-coding called *boundary-seeded coding* (*BS-coding*) and the generated field is called *BS-field*, and a set of one voxel which leads to the coding procedure called *SS-coding,* the corresponding field is called *SS-field*. The SS-coding transforms an object into a collection of clusters. A *cluster* is defined as a set of connected object's voxels of the same SS-code. In addition we can consider some cluster types of great importance. An *LMcluster* is a cluster of the largest code of all adjacent ones (neighbours). A *merging cluster m* is a cluster which has more than one neighbour with same code value of one less than *m's*. The *m's* neighbours of code one less are called *predecessors* of *m*. A *dividing cluster d* is a cluster which has more than one neighbour with the same code value of one larger than *d's*. The *d's* neighbours of code one larger are called successors of *d*. A *medial voxel* of a cluster is defined as a voxel which belongs to the cluster and has the largest BS-code of all the cluster voxels.
The first step of our bridge detection algorithm consists in SS-coding with a specific voxel as seed voxel and the "1-2-3" metric as the coding scheme. The seed voxel is called a reference voxel RV. The search for RV can be done automatically or manually. In our approach an automatic selection has been used. It consists in application of SS-coding by defining an arbitrary voxel of the crack object as a seed. Then, the voxel of maximum SS-code is chosen as the RV. For an object which consists of several disconnected parts, an individual RV is necessary for each part.

## 3 Concept of bridge detection

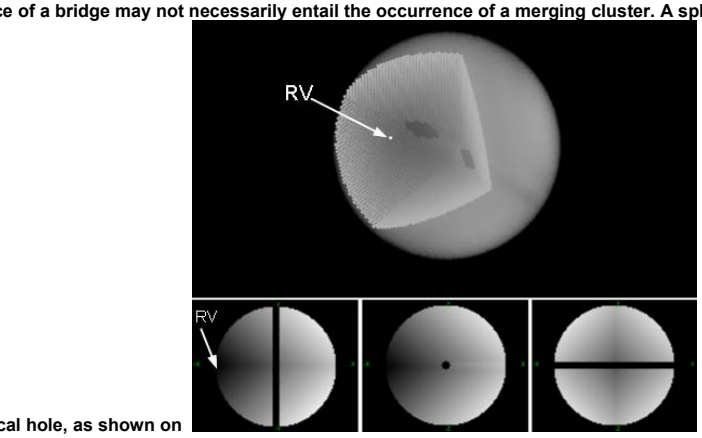
### 3.1 Bridge detection taking advantage of merging cluster

The SS-field and the notion of a merging cluster have been used in our approach to detect bridges. Indeed a merging cluster *m* determines the occurrence of at least two paths of adjacent clusters: $p_0$, $p_1,..,p_n$ and $q_0$, $q_1,..,q_n$ which end in one common dividing cluster *d*. The clusters $p_0$ and $q_0$ are predecessors of *m* and $p_n$, $q_n$ are successors of *d*. The proof of this statement is simple and is schematised in Figure 3. If it is assumed that the paths do not end in one dividing cluster, then three cases can be considered.

- First case: RV belongs to one of the clusters from $p_0$, $p_1,...,p_n$ path (see Figure 3a).. Therefore, $q_0$ has code value greater than *m's* which means that *m* is not a merging cluster.
- Second case: RV belongs to one of the clusters from $q_0$, $q_1,...,q_n$ path. (see Figure 3b). Therefore, $p_0$ has a code value greater than *m's* which is also in disagreement with the fact that m is a merging cluster.
- Third case: RP does not belong to any cluster of $q_0$, $q_1,...,q_n$, $p_0$ ,$p_1,...,p_n$ then both $q_0$ and $p_0$ have the same code value one greater than *m's*, like in the two previous cases (see Figure 3c).

To sum up, the supposition that the two paths of clusters do not end in one dividing cluster contradicts the initial supposition that *m* is a merging cluster. Therefore the cluster paths end in one dividing cluster *d*. Furthermore, the unbounded background voxels are localised between the two paths and a merging cluster which means the occurrence of a bridge.

