

Lehrstuhl für Metallurgie, Department Metallurgie
Montanuniversität Leoben

Development of a Digital Image Processing Software to Evaluate the Solidification Structure in Steel



vorgelegt von Stefan Griesser
Leoben, Juni 2009

Acknowledgements

First of all I would like to thank **Prof. Dr. Christian Bernhard** for the opportunity to write this present master thesis at the chair of metallurgy. Even in the early stage of my academic studies at the Mining University of Leoben he gave me the chance to be a part of his establishment and even the great team around him.

My special thanks go to **Dr. Robert Pierer**, who was the supervising tutor and also a good friend since I was working at the Christian Doppler Laboratory for Metallurgical Fundamentals of Continuous Casting Processes. Thank you for the many enlightening discussions and comments on the work. Rock on Dude!

Prof. Dr. Paul O'Leary for his great inputs and the lots of time he offered to me. We had many hours working on the software and also some very interesting talks. Thank you very much!

Thanks to all my good friends here in Leoben, I'll never forget this time in Leoben. I really hope that we'll never lose our contact and see us all together - some day. See you guys!

To my family, without their backup this would never be possible.

Affidavit

I declare in lieu of oath, that I wrote this thesis and performed the associated research myself, using only literature cited in this volume.

Stefan Griesser
Leoben, Juni 2009

Contents

List of Figures	V
List of Tables	VI
1. Introduction	1
2. Development of the solidification structure	3
2.1. General theory of crystallization	3
2.2. Theories of solidification	6
2.3. Summary and Conclusions with respect to the continuous casting process	16
3. Concept behind the method	18
4. Image Processing	21
4.1. Image acquisition	23
4.2. Segmentation	23
4.2.1. Edge detection	23
4.2.2. Global/Constant thresholding	24
4.2.3. Adaptive thresholding	25
4.2.4. Regiongrowing	25
4.3. Region description	26
4.3.1. Contour-based descriptions	27
4.3.2. Region-based descriptions	27
4.4. Object recognition	29

5. Processing steps of the developed software	30
5.1. Binarization	31
5.2. Smoothing	33
5.3. Skeletonization / Thinning	34
5.4. Searching for skeleton intersection points	35
5.5. Nearest neighbours	36
5.6. Intensity Profiles	38
5.7. Unifying the angles	43
5.8. Illustration	44
5.9. Summary	45
6. Validation	46
6.1. Comparison with manual evaluation	47
6.2. Effect of unifying the angles	49
6.3. Influence of the criterion for the choice of the intensity profile	51
6.4. Influence of the image for intensity profiles	53
6.5. Processing time	53
6.6. Summary and Conclusions	54
7. Summary, Conclusion and Outlook	56
7.1. Summary and Conclusion	56
7.2. Outlook	57
A. Guidelines for use	63
A.1. Preliminary steps	63
A.1.1. Binaryzation	63
A.1.2. Smoothing	64
A.2. Main calculation	64

List of Figures

2-1. The free energy change associated with homogeneous nucleation of a sphere of radius r ¹⁹	4
2-2. The excess free energy ΔG^* of solid clusters for homogeneous and heterogeneous nucleation. Note r^* is independent of the nucleation site ¹⁹	5
2-3. Schematically illustration of a dendrite according to Kurz and Fisher ¹	6
2-4. Miller indices for the directions of a dendrite tip ³⁰	7
2-5. Branching of dendrites ³⁰	8
2-6. Solid/Liquid interface and temperature distribution ¹	8
2-7. Solidification morphology dependent on the amount of constitutional undercooling ¹	9
2-8. Schematic summary of solidification morphologies ¹	10
2-9. Structural zone formation in castings ¹	11
2-10. Schematic illustration of the Walton and Chalmers model for competitive grain growth ^{34,35} . Grain <i>A</i> and <i>B</i> are favourably and unfavourably orientated, respectively.	11
2-11.a) Predicted microstructures and b) predicted undercoolings for different dendrite orientations under flat isothermal conditions: 1) converging, 2) axial and 3) diverging ³⁸	13
2-12.a) Predicted microstructures and b) predicted undercoolings with heterogeneous nucleation for different dendrite orientations under transient thermal conditions: 1) converging, 2) axial and 3) diverging ³⁸	14
2-13. Superposition of various crystallizations in the cast condition for low carbon steels ³³	15

2-14. Grain selection during solidification and primary grains consisting of dendrites with uniform orientations according to Pokorny ³³	16
3-1. a) Austenite grains in a steel with 0.4 wt.%C (etched with nitric acid in alcohol) ⁴³ and b) Primary structure without visible grain boundaries in a steel with 0.7 wt.%C (etched with picric acid).	19
4-1. Hierarchy of the image processing operations according to Jähne ⁴⁴ . Rectangles correspond to the operations, ovals to the data structures.	22
4-2. Skeleton of a rectangle. Both points A and B are skeletal, but point C does not belong to the skeleton. The skeleton is marked by thick black line segments.	28
5-1. Original gray scale image of a 0.7 wt.%C steel etched with picric acid.	31
5-2. a) Schematic illustration of an intensity histogram with a threshold T and b) the result of a global threshold.	32
5-3. a) Binary image using a global threshold and b) using an adaptive threshold	32
5-4. a) Diamond and b) octagon structure elements ⁵²	33
5-5. a) Binary image using an adaptive threshold and b) after a thinning operation (SE = diamond).	33
5-6. Example of a thinning operation. a) Dendrite in a binary image and b) after the thinning operation.	34
5-7. a) Shape of a dendrite in a binary image, b) result of skeletonization, c) result of thinning.	35
5-8. a) Smoothed binary image and b) the appropriate thinned image.	35
5-9. Irregular shape of a dendrite leads to more than one intersection points.	36
5-10. Searching for the nearest neighbours for every intersection point in a defined radius R	37
5-11. M-Code for detecting the nearest neighbours.	38
5-12. Schematic illustration of intensity profiles on different arrangements of the dendrites. a, c) Lines between the actual intersection point and its neighbours and b, d) the plotted intensity profiles.	39
5-13. Calculation of the angle based on the pair of intersection points with the maximum intensity.	41

5-14. Sourced code for calculating the angle between two points with the highest mean intensity value I_{max}	42
5-15. Dividing the image in square tiles, extended by boundaries, in order to reduce the processing time.	42
5-16. Unifying a group of dendrites with a difference in their orientation smaller than a defined tolerance value ϕ_T	43
5-17. Source code for coloring different angles using the colormap "jet".	44
5-18. Different colormaps.	45
6-1. Original image of a steel sample with 0.6 wt.% C.	47
6-2. Result of a manual evaluation.	48
6-3. Result of the software.	48
6-4. Comparison of the result of the software with the manual evaluation.	49
6-5. Result of the pure calculation without the fitting operation.	50
6-6. Result of the calculation including the fitting operation.	51
6-7. Result of the calculation on the original image with the criterion "Ratio".	52
6-8. Result of the calculation on the binary with the criterion "Ratio".	52
6-9. Result of the calculation with the intensity profiles taken on the binary image. . .	53
6-10. Comparison of the computational costs for the calculation of the entire image and in tiles.	54
A-1. Graphical User Interface (GUI).	65
A-2. LOG-File from a calculation.	67

List of Tables

2-I. Direction of Dendritic Growth ³⁰	7
4-I. Structure elements for a thinning operation ⁴⁶	29
6-I. Reference values for the parameters in the <i>Results</i> -frame.	46
6-II. Chosen parameters for the main calculation.	50
6-III. Chosen parameters for the applied fitting operation.	51

1. Introduction

Continuous casting (CC) is nowadays the dominating technology in the transformation from liquid to solid steel. 90 % of the annual worldwide production of more than 1 billion tons is produced via the CC process. Thereby, liquid steel is cooled in a water cooled mould resulting in a heterogeneous nucleation and further solidification. Alloys such as steel show a columnar dendritic solidification, i.e. tree-like primary crystals (called dendrites), which are growing contrary to the heat flux during solidification¹. The most important parameters to characterize a dendrite are the primary (λ_1) and secondary dendrite arm spacing (λ_2). Nevertheless, considering the whole system also the primary grain size has an important influence for various phenomena like segregations, porosities and hot tears. Primary grains (PG) are defined as regions where all the dendrites have the same orientation. Primary grain boundaries (PGB) are areas where primary grains adjoin, which means the regions where a misorientation of neighbouring dendrites occurs.

Because of the relationship between the material properties and the microstructure the knowledge of the microstructural parameters, such as the primary grain size, is important for a better understanding of solidification and will lead to improved processes, which in turn lead to materials with fewer defects and a better quality. The relevant literature mostly deals with the structural parameters λ_1 and λ_2 and rarely with the PG size. As a big problem in the continuous casting of steel, hot tearing may appear during solidification in the mushy zone (range between the liquid temperature and the solidus temperature). This phenomenon preferentially occurs at PGB which are orthographic orientated to the stress direction. The question why hot tearing takes place at PGB can be answered by (1) the different coalescence behaviour of dendrites belonging to the same grain and dendrites at PGB^{2,3}, as well by (2) strain localization^{2,4,5}. The

latter case explains also the reason why a coarse grain structure is more prone to hot tearing than finer grains. In order to quantify the local strain at the PGB, the quantity of the PGB has to be determined. The examination of the primary grain size by hand is very time consuming and also a subjective process. The aim of the present study is to develop a digital image processing software in MATLAB[®] R2007b because existing machine vision based methods are just able to evaluate secondary or tertiary grain boundaries because of their visible shape but no primary grain boundaries. The software developed during this Master Thesis is, to the best of my knowledge, the first automatic procedure for the determination of primary grain boundaries from the orientation of the dendrites.

In the first part of this thesis a literature overview on the development of microstructure is presented. Secondly, the general procedures of classical image processing are introduced. Within the scope of the practical work of this thesis, the concept of the developed software is described and the software procedures are explained in detail. The validation of the developed software and a summary and conclusion are presented in the last part. In the appendix the guidelines for use are presented in order to give an detailed instruction manual for the usage of the software.

2. Development of the solidification structure

The following chapter provides a literature study about the development of the solidification structure, which is subdivided into three sections. The first section gives an overview about the general theory of crystallization. Considering a system of liquid steel, heterogeneous nucleation appears due to a heat removal caused by the contact to a cooled mold wall. Solid particles in the melt are forming to clusters and further to nuclei which can grow much faster, if there is an high grade of undercooling in the melt. After this, the theories about the further solidification are summerized. Within this section, the growth of the emerging solid/liquid interface is described, whereby different morphologies of this interface can appear dependent on the temperature gradient G , the growth rate V and the cooling rate \dot{T} . The phenomenon of competitive grain growth is illustrated and discussed on the basis of the investigations of three different authors. A summary and conclusion, where a definition of the so called primary grains and primary grain boundaries is given, completes this chapter.

2.1. General theory of crystallization

Based on the fundamental thermodynamic works of J. Gibbs the first general theory of crystallization was published by M. Volmer and A. Weber⁶. An advanced theory was published by J. I. Frenkel⁷, which has been confirmed by many other scientists like V. I. Danilov⁸, V. D. Kuznecov⁹, A. N. Kolmogorov¹⁰, I. L. Mirkin¹¹, J. V Grecnyi¹², B. Chalmers¹³, K. A. Jackson¹⁴, R. Kahn¹⁵, D. I. Ovsienko¹⁶, N. N. Sirota¹⁷ and W. Winegard¹⁸.

The generally accepted theory of crystallization, worked out by the named authors, contains the following basic fundamentals. A homogenous nucleation can appear in an undercooled melt. The activation energy for the homogeneous nucleation ΔG_{hom} is given by

$$\Delta G_{hom} = -\frac{4}{3}\pi r^3 \Delta G_V + 4\pi r^2 \gamma_{sl} \quad (2-1)$$

where r is the radius of the nucleus, ΔG_V is the volume free energy and γ_{sl} is the solid/liquid interfacial free energy. Since the interfacial term (on the right hand side) increases as r^2 whereas the volume free energy released (on the left hand side) only increases as r^3 , the creation of small particles of solid always leads to a free energy increase. Due to this increase the liquid phase is able to maintain in a metastable state almost indefinitely at temperatures below the melting point. It can be seen from figure **Figure 2-1** that for a given undercooling there is a certain radius, r^* , which is associated with a maximum excess free energy ΔG^* . If $r < r^*$ the system can lower its free energy by dissolution of the solid, whereas when $r > r^*$ the free energy of the system decreases if the solid grows. Since $r = r^*$ the critical nucleus is effectively in (unstable) equilibrium with the surrounding liquid¹⁹.

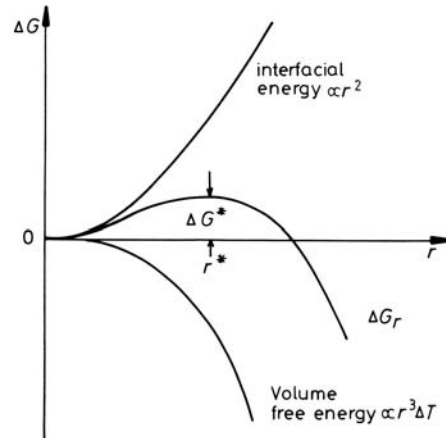


Figure 2-1.: The free energy change associated with homogeneous nucleation of a sphere of radius r ¹⁹.

The velocity of nucleation depends on the degree of undercooling, the surface tension between the melt and the nucleus and the mobility of the atoms during the transition of liquid to solid. By the existence of a small amount of soluble capillary active elements the velocity of nucleation can be increased. In most cases the nucleation happens at insoluble components or the mold wall, which is known as the heterogeneous nucleation.

The activation energy for the heterogeneous nucleation is given by

$$\Delta G_{het} = \left\{ -\frac{4}{3}\pi r^3 \Delta G_V + 4\pi r^2 \gamma_{sl} \right\} S(\theta) \quad (2-2)$$

where

$$S(\theta) = (2 + \cos\theta)(1 - \cos\theta)^2/4 \quad (2-3)$$

This expression is, except the factor $S(\theta)$, the same as that obtained for homogeneous nucleation, see Equation 2-1. $S(\theta)$ is a shape factor, dependent only on the wetting angle θ . Also in this case, the velocity of nucleation depends on the degree of undercooling, but the starting point of nucleation on the insoluble components begins at a smaller undercooling in comparison to the homogenous nucleation, as it can be seen in **Figure 2-2**.

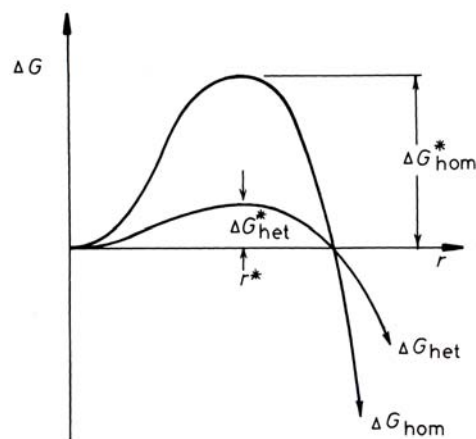


Figure 2-2.: The excess free energy ΔG^* of solid clusters for homogeneous and heterogeneous nucleation. Note r^* is independent of the nucleation site¹⁹.

In practical solidification processes the homogeneous nucleation is rarely encountered. Instead heterogeneous nucleation occurs at crevices in mold walls, or at impurity particles in the liquid. At these nuclei the further solidification proceeds. In the last century many theories about the solidification of castings have been developed, which are all based on the general theory of crystallization. Some of them (V. I. Danilov²⁰, G. P. Ivancov²¹, D. D. Saratovkin²², J. L. Walker²³, V. I. Dobatkin²⁴) are strictly related to the general theory of crystallization, others differ from it (V. O. Gagen-Torn²⁵, N. I. Chworinov²⁶, B. Chalmers¹³, D. K. Cernov²⁷), because

the general theory is apparently universally valid, but under real conditions it is just an approximation. This is because the practical solidification processes are often far away from steady state conditions.

2.2. Theories of solidification

Alloys such as steel show a columnar dendritic solidification, i.e. tree-like primary crystals (called dendrites), which are growing contrary to the heat flux during solidification¹. A schematic illustration of a dendrite is shown in **Figure 2-3**. Behind a short paraboloid tip region, which often constitutes less than 1 % of the length of the whole dendrite, perturbations appear on the initially smooth needle as in the case of breakdown of a planar interface. These perturbations grow and form branches in the four directions which are perpendicular to the trunk. If the primary spacing is sufficiently great, these cell-like secondary branches will develop into dendritic-type branches and lead to the formation of tertiary or higher-order arms. A dendritic form is usually characterized in terms of the primary (dendrite trunk) spacing, λ_1 , and the secondary dendrite arm spacing λ_2 . It is important to note that the value λ_1 measured in the solidified microstructure is the same as that existing during growth, whereas the secondary spacing is enormously increased due to the long contact time between the highly-curved, branched structure and the melt¹. Because of the fact, that the term microstructure is in the literature mostly related to the primary and secondary dendrite arm spacing and unlike to the primary grain size, the term solidification structure is used here instead of microstructure.

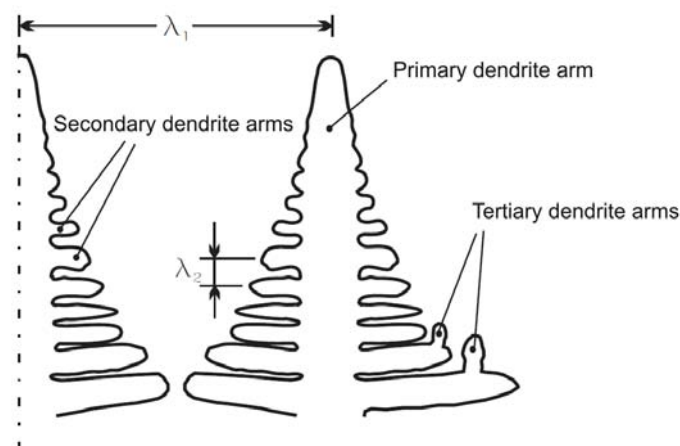


Figure 2-3.: Schematically illustration of a dendrite according to Kurz and Fisher¹.

It was pointed out first by Weinberg and Chalmers²⁸ that the arms of dendrites always grow in crystallographically determined directions, each of which is the axis of a pyramid whose sides are the most closely packed planes with a pyramid can be formed (this excludes the basal plane in the hexagonal structure)²⁹. The directions that conform to this description, and are in fact observed experimentally, are given in **Table 2-1**.

Table 2-1. Direction of Dendritic Growth³⁰.

