Periodica Polytechnica Electrical Engineering and Computer Science

61(4), pp. 305-311, 2017 https://doi.org/10.3311/PPee.10890 Creative Commons Attribution ①

RESEARCH ARTICLE

GrainAutLine: an Environment for Semi-Automatic Processing of Marble Thin Section Images

Kristóf Csorba1*, Ádám Budai1, Judit Zöldföldi2, Balázs Székely3

Received 13 April 2017; accepted 07 June 2017

Abstract

GrainAutLine is an interdisciplinary microscopy image analysis tool with domain specific smart functions to partially automate the processing of marble thin section images. It allows the user to create a clean grain boundary image which is a starting point of several archaeometric and geologic analyses. The semi-automatic tools minimize the need for carefully drawing the grain boundaries manually, even in cases where twin crystals prohibit the use of classic edge detection based boundary detection. Due to the semi-automatic approach, the user has full control over the process and can modify the automatic results before finalizing a specific step. This approach guarantees high quality results both in cases where the process is easy to automate, and also if it needs more help from the user. This paper presents the basic operation of the system and details about the provided tools as a case study for an interdisciplinary, semi-automatic image processing application.

Keywords

marble, thin section, grain boundary, semi-automatic segmentation, software tool

¹Department of Automation and Applied Informatics, Faculty of Electrical Engineering and Informatics, Budapest University of Technology and Economics, H-1117 Budapest, P.O.B. 91, Hungary

²Institute of Material Science, University of Stuttgart, Stuttgart, Germany

³Department of Geophysics and Space Sciences, Eötvös Loránd University, Budapest, Hungary

*Corresponding author, e-mail: csorba.kristof@aut.bme.hu

1 Introduction

Beside classic image processing applications in computer science like medical image processing, motion tracking, and object recognition, there are many interdisciplinary areas where image processing can provide significant advantage. Due to the distance between these areas and information technology, the level of process automation and computer support can be still low. The GrainAutLine project introduced in this paper is developed to fill such a gap: it aims for semi-automatic image processing of marble thin sections for archaeometric and geomorphology applications. Marble provenancing is the identification of the original source of a marble sample which is used for both archaeology applications and for checking the genuinity of high value marble artefacts. Usually, the process is based on a classification using 3 features: two of them are isotope concentration ratios (C^{14} and O^{18}), and the third one is the Maximal Grain Size (MGS). The maximal grain size is the largest grain diameter in the sample which is easy to measure but it is not very stable in a statistical sense, due to its strong dependency on the given sample. Although there are definitions for much more reliable MGS-related measures, their penetration is limited because they need the measurement of many grains which is a very tedious task. If one can automate the measurement of grains in a thin section image, grain diameter histograms could be calculated leading to much better MGSlike measures. One of them is MGS99 defined as the 99% percentile of the grain diameter distribution [1].

The grain size histogram can be calculated based on a grain boundary image. In many cases, boundary images can be easily retrieved using edge detection methods typical in image processing. In case of marble thin sections, this is not the case. Crystal twinning is a grain evolution process where calcite crystals merge together into a single grain. This process leaves near-parallel lines inside the image as shown in Fig. 1. These twin crystal lines are very similar to the grain boundaries, but they need to be omitted during grain size measurements. This turns out to be a challenging image segmentation task.



Fig. 1 Marble thin section with twin crystals causing near-parallel lines inside the grains, making the image segmentation task harder. Sample from the MissMarble [2] dataset. (Contrast increased for better visualization.)

As the grain size histogram has to be accurate, a fully automated image segmentation process is not sufficiently reliable for the end users. In order to automate the process as much as possible, and still guarantee a high accuracy, GrainAutLine aims for a semi-automatic approach: it is a specialized drawing application supporting the user with a set of smart tools specialized for this domain specific image processing task. This way the manual work is significantly smaller than the fully manual drawing of the boundaries, and the high accuracy is still maintained as the user has full control over the segmentation process. This leads to the requirement of not only smart image processing tools, but also ergonomic user interface and a convenient visualization technique.

In this paper, we present the GrainAutLine project from the perspective of software engineering: its architecture, key elements of the user interface, and the provided smart image processing tools. Finally, the results retrieved using this application are introduced.

2 Related work

Using thin section images as the source of many material related measures has a long history in archaeometry [3]. The most comprehensive data set with grain sizes in this field is presented in [4] with more than 1300 samples. However, many authors defined the grain size in different ways. This is due to the fact that the Maximal Grain Size (MGS) is formally the maximum of a stochastic variable in the given sample which leads to significant uncertainties when comparing samples.

One of the suggested methods to overcome this limitation and to simplify the extraction of the values is to measure the grains along a traverse line [3] and take the maximal value or the mean of the 3 largest values from these measurements.

The need for a more standardized and statistically stable grain size measure is also highlighted by the comparison of the results presented by three different authors for the same samples. For example [4], [3] and [5] provide the MGS values between 0.2-4.5, 1.6-4, and 3.0-3.4 for the same sample.