**Unfortunately the presence of a bridge may not necessarily entail the occurrence of a merging cluster. A sphere containing a cylindrical hole, as shown on Figure 4, is a simple proving-example. The sphere with a small cylindrical bridge inside (which is visible in the three projection views), is displayed with semiopaque voxels of a dark grey colour. The grey colour of a voxel in the sphere represents the SS-code of the voxel (the higher the SS-code the brighter the voxel). The RV of the code is marked on the figure. One can also see an example cluster of code 60 presented with light grey coloured voxels. It can be noticed that the small bridge creates two holes in the cluster but it does not divide it into two clusters. It also occurred that there is no merging cluster in the sphere. This situation occurs because the size of the clusters formed in the sphere is far larger than the size of the bridge. Thus the bridge in the crack only cuts out holes in the clusters but is too thin to divide them into two parts. However, such situations will not occur in the present case of this study (or eventually will occur very rarely) because analysed cracks are not very thick compared with bridges size. Indeed, the clusters present in most cases a ribbon-like shape with a length to height ratio larger than 5:1, in contrast to the cluster presented in**



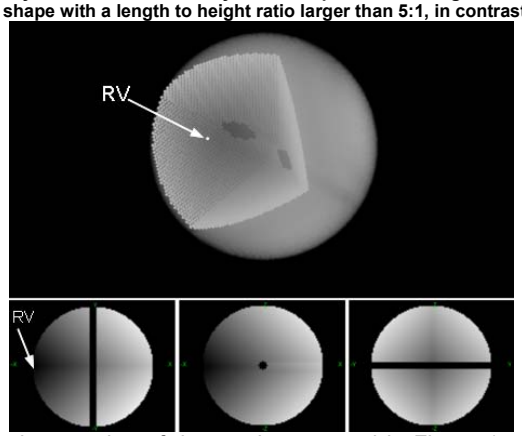




Figure 4. Typical clusters in a portion of the crack presented in Figure 1 are presented in Figure 5 where clusters c1 and c2 of code 45 and clusters c3, c4, c5 of code 72 are emphasised. The figure also highlights a very small bridge BL which divides one cluster into c3 and c4.

The next important problem to consider is the calculation of the number of bridges in a crack. Two new notions are therefore introduced. A *simple merging cluster SMC* is a merging cluster which has exactly two predecessors of the same code one less than the merging cluster's code. A *complicated merging cluster* CMC, which is described in the next subsection, is a cluster which has more than two predecessors of the same code one less than the cluster's code. A *SMC c* identifies exactly one bridge. Indeed, each *c's* predecessors $c_1$, $c_2$ starts a cluster sequence $cs_1$, $cs_2$ respectively, *which* end in the same dividing cluster and there is not any dividing and merging cluster inside $cs_1$, $cs_2$. Thus c, $cs_1$, $cs_2$ create a closed loop of clusters which localise a bridge. Moreover there is no other closed loop of clusters which identifies the same bridge because the two loops would be nested, which is impossible if the method of cluster extraction presented in subsection 2.4 is taken into consideration.

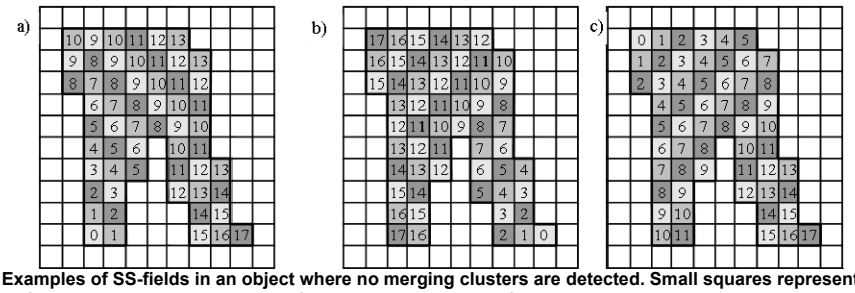



**Figure 3: Examples of SS-fields in an object where no merging clusters are detected. Small squares represent pixels of an image. Grey squares represent pixels of an object. The border of the object is marked with a thick black curve. A value inside a square represents the SS-code of the pixel. Three different gray levels in each image are used to emphasize clusters a) RV situated at the bottom of the left branch (pixel of code 0). Two dividing clusters: codes 9 and 10 b) RV positioned at the bottom of the right branch. Dividing cluster of code 12. c) RV localised outside of the branches (top left hand side corner of the object). Dividing cluster of code 8.**