Structure	Dendritic growth
Face-centered cubic	$\langle 100 \rangle$
Body-centered cubic	$\langle 100 \rangle$
Hexagonal close-packed	$\langle 1010 \rangle$
Body-centered tetragonal (tin)	$\langle 110 \rangle$

The general explanation for the crystallographic features of dendritic growth must be related to anisotropy of the relationship between growth rate and kinetic driving force. If there were no anisotropy, the dendrite should grow in a direction that is controlled entirely by the thermal conditions³⁰. In the cubic metals, which include steels, the direction of the columnar crystals (i.e. dendrites) is well defined and lies in the $\langle 100 \rangle$ direction, which is schematically illustrated in **Figure 2-4**.

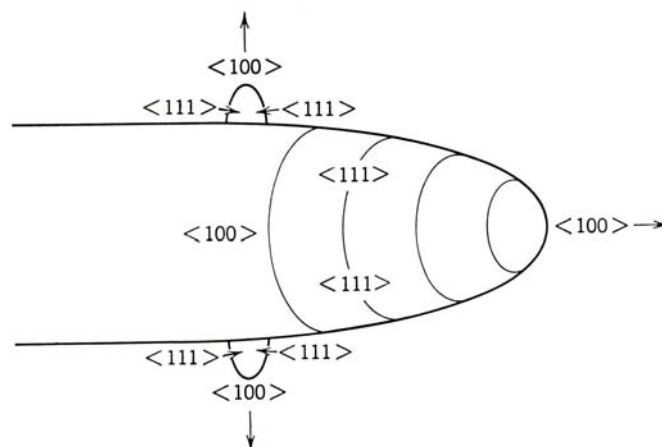


Figure 2-4. Miller indices for the directions of a dendrite tip³⁰.

The conclusions of Mullins and Sekerka³¹ on the dependence of the stability of a curved interface on its radius of curvature indicate that the tip of a dendrite should grow until it becomes

large enough to be unstable. After that, the tip breaks down into a number of separate tips, each of smaller radius. Each of these tips again grows until it becomes unstable. The proposed process is illustrated in **Figure 2-5**³⁰.

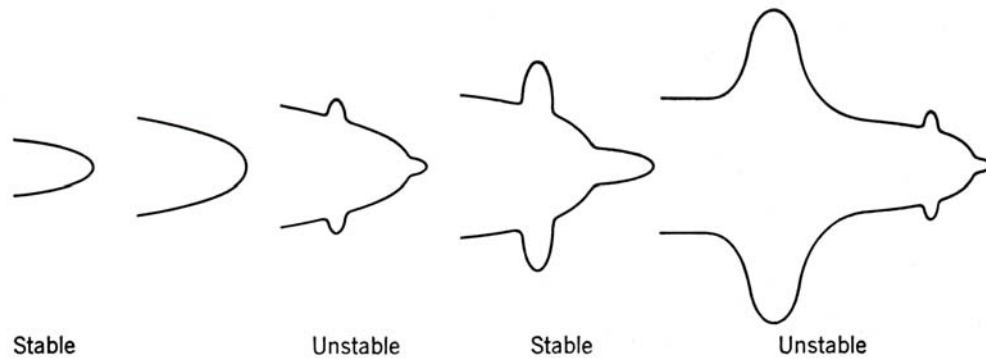


Figure 2-5.: Branching of dendrites³⁰.

In the case of a pure metal, like in **Figure 2-6 a, b**, which is solidifying inwards from the mold wall, the columnar grains (**Figure 2-6 a**) possess an essentially planar interface and grow in a direction which is antiparallel to that of the heat flow. Within the equiaxed region of pure cast metal (**Figure 2-6 b**), the crystals are dendritic and grow radially in the same direction as the heat flow.

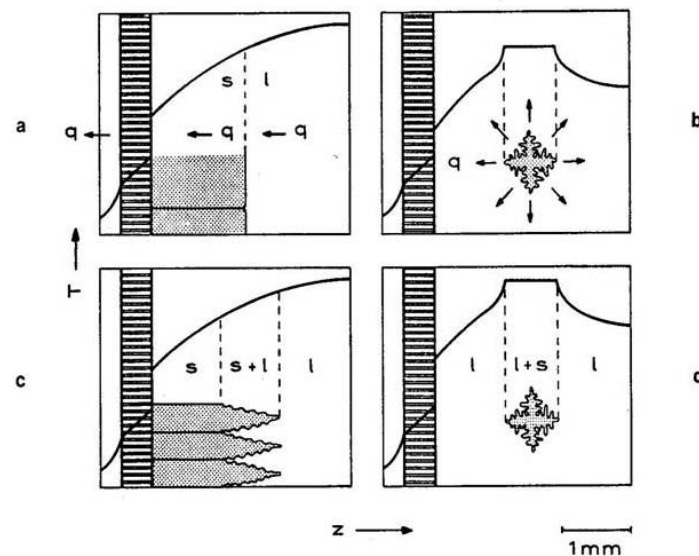


Figure 2-6.: Solid/Liquid interface and temperature distribution¹.

When alloying elements or impurities are present, the morphology of the columnar crystals (**Figure 2-6 c**) is generally dendritic. The equiaxed morphology in alloys (**Figure 2-6 d**) is

almost indistinguishable from that in pure metals, although a difference may exist in the relative scale of the dendrites. This is because the growth in pure metals is heat-flow-controlled, while the growth in alloys is mainly solute-diffusion-controlled. In columnar growth the hottest part of the system is the melt, while in equiaxed solidification the crystals are the hottest part. It follows that the melt must always be cooled to below the melting point (i.e. undercooled) before equiaxed crystals can grow¹. When the temperature gradient due to the heat flux is greater than the liquidus temperature gradient at the solid/liquid interface, the latter is stable (**Figure 2-7a**). On the other hand, it can be seen that a driving force for interface change will be present whenever the slope of the local melting point curve (liquidus temperature) at the interface is greater than the slope of the actual temperature distribution. This is easily understood since the undercooling encountered by the tip of a perturbation advancing into the melt increases and therefore a planar interface is unstable (**Figure 2-7b**). The temperature profile in **Figure 2-7b** is only hypothetical; after the dendritic microstructure shown in the lower figure has developed, the region of constitutional undercooling is largely eliminated. Only a much smaller undercooling remains at the tips of the dendrites¹.

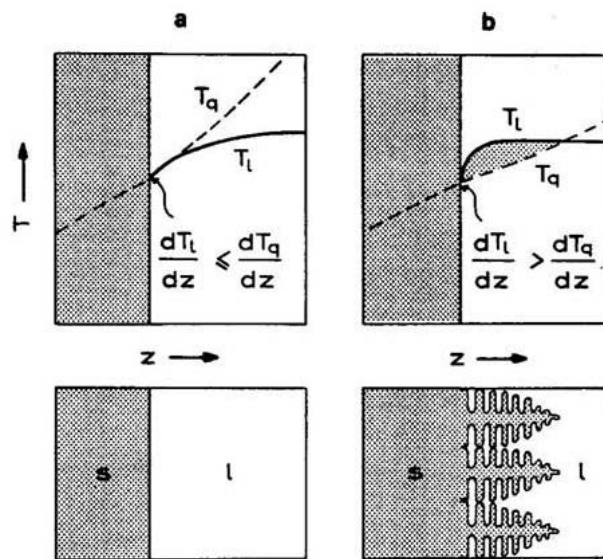


Figure 2-7.: Solidification morphology dependent on the amount of constitutional undercooling¹.

The solidification morphology is depending of the temperature gradient G , the growth rate V and the cooling rate \dot{T} (the product of $G \cdot V$), as it is summarized in **Figure 2-8**. Moving from the lower left to the upper right along the lines at 45° leads to a refinement of the structure for a given morphology. The ratio, G/V , largely determines the growth morphology. Passing from the

lower right to the upper left leads to changes in morphology (from planar, to cellular, to dendritic growth). Superimposed on the diagram are typical conditions for two solidification processes, casting and directional solidification (D.S.)¹.

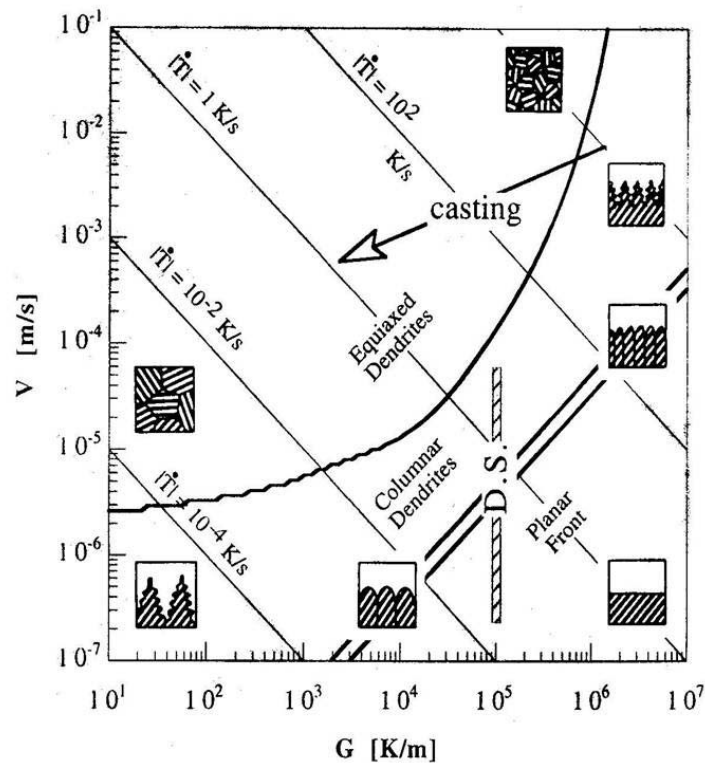


Figure 2-8.: Schematic summary of solidification morphologies¹.

The solidification structure in steel can be described as follows: A fine-grained zone at the surface (initial zone or chill zone) is followed by a zone of columnar grains (columnar zone), sometimes (depending on process conditions) with a zone of equiaxed grains in the centre (inner equiaxed zone), as it is shown in **Figure 2-9**. The fine-grained zone can be explained by the high undercooling resulted by the contact with the mold. The transition from columnar to equiaxed growth is highly dependent upon the degree of convection in the liquid. In order to promote this transition in the continuous casting process, electromagnetic stirring is often used which leads to superior soundness at the ingot centre.

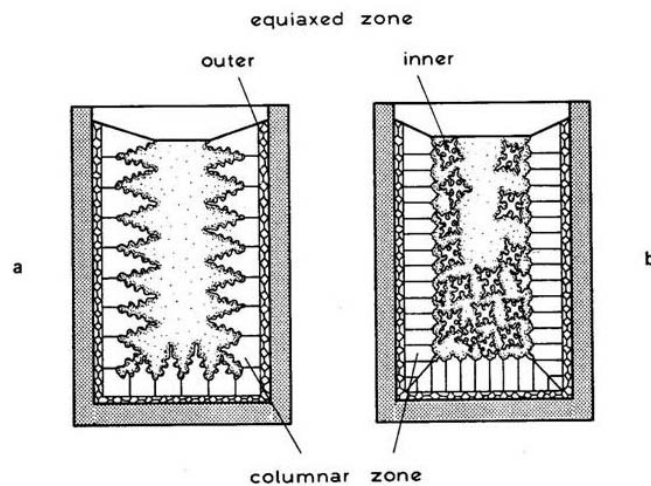


Figure 2-9.: Structural zone formation in castings¹.

As mentioned previously, the preferred growth direction of steel is $\langle 100 \rangle$ and primary as well as secondary arms grow $\langle 100 \rangle$ (see **Figure 2-4**), as far as dendritic growth is concerned. This leads to a phenomenon called the "competitive grain growth". The generally accepted model for competitive grain growth proposed by Walton and Chalmers³² is based on the difference in undercooling of favourably and unfavourably oriented dendrites with respect to the thermal gradient. Amongst the many small grains in the initial zone a certain number have this preferential $\langle 100 \rangle$ direction and these grow more rapidly than their neighbours so that they gradually eliminate the latter by overlapping them.

Rappaz et al.^{34,35} schematically summarized the Walton and Chalmers model (**Figure 2-10**).

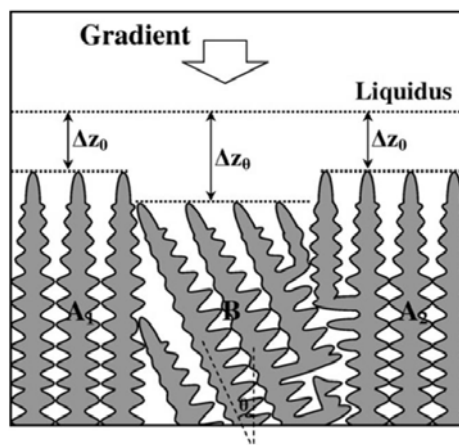


Figure 2-10.: Schematic illustration of the Walton and Chalmers model for competitive grain growth^{34,35}.

Grain A and B are favourably and unfavourably orientated, respectively.

Grains A_1 and A_2 are favourably orientated. Grain B is unfavourably orientated and the $\langle 100 \rangle$ direction has a misorientation, θ , with respect to the heat flow direction. To keep up with the more favourably orientated neighbors, grain B grows at a greater undercooling. In the case of diverging dendrites (grains B and A_2 in **Figure 2-10**), development of new dendrites from grain A_2 can lead to overgrowth of grain B and the grain boundary is thus inclined. In the case of converging dendrites (grains A_1 and B in **Figure 2-10**), the dendrite tips in grain B impinge upon the side of grain A_1 at the grain boundary and are stopped. Since grain A_1 does not develop new dendrites at the grain boundary and the dendrites in grain B cannot overgrow the dendrites in grain A_1 , the grain boundary lies parallel to the dendrites in grain A_1 ³⁶.

In a recent study from Wagner *et al.*³⁷, bicrystal volume samples from a nickel-base superalloy with diverging and converging dendrites were produced to study the mechanism of competitive grain growth. For diverging grains, there is an excess of solute in the liquid between the diverging dendrites. Consequently, dendrites have to be produced to fill the evolving gap between the diverging grains. Secondary dendrite arms develop in the less favourably oriented grain. This is due to these secondary dendrites being more highly undercooled and being more aligned with the temperature gradient than the secondaries from the well-aligned grains. There is no evidence of significant growth of secondary dendrites to expand the size of the well aligned crystal. The orientation of the grain boundary is, therefore, largely parallel to the more favourably oriented grain. The result differed from the Walton and Chalmers model. In the case of converging dendrites, the result from Wagner *et al.*³⁷ was the same as the prediction of the Walton and Chalmers model. Zhou *et al.*³⁶ also deal with bicrystal volume samples from nickel-base superalloys similar to the experiments published by Wagner *et al.*³⁷. In the case of diverging dendrites the results are in good accordance with the Walton and Chalmers model. In the case of converging grains, the result differs from the prediction of the Walton and Chalmers model. Since new dendrites are hard to develop at the grain boundary, the inclination of the grain boundary plane depends on the blocking of dendrites only. The blocking of the unfavorably oriented dendrites at the grain boundary does not lead to grain boundary moving towards the misaligned grain, while the blocking of the favorably oriented dendrites at the grain boundary leads to grain boundary moving towards the favorably oriented grain. Consequently, the misaligned grain is able to overgrow the favorably oriented grain.

In another study, D'Souza *et al.*³⁸ investigated the role of constitutional undercooling and primary dendrite orientation on the stray-grain nucleation and growth by the use of a cellular-

automaton-finite-difference model. The simulated dendritic structures and the amounts of undercooling predicted are presented under various thermal profiles (level and transient isotherms) and different orientations of the primary dendrites (0 and 15 deg). In case of an undercooling under flat isotherms, level columnar fronts were observed as it can be seen in **Figure 2-11**.

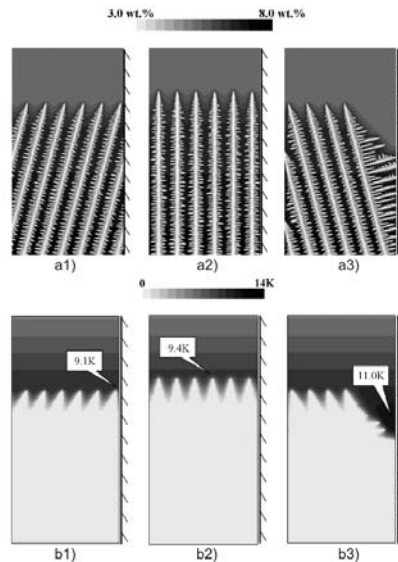


Figure 2-11.: a) Predicted microstructures and b) predicted undercoolings for different dendrite orientations under flat isothermal conditions: 1) converging, 2) axial and 3) diverging³⁸.

The following observations were made by the authors³⁸:

1. Converging: The primary dendrites were terminated when they reached the wall.
2. Axial: All primary dendrites grew at the same rate and had a similar undercooling profile.
3. Diverging: An increasing volume of liquid was formed between the mold wall and the diverging primary dendrites during growth. If the undercooling was lower than the critical nucleation undercooling, secondary and tertiary arms developed to fill up the open body of liquid. Otherwise, nucleation occurred.

In case of an undercooling during the initial thermal transient, the simulated structures for different columnar dendritic orientations including heterogeneous nucleation are given in **Figure 2-12**.

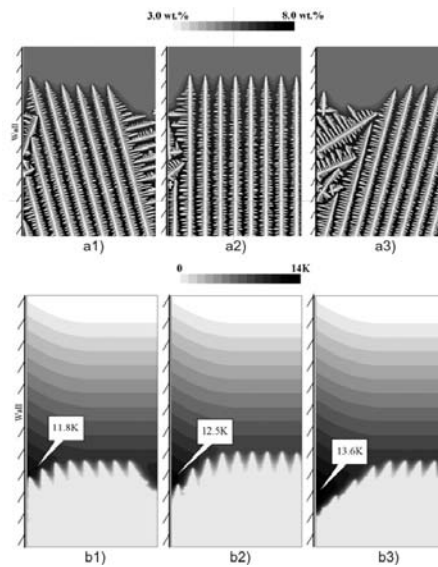


Figure 2-12.: a) Predicted microstructures and b) predicted undercoolings with heterogeneous nucleation for different dendrite orientations under transient thermal conditions: 1) converging, 2) axial and 3) diverging³⁸.

The following observations were made by the authors³⁸:

1. Converging: The primary dendrites near the mold wall were terminated when they reached the wall. The undercooling ahead of the columnar dendrites in the center was the same as the results under level isothermals. New grains nucleated near the mold wall due to the higher undercooling and were overgrown by the growing columnar dendrites.
2. Axial: The columnar dendrites near the mold wall grew at a higher rate during the transient stage than those under level isotherms, resulting in a higher undercooling. New grains formed during the transient stage and were overgrown by the columnar dendrites after the transient stage.
3. Diverging: There is an increasing volume of liquid left between the mold wall and the diverging primary dendrites during growth. Nucleation occurred near the diverging side, as shown in **Figure 2-12 c**. The nucleated grains grew to fill the gap between the mold wall and the columnar dendrites. Due to the open space of the gap left, the stray grains are not likely to be blocked by the original dendrites. Therefore, they could grow into the single-crystal matrix to form columnar grains.

D'Souza *et al.*³⁸ figured out the following points as the main conclusion of their work:

1. For seeds with an axial orientation, the stray grains nucleated across the seed perimeter exhibit a random columnar morphology, showing the absence of an angular relationship.
2. For seeds with an off-axial orientation, the growth of nucleated grains is governed by the morphological constraints associated with competitive growth between the nucleated grains and columnar dendrites. At the converging side, the nucleated grains were overgrown by the columnar dendrites after the transient stage and remain equiaxed. At the diverging side, the nucleated grains can grow significantly into the single-crystal matrix and exhibit a columnar morphology.

However, the fundamentals of the development of the solidification structure are largely understood even though some mechanisms have to be further investigated. At this point, the term solidification structure should be changed in microstructure due to the following reasons. After the formation of the primary grains other important events occur during further cooling. The crystalline transformations of iron occur in the majority of steels and give, during cooling, different superimposed structures. **Figure 2-13** shows the crystalline structures which are normally distinguished: a) The primary or solidification structure (dendritic), b) the secondary or austenitic structure and c) the tertiary or ferritic structure. In steels with a carbon content above 0,5 wt.% the primary structure coincides with the secondary structure. For carbon contents below about 0,5 wt.% the solidification passes through the peritectic phase transition where the primary structure no longer coincides with the secondary structure as it is shown in **Figure 2-13**³³.

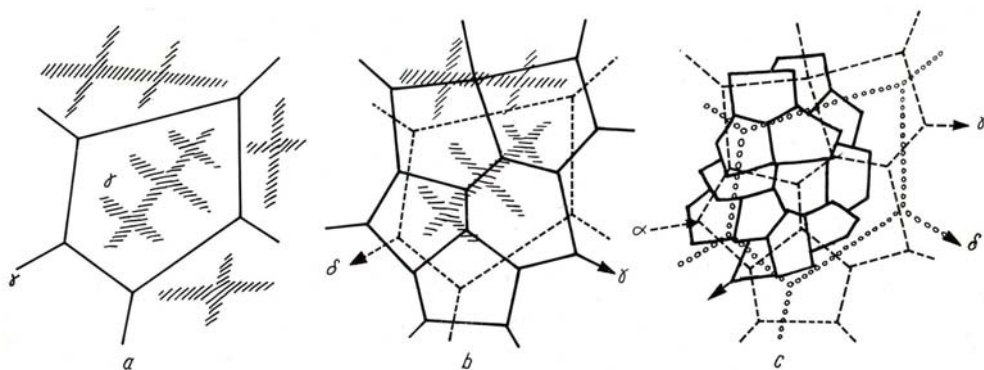


Figure 2-13.: Superposition of various crystallizations in the cast condition for low carbon steels³³.

In the literature the term microstructure mostly deals with the primary and secondary dendrite arm spacing, but also with some descriptors for the secondary or tertiary grains, e.g. the secondary/tertiary grain size or even the interlamellar spacing in pearlite.

2.3. Summary and Conclusions with respect to the continuous casting process

Summarizing, there are some different important events happening during the formation of the solidification structure. Based on the general theory of crystallization a homogeneous and heterogeneous nucleation can appear in an undercooled melt, whereby the heterogeneous nucleation is the most frequent in practical solidification processes. At these nuclei the further solidification takes place and leads to a solid/liquid interface which can grow in different morphologies. The constitution of the morphology is dependent on the temperature gradient G , the growth rate V and the cooling rate \dot{T} . Alloys such as steel show a columnar dendritic solidification, i.e. tree like primary crystals (called dendrites), which are growing contrary to the heat flux during solidification. The groups of dendrites, which are growing with the same orientation, are called primary grains. Primary grain boundaries (PGB) are areas where primary grains adjoin which means the regions where a misorientation of neighbouring dendrites occurs. A schematic illustration of a solidified structure is shown in **Figure 2-14**.

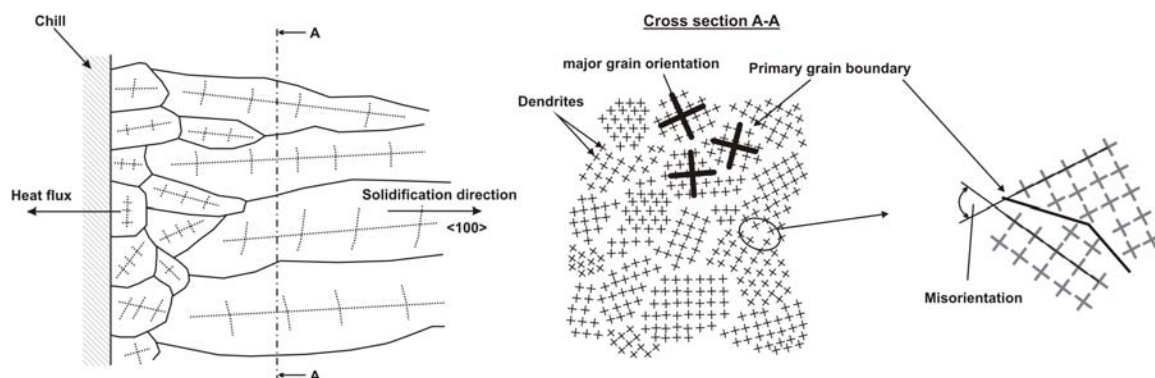


Figure 2-14.: Grain selection during solidification and primary grains consisting of dendrites with uniform orientations according to Pokorny³³.

A fine-grained zone at the surface (initial zone or chill zone) is followed by a zone of columnar grains (columnar zone), sometimes (depending on process conditions) with a zone of equiaxed

grains in the centre (inner equiaxed zone). The mechanism of competitive grain growth can be seen in the left part of the figure, whereby just a few columnar grains can grow further by overlapping those with a bad orientation. This phenomenon is the reason for the varying grain size in different distances from the interface. Considering the middle part of the figure, which shows a cross-section of the solidification structure on the left side, it is conspicuous that there are groups of neighboring dendrites with the same orientation and therefore primary grains. The definition of PBG is clearly illustrated on the right hand side of the figure. The formation of the solidification structure is therefore depending on the chemical composition, the temperature gradient G , the growth rate V and the cooling rate \dot{T} (the product of $G \cdot V$).

3. Concept behind the method

As a big problem in the continuous casting of steel, hot tearing may appear during solidification in the mushy zone (range between the liquid temperature and the solidus temperature). This phenomenon preferentially occurs at primary grain boundaries which are orthographic orientated to the stress direction. The question why hot tearing takes place at PGB can be answered by (1) the different coalescence behaviour of dendrites belonging to the same grain and dendrites at PGB^{2,3}, as well by (2) strain localization^{2,4,5}. For a quantification of the local occurring strain, which can cause a defect formation, the quantity of PGB has to be determined. Another problem is the formation of microsegregations in the area between the dendrites. According to the investigations performed by J. Zhang and R.F. Singer³⁹, the primary dendrite arm spacing λ_1 is shown to be larger across the PGB than in the grain interior. This leads to increased regions for microsegregations and also microporosity. The knowledge of the primary grain size and therefore the quantity of primary grain boundaries is an essential information for a better understanding of solidification and will lead to improved processes, which in turn lead to materials with fewer defects and a better quality.

There are two possibilities for the determination of the primary grain size. As one potential the numeric modeling for the forecast of microstructure and morphology should be mentioned. Introductory and basic material can be found in the standard work by Kurz and Fisher¹. A more special overview for the modelling of microstructures was arranged by Rappaz⁴⁰ in 1989. Newer researches relating to this topic were published by Stefanescu⁴¹ in 1995 and Boettinger⁴² in 2000. Another chance for the evaluation of the primary grain size is the manual encountering by hand or automatically by the use of a software. Existing machine vision based methods are just able to evaluate secondary or tertiary grain boundaries because of their visible

shape but no primary grain boundaries. It must be pointed out that in contrast to the secondary and tertiary grains the primary grains have no visible grain boundaries, which makes it harder to detect them. This can be explained by the nonexistence of a second phase along the grain boundaries which is resistant during etching and therefore visible on a micrograph, as it can be seen in **Figure 3-1 a**. In this figure the pro-eutectoid ferrite clearly indicates austenitic grains. Concerning high equivalent carbon contents little ferrite is visible along the austenite grain boundaries therefore the etching result is not that clear. If a high amount of ferrite is obtained (i.e. low carbon contents), a lot of ferrite grains can be found in the micrograph therefore the contrast differences between grain boundaries and the grain are hence not optimal. Due to this the best visual contrast between the secondary grain boundaries and the grain can be obtained in a carbon range of 0.15 to 0.16 wt.%C⁴³. **Figure 3-1 b** shows the primary grain structure of a steel sample with 0.7 wt.%C etched with picric acid. The examined area is orientated perpendicular to the solidification direction so that the dendrites are shown in their cross-section. A dendrite in its cross-section has the shape of a crux because of its secondary arms. In order to evaluate the primary grains it can be judged the group of cruxes with the same orientation.

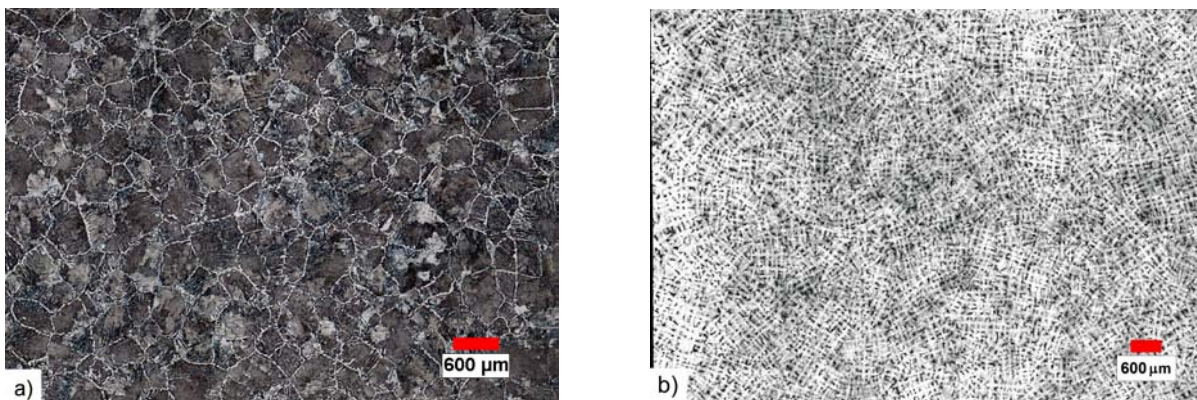


Figure 3-1.: a) Austenite grains in a steel with 0.4 wt.%C (etched with nitric acid in alcohol)⁴³ and b) Primary structure without visible grain boundaries in a steel with 0.7 wt.%C (etched with picric acid).

Therefore, a digital image processing software for the evaluation of the primary grain size should be developed. The concept behind the developed software is based on the calculation of so-called intensity profiles (IP) along the secondary dendrite arms between two neighboring dendrites. If the IP are on a high level it can be suggested, that the two considered dendrites have the same orientation. The IP are calculated in each case from the center of a dendrite

to the center of its neighboring dendrite. In order to find the centers of the dendrites, the image is first binarized and then passed through a thinning operation. The aim of the thinning is to reduce the shape of a dendrite in the binary image to a minimum width (one pixel) representing the general form of the dendrite. These so generated lines, which are similar to the main axes of the dendrites, intersect in the center of the dendrites. By finding these intersection points it is possible to locate the coordinates of the centers of each dendrite in the image.

In order to give a short overview about the general methods of image processing, the next chapter describes the steps from the initial image acquisition up to the concept of object recognition.

4. Image Processing

Jähne⁴⁴ describes the classical image processing as a hierarchical structured process of the necessary image operations, as it is illustrated in **Figure 4-1**. The image analysis is understood here as a multi-level process from the picture recording to the picture understanding. The structuring does not contain any feedbacks between the individual processing steps, but tries to divide the analysis into clearly separated monolayers. The individual operations can be divided also roughly into the categories picture version, picture improvement, segmentation, picture symbol processing and pure symbol processing. The picture version designates the physical production and collection of the picture with a sensor. Before the actual analysis can be begun, an improvement of the picture is often necessary. The operations of the picture improvement work in the recorded pictures, so that they have an image for the input as for the output. Under the generic term segmentation operations are summarized, which identify and mark picture regions. Subsequently, a purely symbolic description from the characteristic picture, which still contains iconic representations of the regions and characteristics, has to be generated. Here now arbitrary, purely symbolic processing levels follow. According to Jähne each conceivable operation can be arranged in this hierarchy. The aim of this classical processing chain is the production of a symbolic description of the iconic picture ("description of objects" and "object classes"). A frequent intermediate objective is the so-called region picture⁴⁵.

In the more recent time there are however more applications, which do not make the detour over a symbolic description in the conventional sense. Thus for example, reactions to certain visually seized situations or attractions are learned directly in the pictures, without a description of the picture content in a processing level, neither a classification of the captured objects. A further large field, in which contents are not explicitly represented any longer, is the "content

based image retrieval". There are systems existing which find to one submitted picture or a simple sketch pictures with same contents due to the similarity to the collecting main (Query by example). In many systems not the similarity of a symbolic description of the picture content, but a "abstract" description of picture is compared⁴⁵.

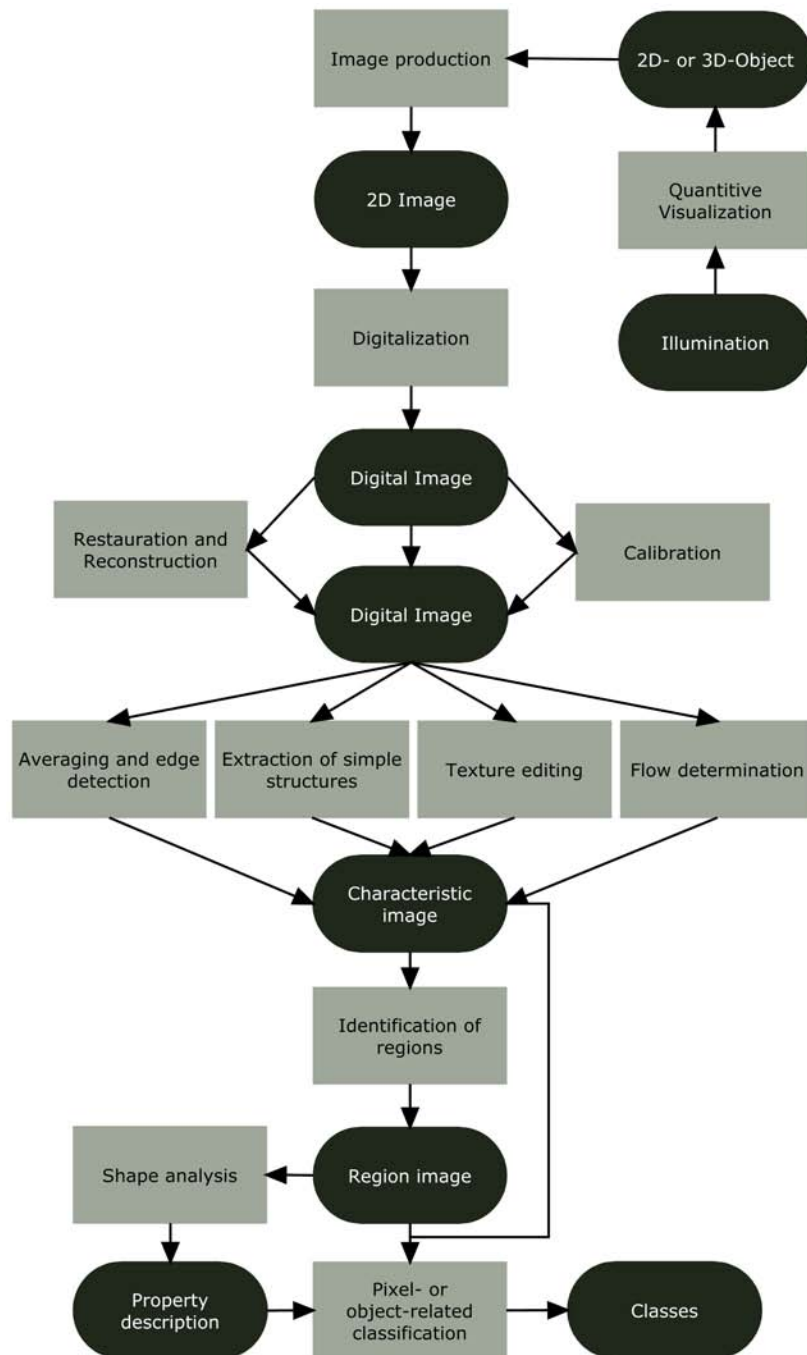


Figure 4-1.: Hierarchy of the image processing operations according to Jähne⁴⁴. Rectangles correspond to the operations, ovals to the data structures.

4.1. Image acquisition

The term image acquisition covers all steps involved in transforming the image of a microstructure into a set of 2-D pictures that provide information on the form and position of the microstructural features. Each material and microstructural element may require an individually designed imaging technique and these techniques are not discussed here - they are the subjects of a number of monographs devoted to separate experimental imaging methods such as: electron and light microscopy, ultrasonic microscopy, radiography and so on. Whatever the imaging technique is, it produces a 2-D image that can be digitalized or recorded either on a film or displayed on a monitor. An image contains specific patterns of points and regions, which differ in their intensity or color. From this point of view, the image is a function of two spatial coordinates, for instance (x,y) , which ascribes to each point on the plane a specific value of point image intensity⁴⁶.

4.2. Segmentation

An important intermediate objective of the processing route is the region image (see **Figure 4-1**). The techniques used for the production of the region image are often summarized under the generic term "segmentation". Originally "segmentation" actually designates only the procedure of the separation, and/or emphasis of the image foreground over the background. In addition, in the broader sense the identification and demarcation of individual picture regions is added. However, these picture regions may not be confounded with an identification of the object surfaces. Picture regions are thus not identical to object surfaces. In order to be able to create such a relationship, further, mostly symbolic processing steps are necessary⁴⁵.

4.2.1. Edge detection

A possibility of finding the picture regions leads across the region borders themselves. With the help of so-called edge detectors these borders can be found. An edge is thereby mostly an abrupt change of brightness. Most procedures use a folding of the examined picture for the computation of an "edge image" with a certain Kernel - the edge detector. Result is a pixel matrix, which contains a value, whose amount indicates the "edge strength" per pixel.