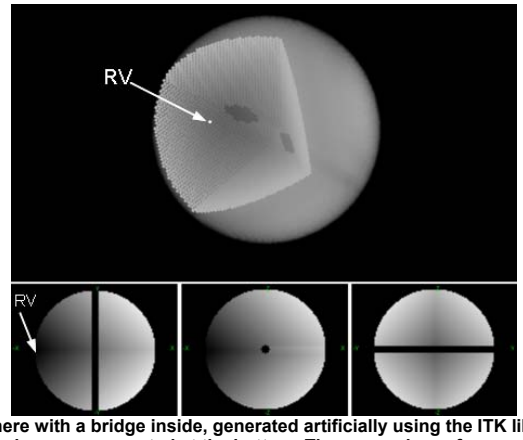



**Figure 4: 3D image of a sphere with a bridge inside, generated artificially using the ITK library (IBANEZ, 2005). Three different projections of the sphere are presented at the bottom. The grey colour of a voxel in the sphere represents its SS-code (the higher SS-code the brighter the voxel). The RV of the code (voxel of code 0) is marked on the figure.**

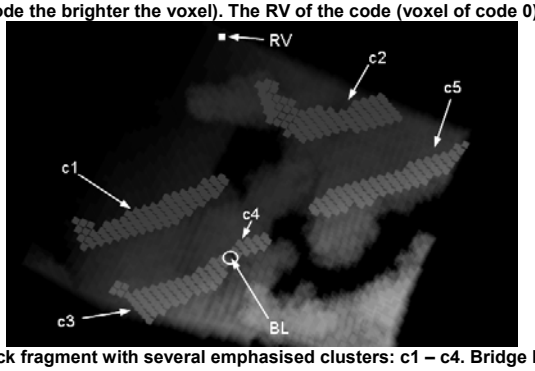



**Figure 5: 3D image of a crack fragment with several emphasised clusters: c1 – c4. Bridge BL which divides a cluster into two others noted c3 and c4 is also shown.**

### 3.2 Complicated merging cluster analysis

A *CMC* identifies more than one bridge. A calculation of the number of bridges identified by a *CMC* is very difficult in general. To simplify this task, we based our analysis on the fact that a crack is generally thin compared to its width and length. This allows us to introduce an order relation between predecessors of a *CMC*. A Typical CMC is shown in Figure 6. One can see a set of voxels, displayed in transparent mode, which correspond to a magnified view of the crack presented in Figure 1. A long CMC is denoted by *c* and covered by its predecessors. These are noted $cp_1$, $cp_2$, $cp_3$ and displayed with a dark grey colour. The predecessors are disconnected and identify 2 bridges. The first one is situated between $cp_1$ and $cp_2$ and the second one between $cp_2$ and $cp_3$. Similarly, when a CMC has *n* predecessors they identify *n* - 1 bridges, because the predecessors can be ordered in a list and only each pair of neighbours from the list can identify a bridge.

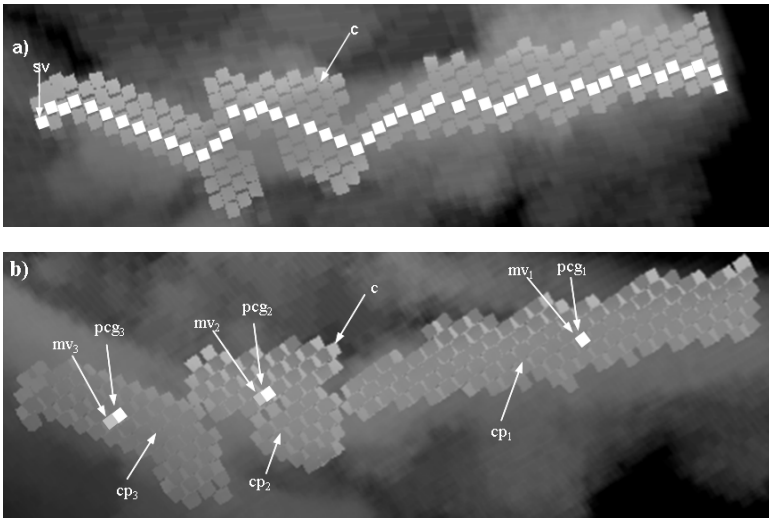



**Figure 6: A transparent view of semiopaque voxels which represents a magnified fragment of the crack isosurface from Figure 1 a) A complicated merging cluster *c* of code 147 displayed in light grey colour and its main skeletal path (white colour) starting from voxel sv. b) The c's predecessors are visualised with a dark grey colour and denoted $cp_3$, $cp_2$, $cp_1$. along with their centres of gravity $pcg_3$, $pcg_2$, $pcg_1$. Cluster *c* in the background is also displayed in light grey colour.**