By "folding" the following operation concerning two-dimensional, "discrete" pixel matrices is understood:

$$G(x, y) = \sum_{a=1}^A \sum_{b=1}^B F(x+a-1, y+b-1)K(a, b) \quad (4-1)$$

whereby $F_{M,N}$ is the picture matrix and $K_{A,B}$ with $A \leq M$ and $B \leq N$ is the so-called "Kernel" or "convolution mask". The result of this operation is the smaller matrix $G_{M-A+1, N-B+1}$, whereby the operation for each place $G(x, y)$ of the resulting matrix must be computed. For the computation of the resulting matrix the Kernel is moved over the picture matrix. The entries of the Kernels are multiplied by the pixel being "under it". The sum of these products forms then the entry in the current place $G(x, y)$. Depending upon the Kernel the sign can likewise possess a meaning, e.g. over the color tone (bright after darkness, darkness after bright) of the edge. These pictures are usually standardized for representation and subsequent treatment, respectively. With the help of a threshold value only region edges of a desired intensity can be let through⁴⁵.

4.2.2. Global/Constant thresholding

Other methods try not to identify the edges but the picture regions directly based on the color values. With "global thresholding" a constant threshold value T is used, in order to decide separately for each pixel on the basis of the brightness value, into which of the two classes it falls. These classes are "background" and "foreground", here designated with 0 and 1. If a multi-channel image, e.g. $f(x, y) = (r, g, b)$, is present, a function $l(r, g, b)$ can be used which computes the luminosity so that the image transforms into a grey value image. Of course there are also special colorsegmentation methods, which work directly with the entire channels. The segmented image $s(x, y)$ can then be computed by the simple decision whether the luminosity lies under or above the threshold value:

$$s(x, y) = s(l(f(x, y))) \quad (4-2)$$

$$s(l) = \begin{cases} 0, & l \leq T \\ 1, & l > T \end{cases} \quad (4-3)$$

In this binary image a region is now defined over the neighborhood of the pixels: All pixels, which got the same value assigned and in the sense of the 4er or 8er neighborhood, belong to the same region. On the basis of one pixel of a region, the entire region can be identified. Problems can happen for this simple procedure due to strong lighting gradients or bright objects in front of a dark background, on which a strong shade is thrown. In such situation frequently no threshold value with satisfying quality (in the sense of correctly classified pixels) will be found⁴⁵.

4.2.3. Adaptive thresholding

In order to correct the failure due to a non-uniform illumination, an adaptive threshold which changes the threshold dynamically over the image can be used instead of the global threshold. Such "adaptive" thresholds can adjust brightness differences between different picture regions quite well, if there are coherent ranges which were relatively uniformly illuminated.

There are two main approaches to finding the threshold: (1) the Chow and Kaneko approach and (2) local thresholding. The assumption behind both methods is that smaller image regions are more likely to have approximately uniform illumination, thus being more suitable for thresholding⁴⁵. Chow and Kaneko divide an image into an array of overlapping subimages and then find the optimum threshold for each subimage by investigating its histogram. The threshold for each single pixel is found by interpolating the results of the subimages.

An alternative approach to finding the local threshold is to statistically examine the intensity values of the local neighbourhood of each pixel. The statistic which is most appropriate depends largely on the input image. Simple and fast functions include the mean of the local intensity distribution, the median value or the mean of the minimum and maximum values. The size of the neighbourhood has to be large enough to cover sufficient foreground and background pixels, otherwise a poor threshold is chosen. On the other hand, choosing regions which are too large can violate the assumption of approximately uniform illumination⁴⁷.

4.2.4. Regiongrowing

A further method for direct identifying of regions is the so-called "Regiongrowing". For the Regiongrowing also a neighborhood relation - usually the 4er or 8er neighborhood - and an additional homogeneity condition for regions is defined

$$H(R_i) = \begin{cases} TRUE \\ FALSE \end{cases} \quad (4-4)$$

whereby in the final region picture then must be valid $H(R_i) = TRUE$ for all R_i and $H(R_i \cup R_j) = FALSE$ for all neighboring regions R_i, R_j . Two regions are here neighbouring if two arbitrarily pixel $x_i \in R_i$ and $x_j \in R_j$ stand in the neighborhood relation. As homogeneity criteria are applicable for example uniform brightness, color or texture as well as arbitrary combinations. By means of this definition of a "valid" region picture a region image can be computed iteratively directly from the pixel or before-segmented picture. Essentially two approaches are to be differentiated:

1. In "Region Merging" with an over-segmented image is begun (e.g. pixel image). The regions are merged iterative with the help of a fusion criterion, until the conditions defined above apply. As fusion criterion for example $H(R_i \cup R_j)$ can serve directly.
2. In "Region Splitting" with an under-segmented image is begun (e.g. entire picture as one region) and the regions are iterative divided (e.g. quartered), until $H(R_i)$ is fulfilled for all regions.

The two procedures can supply quite different results for the same picture. For reasons of the efficiency it should be noted that for an implementation a data structure is usually needed for representing the neighbors. Recent testing of the neighborhood between the regarded with all other regions is too complex due to the definition over the neighborhood of two "element pixels". The procedure that is actually frequently used, is a combination of these two procedures. In the "Split and Merge" procedures the steps of splitting and merging alternate. The neighbors are represented here in a Quadtree, which is developed and updated with the individual iterations. The regions are therefore always quartered when splitting⁴⁵.

4.3. Region description

After the regions were identified and coded in a binary picture, it is useful to find a more space-saving description which can be worked on more efficiently. The iconic representation is now changed to the symbolic processing. Therefore, usually as topologically as possible invariant

descriptions are desired. The multiplicity of the possible representations can be divided again roughly in two classes⁴⁵.

4.3.1. Contour-based descriptions

Contour-based representations describe the edge characteristics of the region. Usual codings are e.g. the chain codes or the Fourier-descriptors. On these descriptions again contour-based measures can be defined, which can be consulted for a comparison of two regions. Typical is here for example the curvature, the edge length or the perimeter, or also the bending energy⁴⁵.

4.3.2. Region-based descriptions

Region-based descriptions try to analyze the form of the region, whereby the accurate edge characteristics can be lost. Examples are e.g. the skeletonization ("non-dividing" erosion of the region up to a 1-pixel thick skeleton) or all decompositional descriptions (e.g. by means of a tree). Also on these descriptions then region-based comparison measures can be computed, depending upon the selected description more or less simply. Typical measures are here: Surface, center, height and width, the eccentricity (the relationship between main and secondary axis), the oblongness (largest diameter by surface) or the compactness (perimeter by surface)⁴⁵.

Because of the fact, that the skeletonization operation is used in the developed software, a deeper explanation of this operation is given here in detail. The aim of the skeletonization is to extract a region-based shape feature representing the general form of an object. In other words, after pixels have been peeled off, the pattern should still be recognized. The notion skeleton was introduced by H. Blum⁴⁸ as a result of the Medial Axis Transform (MAT) or Symmetry Axis Transform (SAT)⁴⁹. The MAT determines the closest boundary point(s) for each point in an object. An inner point belongs to the skeleton if it has at least two closest boundary points, as it can be seen in **Figure 4-2**. A very illustrative definition of the skeleton is given by the prairie-fire analogy: the boundary of an object is set on fire and the skeleton is the loci where the fire fronts meet and quench each other. The third approach provides a formal definition: the skeleton is the locus of the centres of all maximal inscribed hyper-spheres (i.e., discs and balls in 2-D and 3-D, respectively). An inscribed hyper-sphere is maximal if it is not covered by

any other inscribed hyper-sphere. Skeletons are consisting of two types, the structural skeleton and the textural skeleton⁵⁰.

There are three major skeletonization techniques:

- detecting ridges in distance map of the boundary points;
- calculating the Voronoi diagram generated by the boundary points; and
- the layer by layer erosion called "thinning".

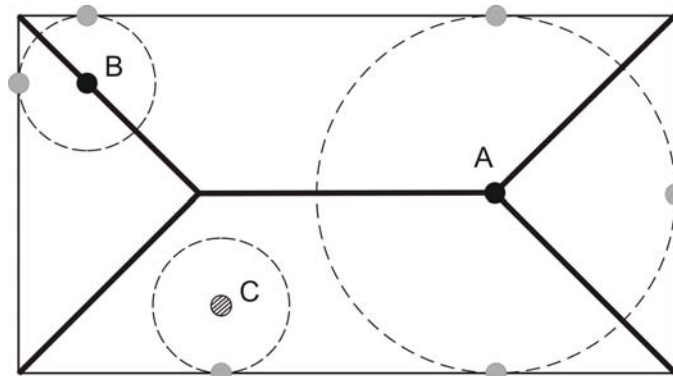


Figure 4-2.: Skeleton of a rectangle. Both points A and B are skeletal, but point C does not belong to the skeleton. The skeleton is marked by thick black line segments.

In digital spaces, only an approximation to the "true skeleton" can be extracted. There are two requirements to be complied with, the topological to retain the topology of the original object and the geometrical forcing the "skeleton" being in the middle of the object and invariance under the most important geometrical transformation including translation, rotation, and scaling.

The thinning has some beneficial properties:

- It preserves the topology (retains the topology of the original object);
- it preserves the shape (significant feature suitable for object recognition or classification is extracted);
- it forces the "skeleton" being in the middle of the object; and
- it produces one pixel width "skeleton".

A thinning operation transforms "thick" lines forming closed figures in the image into a line grid of unit width. Thinning is an anti extensive operation, i.e., it transforms some object points, brightness 1, into background points, brightness 0, whereas background points are not transformed into object points. In Euclidean space, a thinning operation, denoted by \odot , is a superposition of k operations employing structural elements, E_j , generated as a result of rotation of element E , by assigned angles $\alpha_1, \alpha_2, \dots, \alpha_k$ ⁴⁶.

$$\odot : \hat{j}\{P_i \in O \quad C_{OE}(P_i) \cap E_j = E_j \Rightarrow P_i = 0\} \quad j = 1, 2 \dots k \quad (4-5)$$

E_j - E rotated by α_j

The operations are repeated until the whole cycle $j = 1 \dots k$ does not change the image O . A typical structural element used in a square grid for a thinning operation is given as follows:

Table 4-1.: Structure elements for a thinning operation⁴⁶.

initial element	after α_1 rotation	α_2	α_3	α_4
E	E_1	E_2	E_3	E_4
1 1 1	1 0	0 0 0	0 1	1 1 1
1	1 1 0	1	0 1 1	1 etc.
0 0 0	1 0	1 1 1	0 1	0 0 0

4.4. Object recognition

With the help of the existing descriptions of the picture regions, which can be worked on efficiently, more complex symbolic procedures for the object recognition and/or - classification can be performed. Such a classification cannot be accomplished any longer alone with the graphic data, now additional knowledge is needed. For the classification of the objects different techniques stand to the selection. Statistic procedures from the pattern recognition, neuronal networks and the syntactic pattern recognition are here exemplary mentioned as representatives⁴⁵. A description in detail of these techniques and further optimization procedures is not given here until these techniques are not used in the developed software.

5. Processing steps of the developed software

In this chapter the single procedures of the developed software are explained in detail. The code was generated in MATLAB[®] R2007b since this commercial software package enables an easy handling with matrices and provides an image processing toolbox with useful pre-programmed functions. The concept of a digital image is strictly related to the technical conditions of recording the images. Two-dimensional images of the solidification structure can be viewed as sets of different color/gray level dots filling the picture. Every image can be divided into a finite number of elements (pixels) N_p , dependent on the number of rows N_r and columns, N_c , assumed in the division of a given image. The pixels forming the image are distinguished by different gray levels. If the intensity of the gray tone is measured on a scale ranging from 0 to N_b , then the image analyzed can be represented as a matrix M with a size $N_r \times N_c$, with elements p_{ij} , which are of integer value in the range 0 to N_b . The value zero in this case indicates a black pixel and N_b a white one. The numbers $0 \leq p_{ij} \leq 1$ distinguish intermediate gray levels⁴⁶.

$$M = \begin{pmatrix} p_{11} & p_{21} & \dots & p_{1N_c} \\ p_{21} & p_{22} & \dots & p_{2N_c} \\ \vdots & \vdots & \ddots & \vdots \\ p_{Nr1} & p_{Nr2} & \dots & p_{NrN_c} \end{pmatrix} \quad (5-1)$$

The pixels of the image have prescribed geometrical dimensions. If the total area of image analyzed is $A \times B$ then the pixels are quadrilaterals of a size $p_r \times p_c$ such that:

$$p_r = \frac{A}{N_r} \qquad p_c = \frac{B}{N_c} \qquad (5-2)$$

This means that any point in the image has specified coordinates with respect to the frame and can be addressed to an appropriate element of the solidification structure studied.

5.1. Binarization

The images, which were taken from the etched steel samples with a reflected-light microscope, are color or gray scale images. **Figure 5-1** shows an original gray scale image from a steel sample with 0.7 wt.%C etched with picric acid. Polishing and etching often leads to significant differences in the intensity of light reflected from the parts of the area studied that are occupied by different phases or segregations.

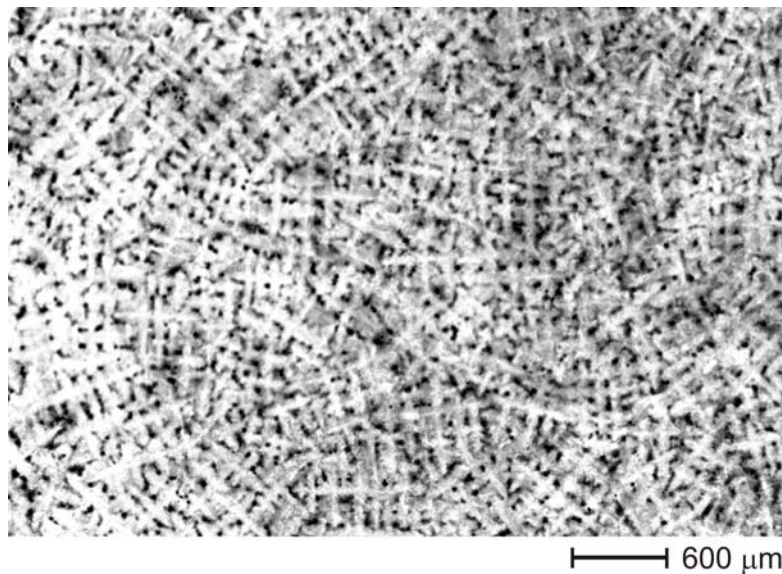


Figure 5-1.: Original gray scale image of a 0.7 wt.%C steel etched with picric acid.

The first step is to convert these images into a binary image, which means a matrix consisting only of zeros and ones. This is done by the use of a threshold operation. Thresholding is used to segment an image by setting all pixels whose intensity values are above a threshold to a foreground value (one) and all the remaining pixels to a background value (zero). The use of

a global threshold for all pixels leads to a loss of information in case of an image with non-uniform illumination (strong illumination gradient), as it can be seen in **Figure 5-2**. This can happen because of a difference of the etching depth over the steel sample, caused by unequal segregations.

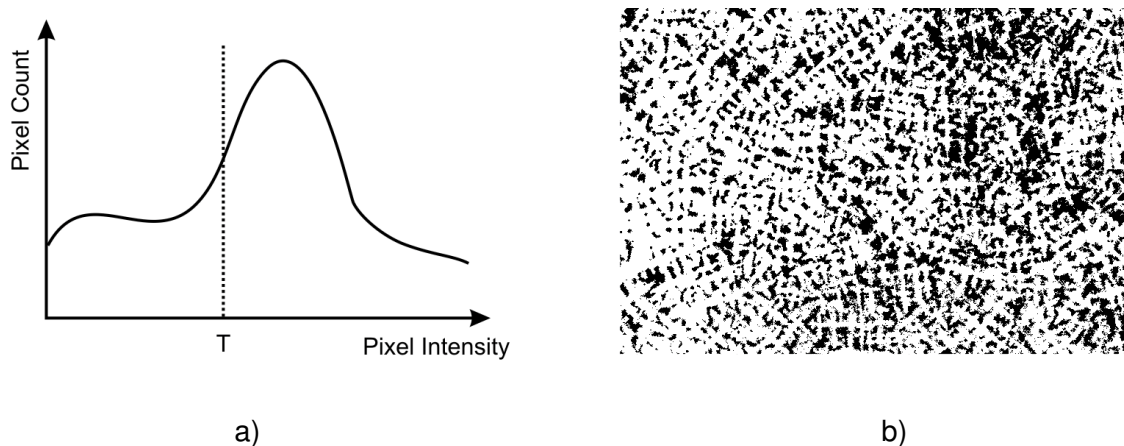


Figure 5-2.: a) Schematic illustration of an intensity histogram with a threshold T and b) the result of a global threshold.

Instead of this, an adaptive threshold was chosen which changes the threshold dynamically over the image. The functioning of this operation is described in Chapter 4.2.3. **Figure 5-3** shows a comparison between the results from a global and an adaptive thresholding applied on the original image (see **Figure 5-1**). Considering the upper right portion of the image, the result of the adaptive threshold contains much more information. Also on the left side, where the strong illumination leads to thick connectedly foreground objects using the global threshold, the adaptive threshold converts this region more precise.

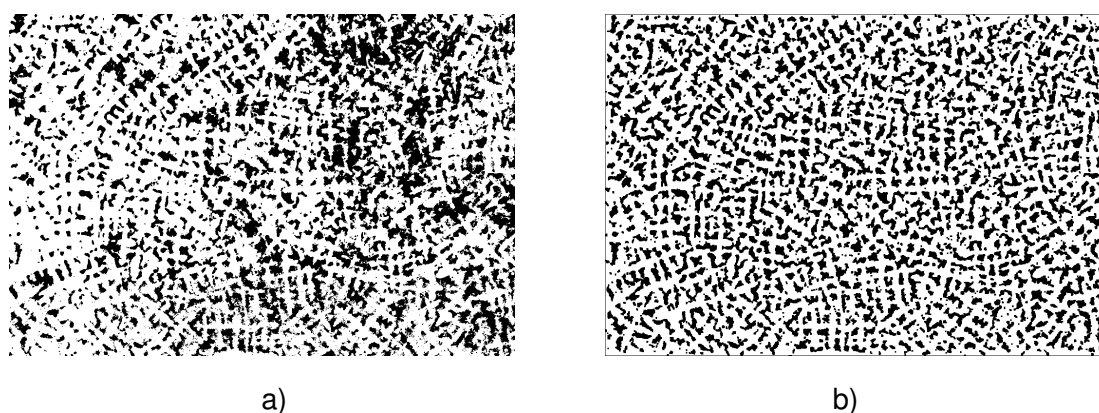


Figure 5-3.: a) Binary image using a global threshold and b) using an adaptive threshold .