The next step towards bridge detection is an algorithm which for a CMC, denoted by *c,* orders the c's *n* predecessors $cp_1$, $cp_2$,...,$cp_n$ in such a way that there is a bridge between each consecutive two predecessors of the list. The algorithm can be presented in several stages:

1. Extraction of the main skeletal path *mp* in *c*. Indeed we can treat *c* as an object and extract its main skeletal path. To do so, any skeltonization algorithm can be used. We use a part of the algorithm presented in (ZHOU, 1999).
2. Localisation of the nearest *mp's* voxel $mv_i$ to the centre of mass $cpg_i$ of clusters $cp_i$ for $i$ = 1,2,...,$n$.
3. Sorting the predecessors list according to the distance ($sc_i$) of $mv_i$ from the beginning of *mp*. The smaller $sc_i$ the smaller index of $cp_i$ in the sorted predecessor list.

Finally the algorithm results in a new list of predecessors in which, for two consecutive predecessors $cp_i$, $cp_{i+1}$: $sc_i \leq sc_{i+1}$, there is not other $cp_k$ ($k$ = 1, 2 ... n; $k \neq i, j$) for which $sc_i < sc_k < sc_{i+1}$.
The direct application of the algorithm for a typical CSC is presented in Figure 6. Figure 6 a) presents the result of the first step of the algorithm presented above - *c's* main skeletal path denoted by *mp* and its starting voxel SV. Figure 6 b) presents the results of the second step – predecessor's centre of gravities $pcg_1$, $pcg_2$, $pcg_3$ and corresponding nearest *mp's* voxels $mv_1$, $mv_2$, $mv_3$. Finally the sorted list of predecessors: $cp_3$, $cp_2$, $cp_1$ is obtained which indicates the presence of two bridges: one between $cp_3$ and $cp_2$ and another between $cp_2$ and $cp_1$.

## 4 RESULTS AND DISCUSSION

The algorithm of bridge detection have been tested on several 3D images of cracks. In each case, the algorithm performed the following operations: SS-coding, cluster extraction, merging cluster detection and finally SMCs and CMCs processing. The algorithm of predecessor ordering has been performed for each CMC as well. The results have been judged manually by visualisation.
Inspection of the data where merging clusters are superimposed to the initial crack showed that they indicate bridges very effectively. During this inspection, we did not find any bridge manually which was not detected by the algorithm. Moreover we have not noticed any CMC for which the algorithm of predecessors ordering corresponded to a wrong result. Furthermore, the merging clusters sometimes identify holes which are very difficult to localise manually, such as the one presented in Figure 6 and located between clusters cp1 and cp2. This hole is 1 voxel wide, which would be undetectable using normal visualisation. This feature is around 10 times smaller than the standard size of bridges found in

**Usunięto:** s

**Usunięto:** a merging cluster

**Usunięto:** s

stainless steel sample (0.7 μm compared to 5-10 μm) and therefore cannot be classified as a bridge. It is rather classified as noise.
Quantification of the number of bridges present in the portion of the crack corresponding to Figure 1 has been done. It shows that *SMCs* occur far more frequently than CMCs. Precisely, the algorithm found 80 SMCs and 2CMCs, one with 4 predecessors and the second with 3, which corresponds in the overall to 85 bridges. The volume containing the crack has a size of 220x200x120 voxels and it took around 2 minutes to process the algorithm presented in this paper on a standard PC equipped with a Pentium 4 microprocessor of 1.8 GHz.
Works are currently in progress to describe more precisely bridges, including parameters related to their location in the crack and their size and shape. This algorithm is based on the construction of a graph of clusters. Nodes in the graph are represented by merging and dividing clusters. Paths of consecutive clusters connect merging and dividing clusters and form edges of the graph. Such graph will be useful to localise bridges by loops of consecutive clusters. It will then be possible for each bridge to extract from coresponded loop border voxels which surround the bridge. This is a fundamental improvement to our algorithm to state if the bridges, or more generally the holes detected with the algorithm correspond to real features of the material's microsctructure, or should be classified as noise due to their small size.