5.2. Smoothing

In order to smooth the abrasive pixelated shape of the foreground objects and to arrange a noise removal, a smoothing operation is adopted. Therefore, the image is eroded by the use of a diamond-shaped (**Figure 5-4a**) or a octagonal-shaped (**Figure 5-4b**) morphological structure element (SE) with a specific size, followed by a dilation of the same.

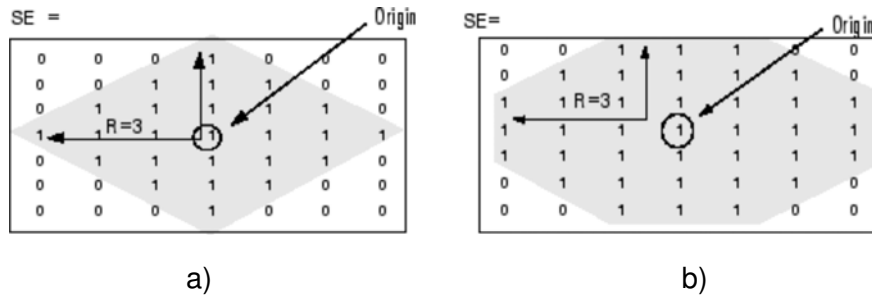


Figure 5-4.: a) Diamond and b) octagon structure elements⁵².

The result of a smoothing operation with a diamond-shaped structure element is shown in **Figure 5-5b**. The shape of the foreground objects is smoothed resulting in a more precise shape representing the dendrites.

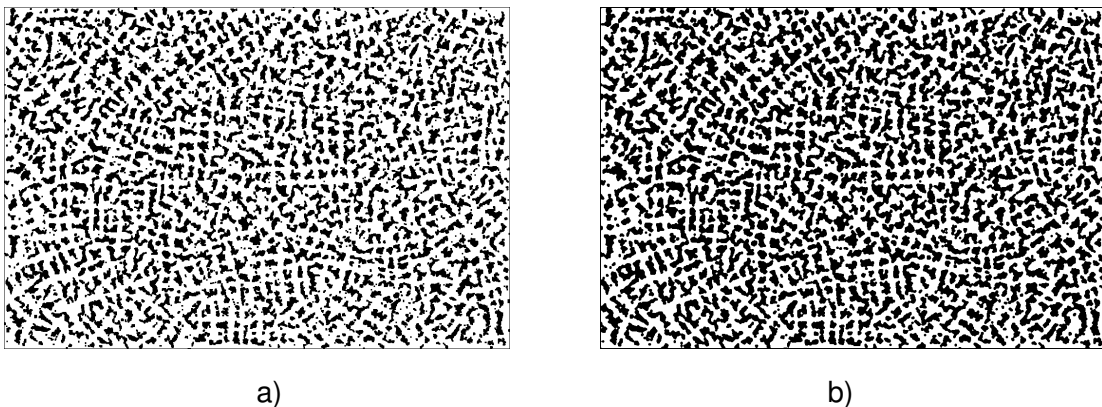


Figure 5-5.: a) Binary image using an adaptive threshold and b) after a thinning operation (SE = diamond).

With respect to the following thinning operation, which is described in the next section, the smoothed image leads to better results than the unsmoothed binary image. This can be explained by a less branching of the created skeleton.

5.3. Skeletonization / Thinning

Figure 5-6 shows the principle of the thinning operation applied on a dendrite in a binary image. The white foreground area, presenting ones in the binary image, is reduced to two lines of minimum width (i.e. one pixel). These "skeleton-lines" are indicating the general shape of the dendrite and constitute the main axes, providing a symmetric shape of the dendrite. The intersection point of the two lines are representing the centre of the dendrite.

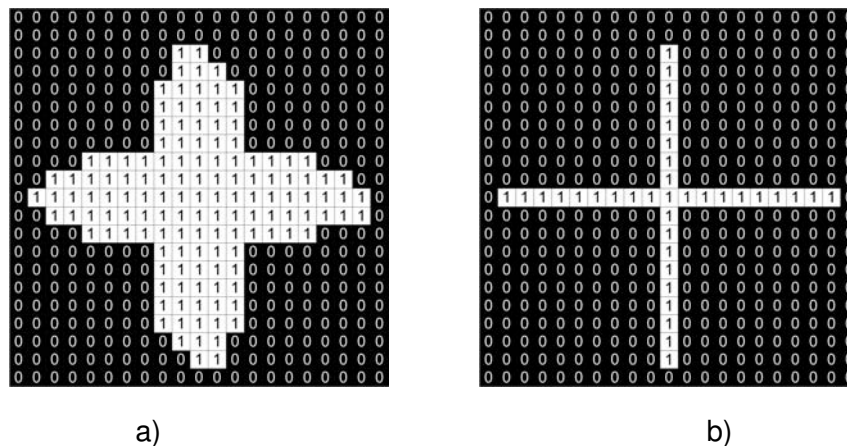


Figure 5-6.: Example of a thinning operation. a) Dendrite in a binary image and b) after the thinning operation.

Figure 5-7 shows a comparison of the skeletonization and the thinning operation applied on the shape of a dendrite in a binary image. Concerning the inner region of the dendrite both operations nearly lead to the same shape of the skeleton but at the end of the side arms there are significant differences. In the latter region, skeletonization produces branched lines in order to reconstruct the shape of the border, whereas the thinning operation represents only the general shape (topology) of the dendrite. For the development of a digital image processing software to evaluate the solidification structure in steel, the implementation of the thinning operation leads to better results with respect to the quantity of skeleton intersection points. The result of the thinning operation in **Figure 5-7 c** leads to one intersection point in the centre of the dendrite, whereas the skeletonization (**Figure 5-7 b**) generates 10 intersection points, one in the centre and 9 others due to branching at the side arms. Due to the amount of excess intersection points it is not possible to describe the center of a dendrite with only one intersection point in the center. This is the reason why the thinning operation is more useful in this application than the skeletonization.

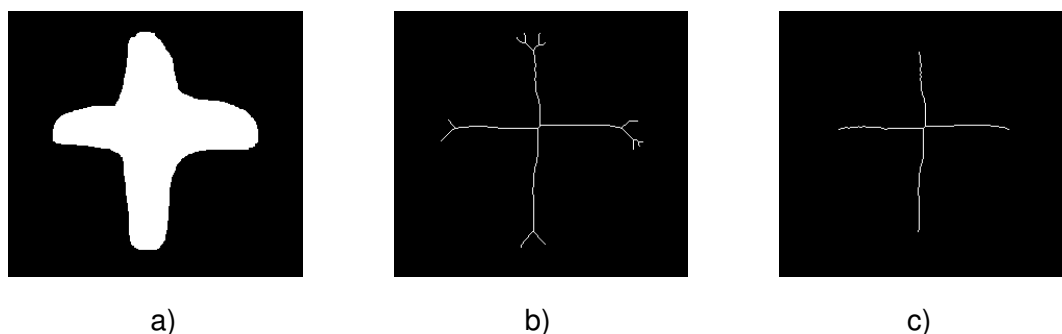


Figure 5-7.: a) Shape of a dendrite in a binary image, b) result of skeletonization, c) result of thinning.

Figure 5-8 shows the result of a thinning operation applied on a practical image (original image in **Figure 5-1**). The intersection points of the thin white lines are indicating the coordinates of the dendrites. Even in this image the information about the orientation of the dendrites is contained and is shown as the slope of the white lines.

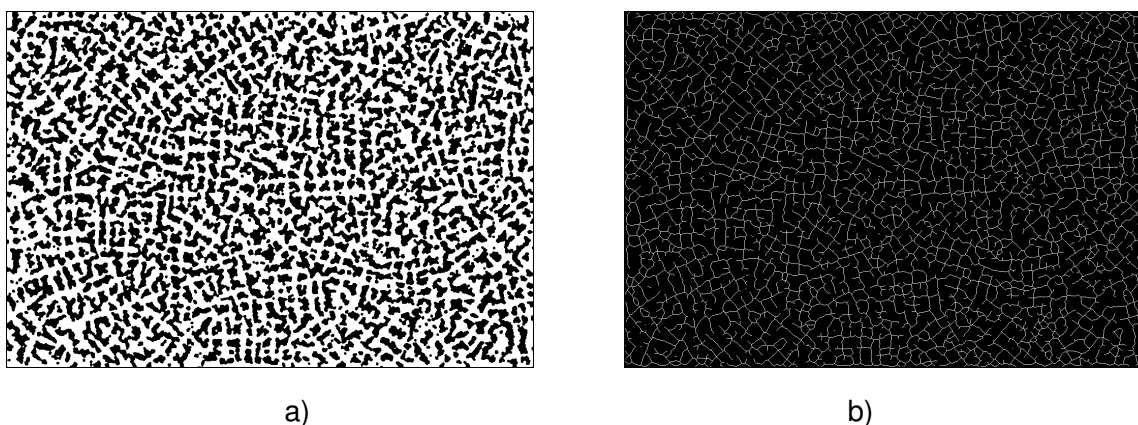


Figure 5-8.: a) Smoothed binary image and b) the appropriate thinned image.

5.4. Searching for skeleton intersection points

Skeleton intersection points are points, where at least two skeleton-lines intersect. Using this definition makes it possible to describe the centre of a dendrite by only one point with fixed coordinates x_n and y_n in the image. An intersection point is, according to this, a replacement character for a dendrite. If the shape of a dendrite differs to the general symmetric form (**Figure 5-9 a**), caused by illumination gradients, thresholding or even various effects during solidification, the generated skeleton produces two intersection points due to the constraint of the skeleton being in the middle of the object (see Chapter 4.3.2). In this case, a merge of

the intersection points has to be made to set the centre of the dendrite to one point forming the mean of the coordinates of the excess intersection points (**Figure 5-9 b**).

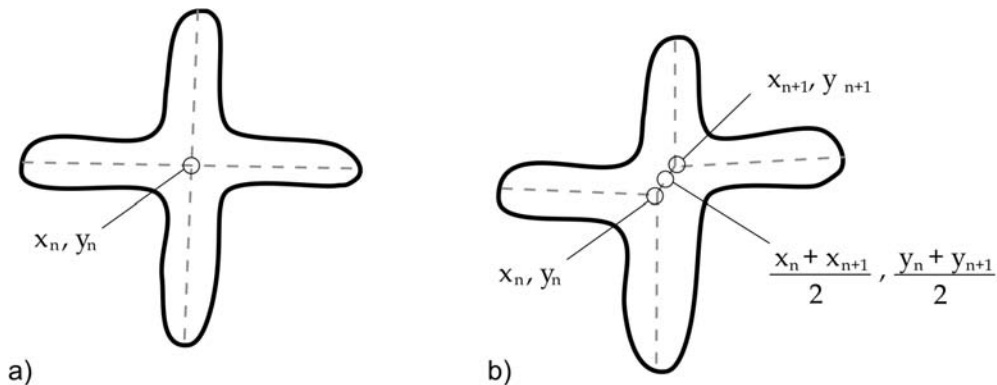


Figure 5-9.: Irregular shape of a dendrite leads to more than one intersection points.

The result of this step is a matrix named *intersection_pts* containing the x-coordinates of all intersection points in the first column and the y-coordinates in the second column. This matrix is an essential part of the software whereon the whole calculation is based. It must be mentioned, that, depending on the quality of the binary image, there is a certain amount of excess intersection points generated which are not relating to a center of a dendrite. This happens due to the irregular shape of some dendrites which leads in turn to a branching of the skeleton. Because of the fact, that these excess intersection points are mostly generated on secondary dendrite arms near the center, they have no negative influence to the calculation of the following intensity profiles.

$$intersection_pts = \begin{pmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \\ x_n & y_n \end{pmatrix} \quad (5-3)$$

5.5. Nearest neighbours

An often used method in this developed software is the so called "nearest neighbour method". It computes nearest neighbours to a set of points (P) from a set of candidate points (X) using euclidean distances⁵¹.

The euclidean distance in 2-D for two points

$$P = (p_x, p_y) \quad X = (x_x, x_y) \quad (5-4)$$

is calculated by

$$d(P, X) = \sqrt{(p_x - x_x)^2 + (p_y - x_y)^2} \quad (5-5)$$

This is done for every single intersection point within a defined radius R , as it is shown in **Figure 5-10**. In case of the merging of intersection points, the radius is set very small in a dimension smaller than the size of a dendrite in order to eliminate the excess intersection points in the middle of a dendrite (**Figure 5-9 b**).

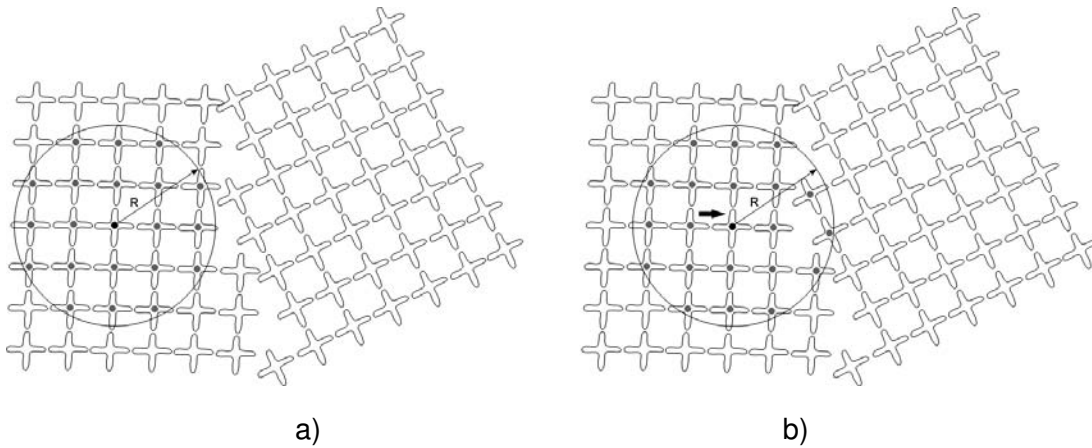


Figure 5-10.: Searching for the nearest neighbours for every intersection point in a defined radius R .

An abstract of the source code for the nearest neighbor determination is shown in **Figure 5-11**. The input parameters are the matrix *intersection_pts*, the number of neighbours (noNeighbours) and the radius (Radius). P is the transposed of the current intersection point. The matrix X is the transposed of the matrix *intersection_pts* without the current intersection point to avoid counting the current intersection point as the nearest neighbour.

```

[rows, cols] = size(intersection_pts);
%
for i = 1:rows
    P = intersection_pts(i,:)';
    inter_pts = intersection_pts;
    inter_pts(i,:) = [];
    X = inter_pts';
    I = nearestneighbour(P, X, 'n', noNeighbours, 'r', Radius);
    n = length(I);
    neighbours = zeros(n,2);
    %
    for j = 1:n
        neighbours(j,:) = inter_pts(I(j),:);
    end;
    %
    ...
    ...
    ...
end;

```

Figure 5-11.: M-Code for detecting the nearest neighbours.

The result of this step is a matrix named *neighbours* for every intersection point containing the coordinates of all neighbours within the chosen radius.

$$neighbours = \begin{pmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \\ x_n & y_n \end{pmatrix} \quad (5-6)$$

5.6. Intensity Profiles

The *improfile* function in MATLAB[®] calculates the intensity values along a line segment or a multiline path in an image. This is used to distinguish whether a nearby intersection point (actually a dendrite) has the same orientation than the currently considered one or not. **Figure 5-12** illustrates the principle of this method. There are two possibilities how the dendrites can be arranged, like in **Figure 5-12 a** side by side or like in **Figure 5-12 c** side by side and also in-between.

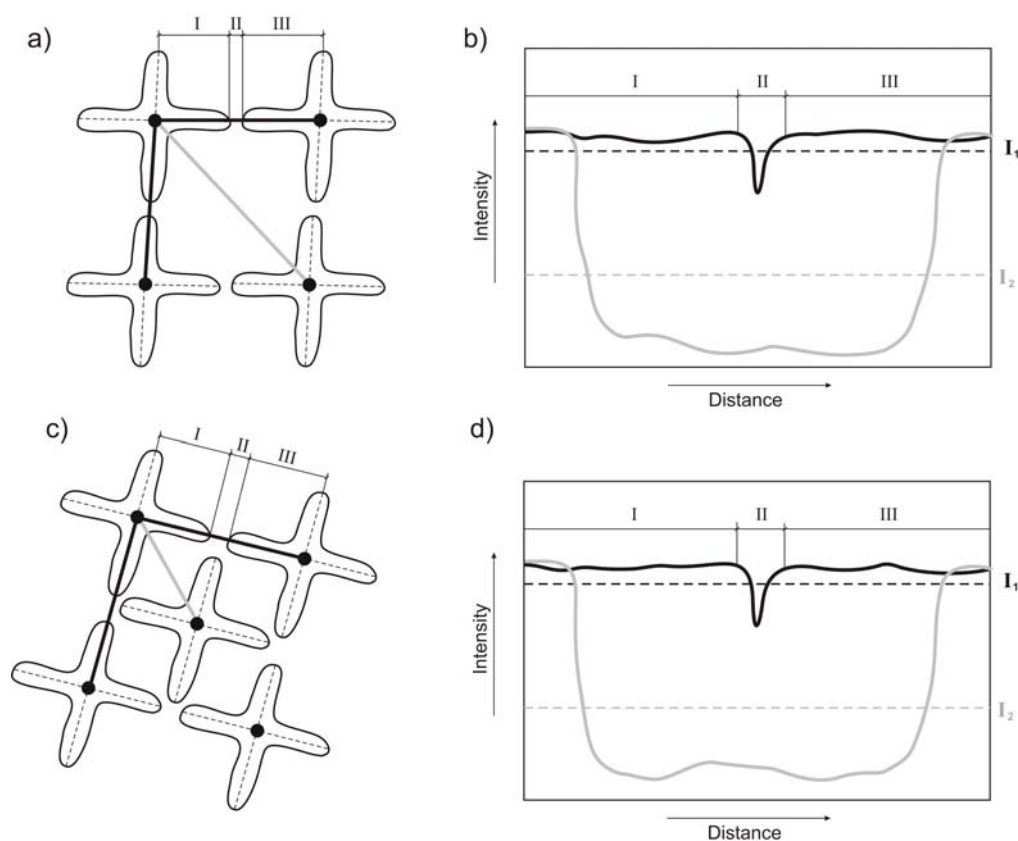


Figure 5-12.: Schematic illustration of intensity profiles on different arrangements of the dendrites. a, c) Lines between the actual intersection point and its neighbours and b, d) the plotted intensity profiles.

According to **Figure 5-12 a**, the intensity profile (IP) between the actual considered intersection point and the next neighbours is taken on the original image. The gray scale intensity gradient along a dendrite arm is very small since the illumination is nearly constant (**Figure 5-12**, region I and III), but when the dendrite is left the intensity decreases due to the illumination gradient caused by the different etching depth (region II). When the secondary dendrite arm of the neighbouring dendrite is reached, the intensity increases again until the nearby intersection point is reached. In the case of at least two dendrites with the same orientation side by side, the intensity profile is on a high level over a long distance resulting in a high mean intensity level. The presentable intensity profile is assigned by a high plateau with only a short decrease of the intensity. Considering the IP between to intersection points, whose according dendrites are not equal orientated, the IP decreases earlier when the dendrite is left (the line for the intensity profile is not consistent with the orientation of the secondary arm) and increases at a higher distance not until the second considered dendrite is reached. The lower plateau in the latter

case is held over a much longer distance so that the mean intensity level is lowered in comparison to the first case. By the use of the nearest neighbour method a matrix named *neighbours* is defined for every intersection point. After that, the IP from the currently considered intersection point to every entry in *neighbours* are assigned and saved in a matrix named *Intensities*. The criterion how the intensities are calculated can be set as the mean of the intensity values or the ratio from the maximum intensity value to the mean (**Equation 5-8** and **Equation 5-9**). The first column in *Intensities* is filled with the mean intensity values or the ratio I_n , the second column is filled with the length of the line for the intensity profiles L_n (distance between the two intersection points).

$$Intensities = \begin{pmatrix} I_1 & L_1 \\ I_2 & L_2 \\ \vdots & \vdots \\ I_n & L_n \end{pmatrix} \quad (5-7)$$

with

$$I_n = \text{mean}(IP) \quad (5-8)$$

or

$$I_n = \min \left(\frac{\max(IP)}{\text{mean}(IP)} \right) \quad (5-9)$$

The intensity value in the n^{th} row of *Intensities* correlates to the n^{th} entry in *neighbours*, so that the maximum intensity value can be allocated to one defined intersection point. By finding this maximum intensity value in *Intensities* it is now possible to specify one point in *neighbours*, with which the actual considered intersection points leads to the highest intensity value (**Figure 5-13 a**). The currently considered intersection point is described by the coordinates x_n and y_n whereas the appropriate neighbour is defined with x_{n+1} and y_{n+1} , respectively.

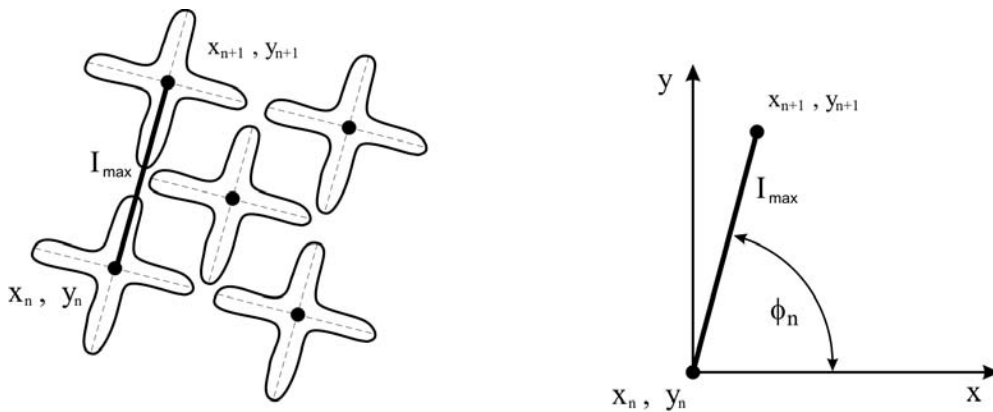


Figure 5-13.: Calculation of the angle based on the pair of intersection points with the maximum intensity.

In a cartesian coordinate system it is now very easy to get the angle ϕ_n which is defined by the pair of points with the highest mean intensity value I_n , as it is shown in **Figure 5-13 b**.

$$\phi_n = \arctan \left(\frac{y_{n+1} - y_n}{x_{n+1} - x_n} \right) \quad (5-10)$$

This so calculated angle ϕ_n corresponds to the orientation of the dendrite with the actual considered intersection point. Because of the axis-symmetric shape of a dendrite it is possible to define the orientation in a range from 0° to 90° . As the main result of the calculation part of the developed software a new matrix named *intersection_pts_phi* is created with the coordinates of the intersection points in the first two columns and the related angle for every intersection point in a third column.

$$\textit{intersection_pts_phi} = \left\{ \begin{array}{ccc} x_1 & y_1 & \phi_1 \\ x_2 & y_2 & \phi_2 \\ \vdots & \vdots & \vdots \\ x_n & y_n & \phi_n \end{array} \right\} \quad (5-11)$$

The source code for the calculation of the angle between two points producing the highest mean intensity value is shown in **Figure 5-14**.

```

[roIntPts, coIntPts] = size(intersection_pts);
[roNeighbours, coNeighbours] = size(neighbours);
S = size(intersection_pts);
intersection_pts_phi = [intersection_pts, zeros(S(1),1)];
%
for i = 1:roIntPts
    Intensities = zeros(roNeighbours,2);
    for j = 1:roNeighbours
        x = [intersection_pts(i,1), neighbours(j,1)];
        y = [intersection_pts(i,2), neighbours(j,2)];
        IP = improfile(IM, x, y);
        Intensities(j,1) = mean(IP);
        Intensities(j,2) = length(IP);
    end;
    %
    [maxVal, maxAt] = max(Intensities(:,1));
    %
    x1 = intersection_pts(i,1);
    x2 = neighbours(maxAt,1);
    y1 = intersection_pts(i,2);
    y2 = neighbours(maxAt,2);
    k = atan((y2-y1)/(x2-x1));
    intersection_pts_phi(i,3) = k;
end;

```

Figure 5-14.: Sourced code for calculating the angle between two points with the highest mean intensity value I_{max} .

A very effective method to reduce the processing time is to divide the image into smaller sub-images. An image with the size of 2300 x 1600 pxl is handled as a matrix with 3.680.000 entries, which is very time consuming. By dividing the image into square sub-images of the size 300 x 300, the examined matrix has only 90.000 pxl. The sub-images are extended with a boundary in order not to constrain the nearest neighbours for border near intersection points, as it is shown in **Figure 5-15**.

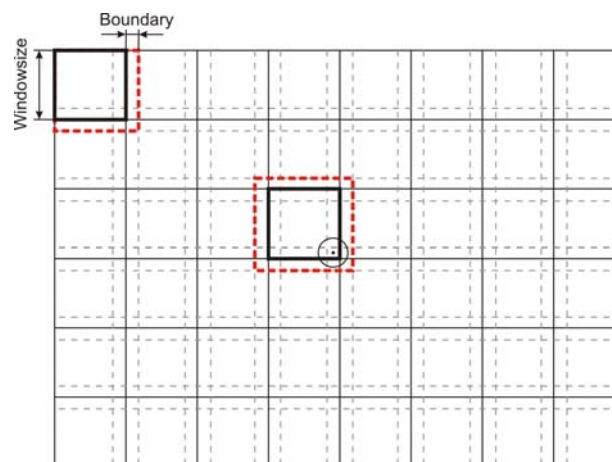


Figure 5-15.: Dividing the image in square tiles, extended by boundaries, in order to reduce the processing time.

Using this technique a reduction of processing costs of more than 90 % could be achieved (see Chapter 6.5).

5.7. Unifying the angles

In order to get a better recognizability of regions with equal orientations (i.e. primary grains) a fitting procedure is applied to the entries of the last column of the matrix *intersection_pts_phi* (i.e. the angles of the single dendrites). Therefore, every dendrite is compared with it's neighbouring dendrites orientation by the use of the nearest neighbours method. The benefit is that only the dendrites in a defined radius are considered in order to eliminate negative effects of dendrites with a related orientation in a farer distance (dendrites from other grains). The effect of this fitting is illustrated in **Figure 5-16**.

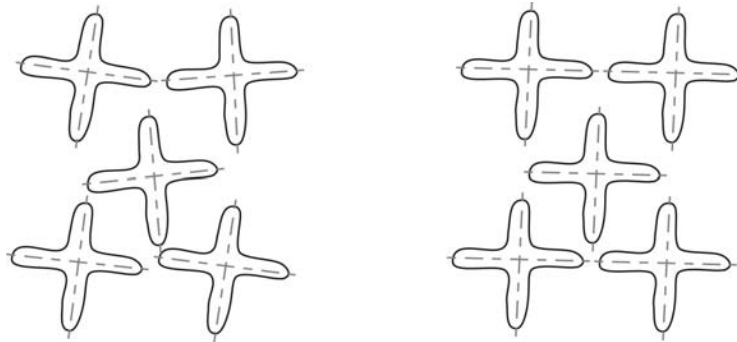


Figure 5-16.: Unifying a group of dendrites with a difference in their orientation smaller than a defined tolerance value ϕ_T .

If the difference of the orientation $\Delta\phi$ between the actually considered intersection point ϕ_a and it's neighbours ϕ_n is smaller than a defined tolerance value ϕ_T , the angle of these dendrites are set to the mean of the observed dendrites ϕ_m .

$$\phi = \begin{cases} \phi_m, & \text{for } \Delta\phi < \phi_T \\ \phi, & \text{for } \Delta\phi > \phi_T \end{cases} \quad (5-12)$$

with

$$\phi_m = \frac{\phi_n + \phi_a}{i + 1} \quad i = \text{length}(n) \quad (5-13)$$

and

$$\Delta\phi = \phi_a - \phi_n \quad (5-14)$$

At this point it must be mentioned, that big angles are similar to very small angles (e.g. 89° and 1°). In order to include these overlapping angles, a 90° -merge is computed. The unifying procedure is done two times with a slightly different method. The first run uses a diversion of the image in order to avoid a smoothly changeover of the orientation over the image. Therefore, the image is divided into a set of smaller subimages with a defined window size. The second run is done to fix the mean orientations from the first run, operated on the whole image. Both runs are using the nearest neighbour method.

5.8. Illustration

Every intersection point and its calculated angle is displayed as a cross of variable size with the original image in the background. For a better determination of the different angles, every cross is colored in a specific color. Every angle has its specific color assigned from the colormap "jet" (RGB), the code for the assignment is shown in **Figure 5-17**.

```
[rows, cols] = size(intersection_pts_phi);
%
for i = 1:rows
    phi = intersection_pts_phi(i,3)
    map = colormap(jet);
    val = abs(phi/(pi/2));
    num = size(map,1) - 1;
    val = round(val*num)+1;
    col = map(val, :);
    ...
    ...
    ...
end;
```

Figure 5-17.: Source code for coloring different angles using the colormap "jet".

A colormap is a m -by-3 matrix of real numbers between 0.0 and 1.0. Each row is an RGB vector that defines one color. The k^{th} row of the colormap defines the k^{th} color, where $\text{map}(k,:) = [r(k) \ g(k) \ b(k)]$ specifies the intensity of red, green and blue.

"Jet" ranges from blue to red and passes through the colors cyan, yellow and orange. It is a variation of the hsv colormap. The jet colormap is associated with an astrophysical fluid jet simulation from the National Center for Supercomputer Applications⁵². This colormap is very capable for this application because it has a wide spectrum of colors which can be easily

optically differentiated and does not show a continuous mild transition between two similar looking colors like e.g. autumn (**Figure 5-18**).

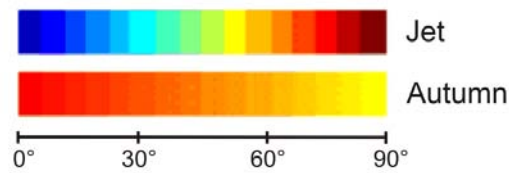


Figure 5-18.: Different colormaps.

5.9. Summary

Summarizing, the processing steps of the developed software can be described as follows. The original gray scale image, which is taken with a reflected-light microscope, is converted into a binary image by the use of a threshold operation. Using a global threshold for the entire image leads to a loss of information due to strong illumination gradients and non-uniform illumination. Instead of this, an adaptive threshold was chosen which changes the threshold dynamically over the image. The result of this step is a binary image in which the dendrites are the foreground objects (ones) and the rest is set to the background (zeros). After that, a smoothing operation is adopted in order to smooth the abrasive and pixelated shape of the foreground objects, which leads to better results in the following thinning operation. The concept behind this step is to reduce the shape of a dendrite in the binary image to a minimum width (one pixel) resulting in a skeleton representing the general form of the dendrite. The so created skeleton intersection point is used to describe the center of each dendrite with fixed coordinates in the image. In case of an unsymmetric shape of the dendrites, a merging procedure of the excess intersection points has to be done. On the basis of these intersection points, so-called intensity profiles are created between every intersection point (i.e. the center of a dendrite) and its nearest neighbors along the secondary dendrite arms. If the intensity profiles are on a high level it can be suggested, that the two considered dendrites have the same orientation. Every intersection point (i.e. dendrite) and its calculated angle is then plotted as a cross with the original image in the background. For a better determination of the different orientations, every cross is colored in a specific color. In order to get a better recognizability of regions with equal orientations (i.e. primary grains) a fitting procedure is applied on the cruxes, which aligns neighboring cruxes with just a slightly difference in their orientation to a uniform orientation.

6. Validation

In this chapter the validation of the developed software is described. In the first section, a comparison to a manual evaluation is given. The manual evaluation was done on a steel sample containing 0.6 wt.%C etched with picric acid. Then, the influence of the fitting operation for a conditioning of the colored cruxes is illustrated for a steel sample with 0.7 wt.%C etched with picric acid. In the next sections the influence of the main parameters "criterion" and the "Image for IP" are shown. In the last part, the processing time is shown for two different methods of calculation. The validation of the parameters in the *Results*-frame in the GUI will not be described because these settings are different for every image and have no specific values. Roughly reference values can be given as follows (6-1):

Table 6-1.: Reference values for the parameters in the *Results*-frame.

Parameter	1 st unifying the angles in tiles	2 nd unifying the angles
Radius	≤ 100	≤ 100
Tolerance for uniform angles	≤ 20	≤ 20
Tolerance for 90°-Merge	≤ 15	≤ 15
Windowsize	≤ 200	—

The output format should be approximately adjusted to the image size for the corresponding output image. Displaying the cruxes on the original image leads to a better identification of the primary grains and a checkup at the same time. A coloring of the plotted cruxes is also very helpful for the primary grain detection.

6.1. Comparison with manual evaluation

In order to compare the results of the software with a manual counting, a sample was evaluated by hand with respect to the primary grain size. This manual evaluation is a very time consuming, exhausting and subjective process. In order to arrange the manual counting in a passable space of time, the manual counting was done on an image of a steel sample with a carbon content of 0.6 wt.%C with clearly visible and observable dendrites (**Figure 6-1**).

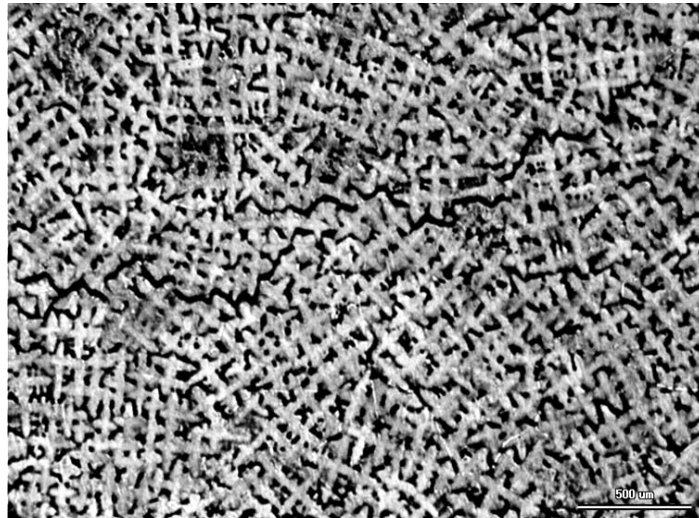


Figure 6-1.: Original image of a steel sample with 0.6 wt.% C.

On the first view on this image it seems to be quiet easy to detect the primary grains, but in detail it is much more difficult. After staring only a short period of time on the image, everything seems to look equal and a determination of the PGB without the plotted colored cruxes above the dendrites is very hard. The result of a manual evaluation is shown in **Figure 6-2**.

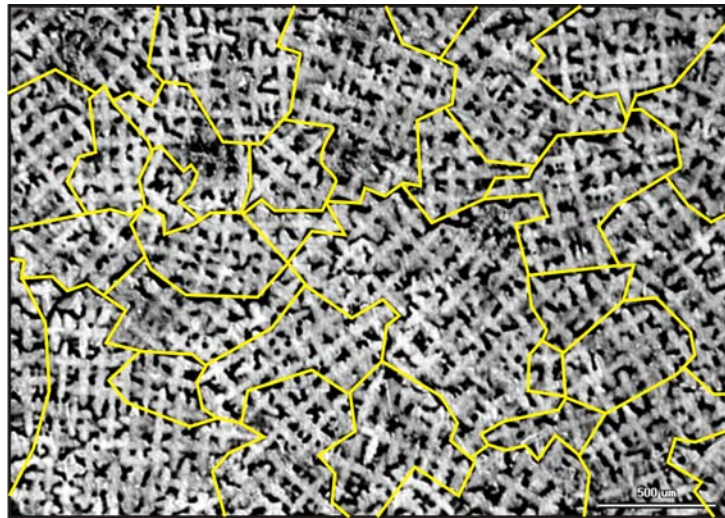


Figure 6-2.: Result of a manual evaluation.

Figure 6-3 shows the result of the developed software. Every dendrite is marked by a crux with its specific orientation. In order to get a better recognizability, every different orientation is different colored. Primary grains can therefore be judged as the group of cruxes with the same orientation and consequently by the same color.

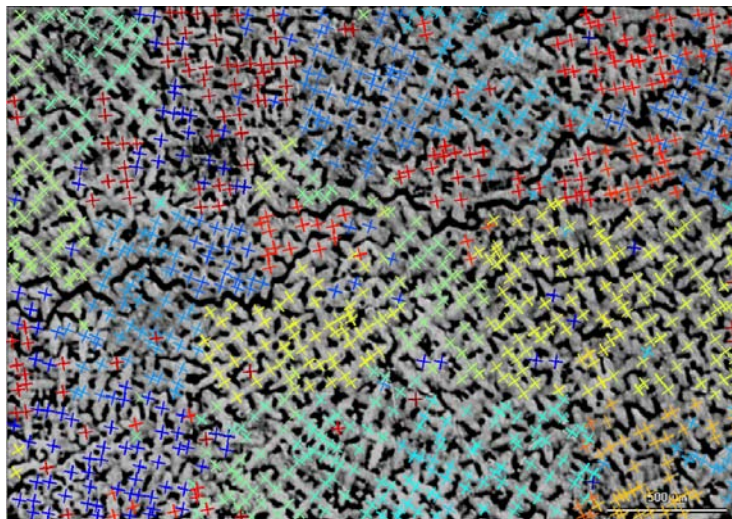


Figure 6-3.: Result of the software.