Usunięto: a
Usunięto: this
Usunięto: s

## 5 CONCLUSIONS

Although cluster extraction is not a sufficient technique to detect bridges in any 3D image, it proved to be a very efficient method for bridges detection in 3D images of a crack, especially in the case of intergranular stress corrosion crack in stainless steel. The extracted merging clusters detect even very small entities which are very difficult to localise using visual inspection of the volume. (These may be noise in the tomography image (or where the crack opening displacement is of the order of the voxel size)
The next step of our study will be to localise bridges and calculate dimensional parameters of the closest path of voxels which surround bridges. The main drawback of merging clusters, form the bridge localisation point of view, is their size. Their ribbon-like shape weakens the accuracy of bridge localisation. A complementary approach is currently under development to obtain precise bridge localisation and extract dimensional parameters. This will allow to classify the holes as bridges or noise. This information is needed in order to model intergranular stress corrosion cracking.

## 6 ACKNOWLEDGMENTS

The authors are grateful to Dr. Cloetens (beamline scientist on ID19, ESRF), Prof. Buffiere (MATEIS, INSA Lyon, France), Dr. Engelberg and Mr. Johnson (School of Materials, University of Manchester, UK) for their assistance during the experiment.

## 7 REFERENCES

BABOUT L., MARROW T.J., ENGELBERG D., WITHERS P.J., (2006), X-Ray Microtomographic Observation of Intergranular Stress Corrosion Cracking in Sensitised Austenitic Stainless Steel, *Mater. Sci. Technol.,* 22 1068-1075

BARUCHEL J., BUFFIERE J.Y., MAIRE E., MERLE P., PEIX G., (2000), *X-Ray Tomography in Materials Science*, Hermes Science

BORGEFORS G., (1986), Distance Transforms on Digital Images, *Computer Vision Graphics and Image Processing,* 34 344-371

CLOETENS P., PATEYRON-SALOME M., BUFFIERE J.Y., PEIX G., BARUCHEL J., PEYRIN F., SCHLENKER M., (1997), Observation of Microstructure and Damage in Materials by Phase Sensitive Radiography and Tomography, *Journal of Applied Physics,* 81 5878-5886

IBANEZ L., SCHROEDER W., NG L., CATES J., (2005), *The Itk Software Guide*, Kitware, Inc.

JIVKOV A.P., STEVENS N.P.C., MARROW T.J., (2006), A Three-Dimensional Computational Model for Intergranular Cracking, *Computational Materials Science,* 38 442-453

KONG T.Y., (1989), A Digital Fundamental Group, *Computer Graphics,* 13 159-166

MARROW T.J., BABOUT L., JIVKOV A.P., WOOD P., ENGELBERG D., STEVENS N., WITHERS P.J., NEWMAN R.C., (2006), Three Dimensional Observations and Modelling of Intergranular Stress Corrosion Cracking in Austenitic Stainless Steel, *Journal of Nuclear Materials,* 352 62-74

NIBLACK C.W., GIBBONS P.B., CAPSON D.W., (1992), Generating Skeletons and Centerlines from the Distance Transform, *CVGIP: Graph. Models Image Process.,* 54 420-437

PALÁGYI K., (2002), A 3-Subiteration 3d Thinning Algorithm for Extracting Medial Surfaces, *Pattern Recognition Letters,* 23 663-675

RUSS G.C., (1995), *The Imaging Processing Handbook*, CRC Press

SCOTT P.M., (2000), Stress Corrosion Cracking in Pressurized Water Reactors-Interpretation, Modeling and Remedies, *Corrosion,* 56 771-782

TAFFOREAU P., BOISTEL R., BOLLER E., BRAVIN A., BRUNET M., CHAIMANEE Y., CLOETENS P., FEIST M., HOSZOWSKA J., JAEGER J.J., KAY R.F., LAZZARI V., MARIVAUX L., NEL A., NEMOZ C., THIBAULT X., VIGNAUD P., ZABLER S., (2006), Applications of X-Ray Synchrotron Microtomography for Non-Destructive 3d Studies of Paleontological Specimens, *Applied-Physics-A-Materials-Science-Processing,* A83 195-202

TRAN S., SHIH L. (2005). Efficient 3d Binary Image Skeletonization. Computational Systems Bioinformatics Conference Workshops.

ZHOU Y., TOGA A.W., (1999), Efficient Skeletonization of Volumetric Objects, *IEEE Transactions on visualization and computer graphics,* 5 196-209