Figure 6-4 shows the comparison of the manual evaluation and the result of the developed software. In regions where a big difference of neighbouring grain orientations (i.e. dendrite orientations) occurs, the detection of the primary grain boundaries is very easy because of the different colors of the neighboring dendrites at the primary grain boundaries. This happens at

the upper portion of the image. When the difference of the orientations over a wide area does not change very much, the detection gets harder. This case occurs at the lower middle part of the image. The color changes slightly from yellow to green, so that an exactly identification of the primary grain boundaries, only based on the different colors, gets very hard. Therefore, a manual comparison to the original image has to be done. In this case, an indicator for the primary grain boundaries are those regions, where the primary dendrite arm spacing λ_1 increases, as it was published by J. Zhang and R.F. Singer³⁹. Another indicator could be, that at primary grain boundaries one of the four secondary dendrite arms is larger scaled than the others of the same dendrite.



Figure 6-4.: Comparison of the result of the software with the manual evaluation.

6.2. Effect of unifying the angles

In this section the influence of the fitting operation, described in Chapter 5.7, is shown. **Figure 6-5** shows the result of the pure calculation without the conditioning of the angles, executed on an image with 0.7 wt.%C but with a poorer quality of the observing dendrites as the sample in Chapter 6.1. However, the detected orientations of the dendrites are correct but differ about a few degree between the neighboring cruxes so that the cruxes are different colored. A grouping of cruxes with an uniform color is not observable.

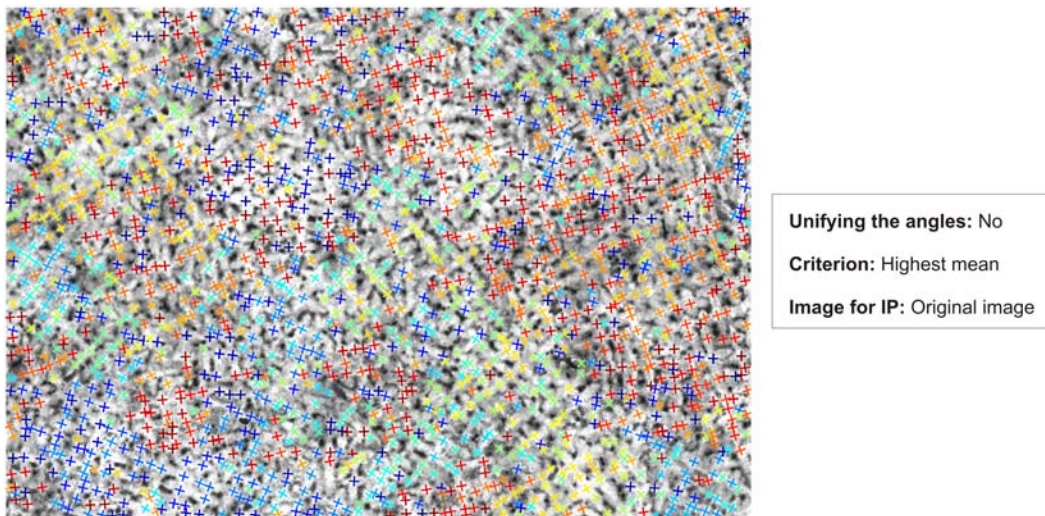


Figure 6-5.: Result of the pure calculation without the fitting operation.

The calculation was performed on the original image in tiles with the highest mean criterion for the following parameters (**Table 6-II**):

Table 6-II.: Chosen parameters for the main calculation.

Parameter	Value
Number of neighbors	30
Radius	100
Min. Distance	15
Window size	150
Boundary	50

The same calculation, where the unifying operation of the dendrites is applied, is shown in **Figure 6-6**. It can be seen, that those cruxes with a nearly equal orientation but different colors in **Figure 6-5** are grouped to a majority of cruxes with an uniform color. Although that there are still some artefacts, the detection of primary grains is much easier than on the image without this fitting operation.

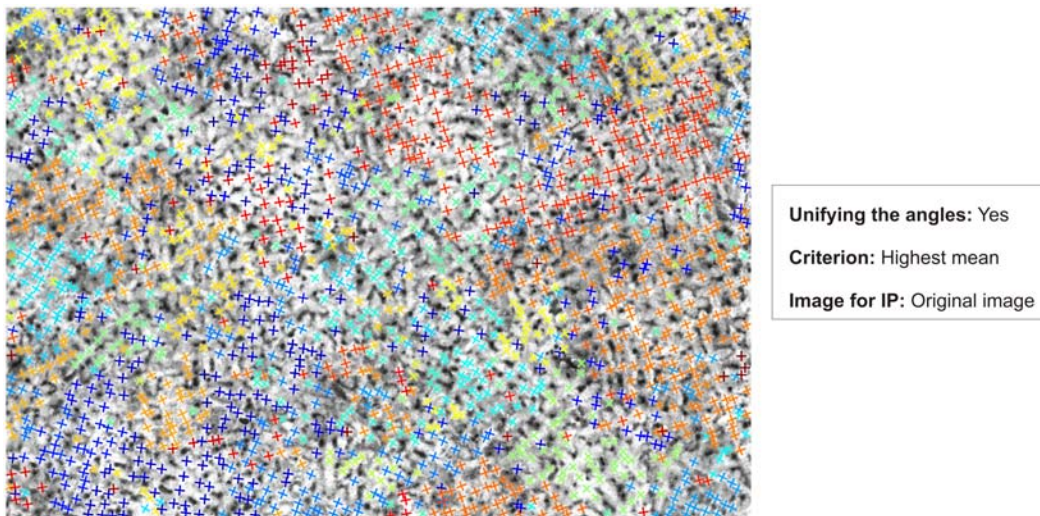


Figure 6-6.: Result of the calculation including the fitting operation.

The chosen parameters for the fitting operations are summarized in **Table 6-III**:

Table 6-III.: Chosen parameters for the applied fitting operation.

Parameter	1 st unifying the angles in tiles	2 nd unifying the angles
Radius	30	30
Tolerance for uniform angles	15	15
Tolerance for 90°-Merge	9	5
Window size	100	—

From the comparison of **Figure 6-5** and **Figure 6-6** it can be concluded, that a clearly detection of the primary grains without this fitting operation is not possible. The implementation of this operation in every calculation is therefore sensible.

6.3. Influence of the criterion for the choice of the intensity profile

The influence of the criterion for the choice of the best matching neighbor with respect to the determination of the orientation is explained in this section. As it is described in Chapter 5.6, there are two possibilities for the criterion. The result of the calculation with the same parame-

ters like in **Figure 6-6** (**Table 6-II**), except the criterion, is shown in **Figure 6-7**. The criterion in this calculation was set to "Ratio" and the image for the IP was the original image. The parameters for the unifying of the angles were also the same (**Table 6-III**).

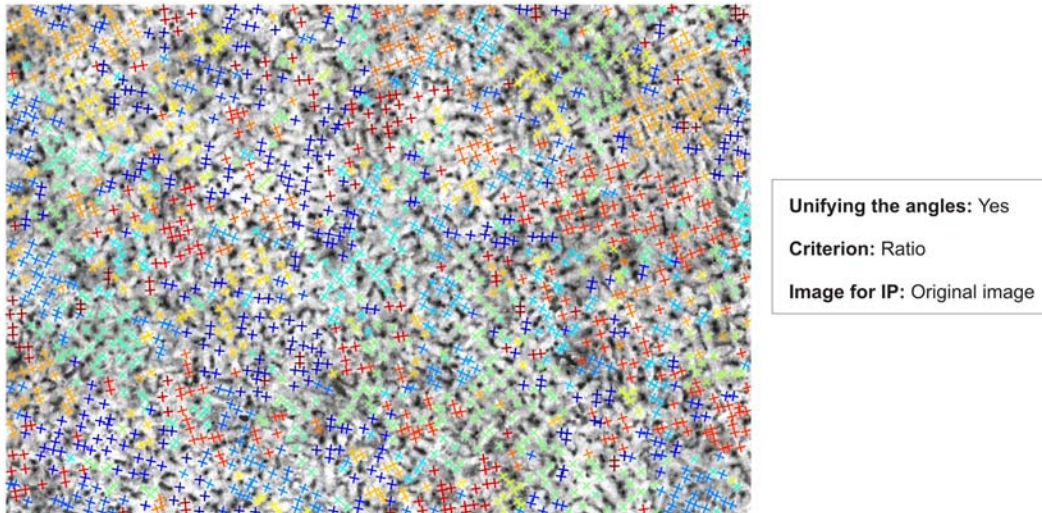


Figure 6-7.: Result of the calculation on the original image with the criterion "Ratio".

The same calculation, where the IP were calculated on the binary image, leads to the result in **Figure 6-8**. In comparison to **Figure 6-7** it can be seen, that there are no significant differences in order to get a better recognizability of the primary grains.

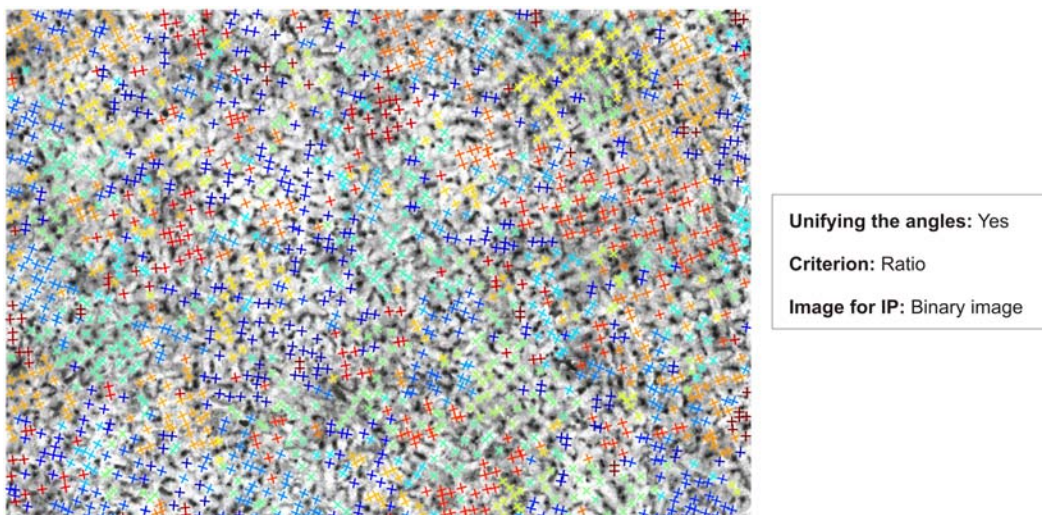


Figure 6-8.: Result of the calculation on the binary with the criterion "Ratio".

Summarizing, in comparison to **Figure 6-6** it can be concluded that for this image the "highest mean" criterion leads to the best results. Calculations on different images have also shown, that the "highest mean" criterion leads to better results than the criterion "Ratio" in most cases.

6.4. Influence of the image for intensity profiles

In this section the influence of the image, on which the intensity profile are taken, is shown. The result of the calculation with the same parameters, except the "Image for IP", like in Chapter 6.2 is shown in **Figure 6-9**. For this calculation, the intensity profiles were taken on the binary image. Due to the conclusions in Chapter 6.3 the criterion for the IP was set to "highest mean".

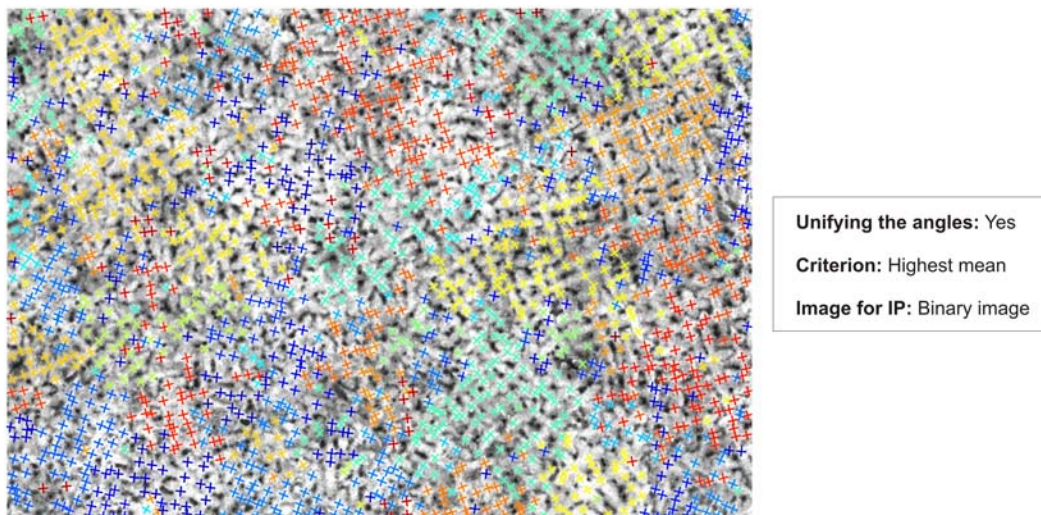


Figure 6-9.: Result of the calculation with the intensity profiles taken on the binary image.

It can be seen, that there are grains which are better detected using the binary image for calculation, but also grains which are better detected by the use of the original image for the calculation of the intensity profiles. However, this shows that a comparison of both methods is sensible in order to get the correct result.

6.5. Processing time

There are two different methods for the calculation of the image. Firstly, the calculation can be done by dividing the image into a set of smaller subimages, like it is explained in **Figure 5-15**.

The second method is to calculate the entire image at once, which is connected to a high amount of computational costs due to a much bigger matrix to process. A comparison of the two methods with respect to the processing time is given in **Figure 6-10**. It can be seen, that with a calculation executed in tiles a reduction of processing costs of more than 90 % can be achieved.

The image on which the calculations were operated is of the size 1021×701 pixel. The window size for the calculation in tiles was set to 150 and the boundary to 50 pixel.

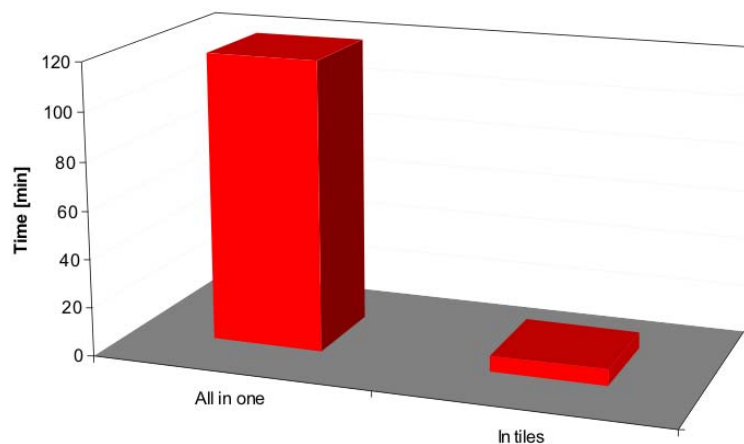


Figure 6-10.: Comparison of the computational costs for the calculation of the entire image and in tiles.

6.6. Summary and Conclusions

In this chapter, different adjustable parameters have been validated. The comparison to a manual evaluation showed, that in regions where a obvious difference of the orientation of neighboring dendrites occurs, the PGB can easily be detected based on the different color of the grouped cruxes. However, when the difference of the orientations of the cruxes over a wide area does not change very much, the detection gets harder. This is caused by the continuous mild transition between two similar looking colors. The comparison of the two criteria for the selection of the most suitable IP showed, that the "highest mean" criterion leads to the best result in most cases. The validation for this parameter was done for two different images for the calculation of the IP. Nevertheless, there may be some images where the second criterion generates the best result. From the investigations about the influence of the image for the

intensity profiles no clearly conclusion can be given. In order to detect the PGB as good as possible, a comparison of both results should be carried out. Lastly, the processing time for the calculation of the validated image was compared for the calculation of the image in tiles and the entire image. By executing the calculation in tiles a reduction of processing costs of more than 90 % could be achieved.

7. Summary, Conclusion and Outlook

Finally, a summary and a conclusion as well as an outlook are given in this chapter.

7.1. Summary and Conclusion

The presented thesis deals with the development of a digital image processing software to evaluate the solidification structure in steel. In a comprehensive review of the literature the development of the microstructure and the principle of image processing were presented. The main reason for the development of this software was the circumstance that existing machine vision based systems are just able to evaluate secondary and tertiary grains because of their visible boundaries, but no primary grains. Furthermore, the evaluation of the primary grain size by hand is a very time consuming and subjective process.

The basic steps of image processing can be summarized as follows: After an image acquisition, the original image is transformed into a binary image by the use of a threshold operation. Due to a non-uniform illumination and therefore a strong illumination gradient, the implementation of an adaptive threshold instead of a global threshold leads to better results. After a smoothing of the abrasive, pixelated shape of the foreground objects, a region-based morphological operation called "skeletonization" is adopted. The aim of the skeletonization is to extract a region-based shape feature representing the general form of an object. The so created skeleton intersection points can be used to locate every single dendrite in the image by the description of the centers of the latter. On the basis of these intersection points, intensity profiles are created between every intersection point (i.e. the center of a dendrite) and its near-

est neighbors along the secondary dendrite arms. If the intensity profiles are on a high level it can be suggested, that the two considered dendrites have the same orientation.

The validation with the results of a manual evaluation showed, that in regions where a big difference of neighbouring grain orientations (i.e. dendrite orientations) occurs, the detection of the primary grain boundaries is very easy because of the different colors of the neighboring dendrites at the primary grain boundaries. When the difference of the orientations over a wide area does not change very much, the detection gets harder. The color changes slightly between two colors, so that an exactly identification of the primary grain boundaries, only based on the different colors, is very hard. In this case, an indicator for the primary grain boundaries are those regions, where the primary dendrite arm spacing λ_1 increases. Another indicator could be, that at primary grain boundaries one of the four secondary dendrite arms is often larger scaled than the others of the same dendrite. The implementation of a fitting operation in order to get a better recognizability of regions with equal orientations (i.e. primary grains) was shown to be senseful. The validation with respect to the criterion to selection the most suitable intensity profile showed that in most cases the "highest mean" criterion leads to the best results. From the investigations about the influence of the image for the intensity profiles no clearly conclusion can be given. In order to detect the PGB as good as possible, a comparison of both results should be carried out.

Because of the fact, that the parameters for the presentation of the results vary for different images, the developed software is designed in two main parts: (1) the calculation part and (2) the results part. In order not to run the whole calculation every time, when the *Results*-parameters are set, the result of the calculation can be saved and loaded at any time.

7.2. Outlook

The conclusions and the results from the validation of this developed software provide promising perspectives for the adoption of this software to the evaluation of the primary grain size in steel. Nevertheless, a further validation with respect to the carbon content of the steel samples should be done in order to make a definition of the working range of the developed software. The code was generated in a manner that other add-on modules can be integrated in the software any time later. One of these modules could be the automatic detection and plotting of the primary grain boundaries based on the different orientations in regions of adjoining pri-

mary grains. Another extension could be an automatic evaluation of the primary dendrite arm spacing, λ_1 , based on the intersection points which are actually the centers of the dendrites. Therefore, a filtering operation in order to eliminate the excess intersection points has to be done previously.

Bibliography

- [1] Kurz, W. and D. J. Fisher: Fundamentals of solidification, Trans Tech Publications LTD, Uetikon-Zuerich (1984), Chapter 4.
- [2] Wang, N., S. Mokadem, M. Rappaz and W. Kurz: Acta Materialia, 52 (2004), 3173-3182.
- [3] Rappaz, M., A. Jacot and W. J. Boettinger: Metallurgical Materials Transactions B, 34 (2003), 467-479.
- [4] Campbell, J.: Castings, Butterworth-Heinemann LTD, Oxford (1991).
- [5] Rappaz, M., I. Farup and J.-M. Drezet: The Merton C. Flemings Symposium on Solidification and Materials Processing, Cambridge, USA (2001), 213-222.
- [6] Volmer, M. and A. Weber: Zeitschrift für Physikalische Chemie, 119 (1926), 277.
- [7] Frenkel, J. I.: Kinetische Theorie der Flüssigkeiten, Izd.-vo AN SSSR (1945).
- [8] Danilov, V. I.: Struktur und Kristallisation von Flüssigkeiten, Kiev, Izd.-vo AN USSR (1956).
- [9] Kuznecov, B. D.: Kristalle und Kristallisation, GITTL (1954).
- [10] Kolmogorov, A. N.: Zur Statistik der Kristallisation von Metallen, Izvestija AN SSSR (Serie Mathematik), 3 (1937).
- [11] Mirkin, I. L.: Analytische Untersuchung des Kristallisationsprozesses, Sammelband: Probleme der theoretischen Metallkunde, Berichte des MIS, 10, Moskau, Oborongiz (1938).
- [12] Grecnyj, J. V.: Zur Keimbildung in binären Legierungen, Sammelband: Kristallwachstum (Vorträge der 1. Tagung über das Kristallwachstum), Izd.-vo AN SSSR (1957).

- [13] Chalmers, B.: Kristallisationstheorie, Moskau, Metallurgija (1968).
- [14] Jackson, K. A.: Mechanismus des Kristallwachstums, Sammelband: Flüssige Metalle und ihre Erstarrung, Moskau, Metallurgizdat (1962).
- [15] Kahn, R.: Physikalische Metallkunde, Moskau, Mir (1968).
- [16] Alfincev, G. A. and D. E. Ovsienko: Zum Kristallisationsmechanismus, Sammelband: Wachstum und Fehlstellen metallischer Kristalle, Kiev, Naukova dumka (1966).
- [17] Sirota, N. N.: Inhalt und Probleme der Kristallisationstheorie, Sammelband: Kristallisation und Phasenübergang, Minsk, Izd.-vo AN BSSR (1962).
- [18] Winegard, W.: Einführung in die Physik kristallisierender Metalle, Moskau, Mir (1967).
- [19] Porter, D. A. and K. E. Easterling: Phase Transformations in Metals and Alloys, Van Nostrand Reinhold (International) Co. Ltd, England (1992).
- [20] Danilov, V. I.: Physikalische Grundfragen der Kristallisation, Stal, 10 (1955).
- [21] Ivancov, G. P.: Über die Möglichkeit des Kristallregens im Stahlblock, Stal, 10 (1952).
- [22] Saratovkin, D. D.: Dendritische Kristallisation, Moskau, Metallurgizdat (1957).
- [23] Walker, J. L.: Struktur von Blöcken und Gußstücken, Sammelband: Flüssige Metalle und ihre Erstarrung, Moskau, Metallurgizdat (1962).
- [24] Dobatkin, V. I.: Barren aus Aluminiumlegierungen, Moskau, Metallurgizdat (1960).
- [25] Gagen-Torn, V. O.: Kristallisation und Blockstruktur, Moskau, Metallurgizdat, 11 (1937).
- [26] Chworinov, N. I.: Kristallisation und Inhomogenitäten von Stahl, Moskau, Masgiz (1958).
- [27] Cernov, D. K.: Wissenschaft von Metallen, Sammelband: Werke D. K. Cernovs, Moskau, Metallurgizdat (1963).
- [28] Weinberg, F. and B. Chalmers: Canadian Journal of Physics, 30 (1952), 488.
- [29] Chalmers, B.: Transactions AIME, 519 (1954).
- [30] Chalmers, B.: Principles of solidification, John Wiles & Sons Inc., New York, London, Sydney (1964).
- [31] Mullins, W. W. and R. F. Sekerka: Journal of Applied Physics, 34 (1963), 323.
- [32] Walton, D. and B. Chalmers: Transactions AIME (1959), 215-447.

- [33] Pokorný, J.: De Ferri Metallographia III, Berger-Levrault, Paris & Nancy (1967), 28-31.
- [34] Rappaz, M., C. A. Gandin, J. L. Desbiolles and P. Thevoz: Metall Mater Trans, 27A (1996), 695.
- [35] Rappaz, M. and C. A. Gandin: Acta Metallurgica et Materialia, 42 (1994), 2233.
- [36] Zhou, Y. Z., A. Volek and N. R. Green: Mechanism of competitive grain growth in directional solidification of a nickel-base superalloy, Acta Materialia 56 (2008), 2631-2637.
- [37] Wagner, A., B. A. Shollock and M. McLean: Grain structure development in directional solidification of nickel-base superalloys, Materials Science and Engineering A, 374 (2004), 270-279.
- [38] D'Souza, N., P.A. Jennings, X.L. Yang, H.B. Dong, P.D. Lee and M. McLean: Seeding of Single-Crystal Superalloys - Role of Constitutional Undercooling and Primary Dendrite Orientation on Stray-Grain Nucleation and Growth, Metallurgical and Materials Transactions B, 36B (2005), 657-666.
- [39] Zhang, J. and R.F. Singer: Effect of Grain-Boundary Characteristics on Castability of Nickel-Base Superalloys, Metallurgical and Materials Transactions A, 35A (2004), 939-946.
- [40] Rappaz, M.: Modelling of microstructure formation in solidification processes, International Materials Reviews, 34 (1989), 3, 93-123.
- [41] Stefanescu, D. M.: Methodologies for modeling of solidification microstructure and their capabilities, ISIJ International, 35 (1995), 6, 637-650.
- [42] Boettinger, W., S. Coriell, A. Greer, A. Karma, et al.: Solidification Microstructures: Recent Developments, Future Directions, Acta Materialia, 48 (2000), 43-70.
- [43] Reiter, J., C. Bernhard, H. Presslinger: Austenite grain size in the continuous casting process: Metallographic methods and evaluation, Materials Characterization 59 (2008), 737-746.
- [44] Jähne, B.: Digitale Bildverarbeitung, 4. neubearbeitete Auflage, Springer-Verlag, Berlin und Heidelberg (1997).
- [45] Lange, S.: Verfolgung von farblich markierten Objekten, Seminarvortrag Computergrafik II, Universität Osnabrück (2003).

- [46] Kurzydowski, K. J. and B. Ralph: The quantitative description of the microstructure of materials, CRC Press Inc., Florida (1995).
- [47] <http://homepages.inf.ed.ac.uk/rbf/HIPR2/adpthrsh.htm>, viewed on April 6th, 2009.
- [48] Blum, H.: A transformation for extracting new descriptors of shape, Symposium on Models for the Perception of Speech and Visual Form, M.I.T Press, Cambridge, MA (1967), 362-380.
- [49] <http://www.inf.u-szeged.hu/palagyi/skel/skel.html>, viewed on April 6th, 2009.
- [50] W.B.Goh and K.Y.Chan: The multiresolution gradient vector field skeleton, Pattern Recognition, 40 (2007), 4, 1255-1269.
- [51] <http://www.mathworks.com/matlabcentral/forums/12574/1/content/demo/html/nndemo.html>, viewed on April 17th, 2009.
- [52] <http://www.mathworks.com>, viewed on April 17th, 2009.

A. Guidelines for use

This chapter is written as a instruction manual for the developed software. In the following sections every step is explained in detail. The preliminary work for the generation of the binary image consists of the binaryzation and the smoothing operation. After that, the usage of the main program is described.

A.1. Preliminary steps

The preliminary work for a segmentation of the original image are explained in this section. Based on the so created binary image, the actual calculation can be done in the main step.

A.1.1. Binaryzation

The porgram for this step can be found in the folder ". . . \02 Adaptive Threshold" and is started with *"RUN"*. Three parameters can be adjusted:

- The local window size *ws*,
- a Constante *C*, and
- the method *tm*.

The local window size "*ws*" is the size of the subimages, in which the whole image is divided. Larger windows yield the poorer result, because it is more adversely affected by the illumination gradient. It is also more computationally intensive than thresholding using the smaller window. Areas with a high range of intensity values within a local neighbourhood, i.e. the mean value lies

between the intensity values of foreground and background, can be separated easily by using the mean intensity. In regions, where the foreground and background are just slightly different, the mean of the local area is not suitable as a threshold. The situation can be improved if the threshold employed is not the mean, but mean- C , where " C " is a constant. Using this statistic, all pixels which exist in a uniform neighborhood are set to background. The method " tm " can be set as 0 or 1, which means a switch between mean and median, respectively. However, the mean leads to the better results in this application. The original image has to be copied in the folder ... \02 Adaptive Threshold\Input\ and is saved after the binarization in the folder ... \02 Adaptive Threshold\Output\ with the name of the original image extended with _ "bw" _ "ws" _ " C " _ " tm ". For example, if the name of the original image is "Sample1.tif", the output filename would be "Sample1_bw_21_0.05_1.tif" (executed with the parameters $ws = 21$, $C = 0.05$ and $tm = 1$).

A.1.2. Smoothing

The program for this step can be found in the folder "... \03 Smoothing" and is started with "*RUN*". For the use of the diamond-shaped structure element, the variable *StructElement* has to be set as 1, for the octagonal-shaped it must be changed to 2. Similar to the binarization, the original image has to be copied in the folder ... \03 Smoothing\Input\ and is saved after the smoothing operation in the folder ... \03 Smoothing\Output\ with the name of the original image extended with _ "smooth". For example, if the name of the original image is "Sample1_bw_21_0.05_1.tif", the output filename would be "Sample1_bw_21_0.05_1_smooth.tif".

A.2. Main calculation

A graphical user interface (GUI) has been generated to allow different users to execute the program without changing the source code. The GUI is shown in figure **Figure A-1**. The principle of the interface is split into two main parts. The *Calculation*-frame includes all parameters which are used in the calculation to generate the angles. The parameters in "*WindowSize*" define the portion of the image to be calculated. x_1 and x_2 sets the left and the right border of the subimage, y_1 and y_2 the top and the bottom. This is very useful to minimize the processing time in the

initial stage to vary the parameters to find the best values. If "All" is enabled, the whole image is read in for calculation.

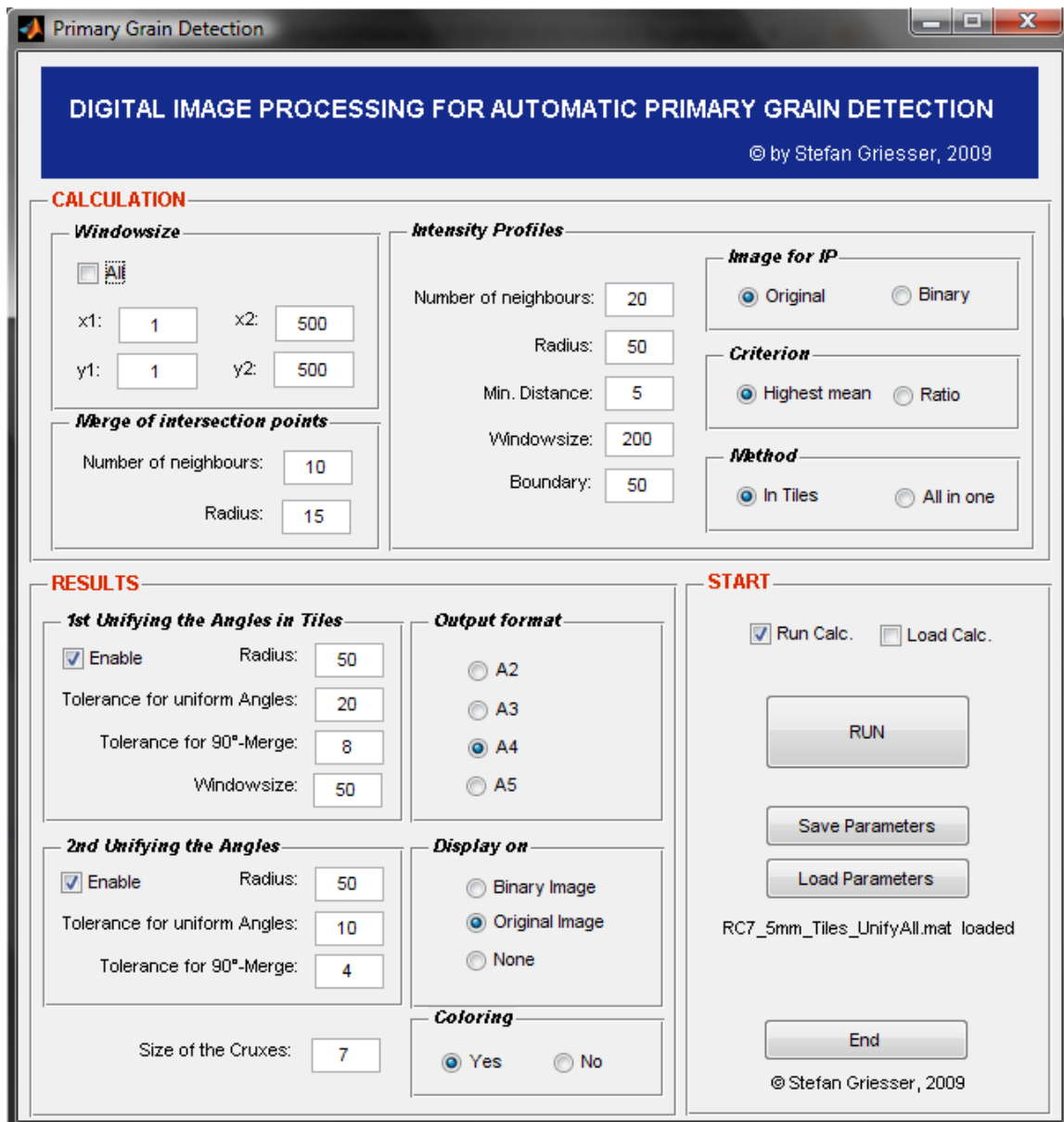


Figure A-1.: Graphical User Interface (GUI).

In "*Merge of intersection points*" the amount of the nearest neighbours in a defined Radius can be set up (see Chapter 5.4). In "*Intensity Profiles*" the parameters for the main calculation can be defined. It contains the number of neighbours and the radius to create the matrix *neighbours* (see Chapter 5.5), the minimum allowed distance "*Min. Distance*" between two intersection points and the window size and the boundary for the calculation of the image in tiles (see **Figure 5-15**). The latter is used if the "*Method*" is set as "*In Tiles*". Additionally the

calculation can be done "All in one" over the whole image. The intensity profiles can be taken from the original or the binary image, which can be set in "Image for IP". The "Criterion" for the choice of the best Intensity value, i.e. the proper neighbouring intersection point, can be set as the "Highest mean" or the "Ratio" (see Chapter 5.6).

The *Results*-frame includes all the parameters to set the size of the crosses, the output format, the coloring, the image background and the unifying procedure of the angles. As it is described in Chapter 5.7, the unifying procedure is done two times with slightly different methods. The "Tolerance for 90°-Merge" defines the value to be counted to the 90° in order to find similar small angles, which are in fact the same orientation (e.g. 89° and 1°).

By clicking the RUN-Button in the *Start*-frame the calculation starts by using the parameters as they are set. The Button "Save Parameters" saves all parameters in a *.mat-file in the folder ... \Save\Parameter\ in the current directory. By doing this it is possible to save the best parameters for a specific sample in a *.mat-file. Therefore, it is sensible to set the filename of the *.mat-file as the name of the sample. This parameters can then be loaded in with the "Load Parameters"-Button anytime later.

If a calculation is done and you only want to adjust the parameters in the *Results*-frame, the calculation (i.e. the created matrix *intersection_pts_phi*) can also be saved in a *.mat-file in the folder ... \Save\CalcData\ in the current directory. To start a new calculation, "Run Calc." must be enabled and "Load Calc." disabled, if the last calculation should be loaded both must be disabled and if a saved calculation should be loaded "Load Calc." must be enabled and "Run Calc." disabled.

The result of the calculation is saved as an *.tif-image in the folder ... \Output\. The filename is the actual date extended with the time when the calculation was started. For every output image also a LOG-File is created, in which all the chosen parameters are saved. An example of such a LOG-File can be seen in **Figure A-2**.

```
Digital Image Processing Software for Automatic Primary Grain Detection.
© Stefan Griesser, 2009

LOG-File from the last calculation started at:    19-Apr-2009 14:40:14

=====

Calculation included.

WINDOWSIZE:
  Calculated the whole image.

MERGE OF THE INTERSECTION POINTS:
  Number of neighbours:    10
  Radius:                  15

INTENSITY PROFILES:
  Number of neighbours:    20
  Radius:                  100
  Minimum Distance:        5
  IP_Image:                Original image
  Criterion:               Highest mean
  Method:                  In Tiles
  Window size (Tiles):     200
  Boundary:                40

1st UNIFORM ANGLES:
  Enabled:                 yes
  Radius:                  50
  Tolerance:               15
  Merge Tolerance:        3
  Window size:            200

2nd UNIFORM ANGLES:
  Enabled:                 yes
  Radius:                  50
  Tolerance:               10
  Merge Tolerance:        1

RESULTS:
  Scale:                   7
  Coloring:                Yes
  Output Figure:           Original Image
  Paper Size:              A3

=====

Calculation completed.

PROCESSING TIME:          5 min and 51.517 s.
File saved as:           Output/19-Apr-2009_14-40-14.png
```

Figure A-2.: LOG-File from a calculation.