The search for a more stable solution lead to many propositions [6], [7], [8]. One of the most promising ones is MGS99 [1] which is the 99% percentile of the grain diameter distribution. It is more stable because it is not related to the maximum in a sample, but its drawback is the need for the grain size histogram over the whole sample.

As mentioned before, marble thin section images contain lots of twin crystal lines inside the grains which makes the image segmentation a much harder task. Classic edge detection based approaches [9] and color based methods with special optical requirements [10] were tested, but a sufficiently accurate solution for generic cases is still a subject for research. Although the image processing community has provided many techniques like superpixels [11], active contours [12], region competition [13], and graph cuts [14], the presence of the twin crystals makes the marble thin sections a special case still waiting for proper solutions.

The GrainAutLine system presented here was developed to facilitate the image segmentation research specialized in marble thin sections. The evaluation results are based on the Miss-Marble dataset [2].

3 Core architecture

The core of GrainAutLine consists of the following:

- Processing State Descriptor (PSD): the data model containing the image itself and the current state of the segmentation
- Processors: various kinds of operations modifying the PSD
- The visualization of the PSD
- The user interface

The Processing State Descriptor contains the following (Fig. 2):

- The original image
- Set of blobs. Blobs are non-overlapping polygons, connected components of the segmentation. The goal is to have every grain in the original image covered by exactly one blob.
- Selection: a subset of the blobs, represented as a set of blob indices.
- Auxiliary (aux) drawing: an image where the user can draw anything.



Fig. 2 Parts of the processing state descriptor, the underlying data model of GrainAutLine.

The key idea is that the sole input and output of all operations is the PSD: beside configuration and mouse/keyboard events, they read everything from it and the result is written to the PSD as well. For example a simple segmentation task can be performed as follows:

- The user opens an image which is loaded into the PSDs original image.
- The user starts a standard image segmentation algorithm which draws its result into the aux image.
- The user may modify the aux image by means of drawing or erasing lines. (This is also done via processors.)
- The user instructs to use the current aux content as segmentation. A processor creates blobs in the PSD for every connected component in the aux image and clears the aux image.
- The user finds two blobs which belong to the same grain. A simple line is drawn on the aux image connecting these two blobs. The user instructs to merge the blobs which have intersection with the line in the aux image.
- The user finds a blob covering two grains because there was a short gap in the boundary between them. This blob is selected (its index is added to the selection in the PSD) and a processor is invoked to automatically cut the selected blob at the narrowest bottleneck.

After the initial segmentation, the user can perform simple corrections, merging and splitting operations. And most of these operations do not need precise drawing, only some guiding in the form of selections and rough mouse strokes.

The visualization in the user interface is based on the visualization of the components of the PSD, as that contains all input and output of the processing steps. A multi-layer approach is implemented where the user can configure the visibility and transparency of every layer. Mainly for additional debugging and R&D related inspection purposes, every processor can have additional layers where they can overlay whatever they want over the PSD.

The architecture of GrainAutLine was designed to have a plug-in style processor set working on the central data model, the processing state descriptor. This allows for high flexibility and easy experimenting suitable for the parallel work of several researchers. The additional visualization layers available for processors provide the means for high level diagnostic features which is essential in image processing, as traditional, break point based debugging of such functionalities is extremely inefficient.

GrainAutLine is implemented in C++14 using the OpenCV computer vision library and the Qt+QML framework, extended with GPU support for the rendering of the visualization layers.

4 User Interface

The user interface of GrainAutLine (Fig. 3) is focused on the visualization of the PSD and the configuration and activation of processors. Using the PSD as the unified input and output of processors has a significant advantage here: the user only needs 3 simple operations to edit the PSD: toggling the selection of blobs, drawing freehand line on the aux image, and erasing from the aux image. This is sufficient to provide the necessary input to all tools.

The most important task of the user interface is the proper visualization: editing an image segmentation requires the user to see the segmentation, the current changes, and the original image even at the same location. Furthermore, images may have very different contrast and colors. This makes the question of how to visualize all these things challenging. GrainAutLine solves this with the combination of two separate approaches: a wide range of visualization settings and a supplementary view with independent visualization settings.

The appearance of the PSD can be controlled in 3 independent ways: (1) all layers have a transparency. This is usually used to modify the transparency of the blobs to accommodate to the contrast of the underlying original image. (2) The blobs can have the same or different random colors. This proved to be especially useful when the user wants to find blobs accidentally covering multiple grains due to small discontinuities in their boundary. If the blobs have different color, these cases are easy to recognize. (3) The user can switch whether the current set of selected blobs is highlighted, or at the contrary, every unselected blob is highlighted.

We found a special combination of selection visualization and automatic selection very useful: if the only selected blob is always the one under the mouse cursor, and the selected blob is made transparent, the user can see the original image of the current grain, together with the blobs around it (Fig. 4). As the boundaries themselves are not part of any blob, they are visible around the neighbouring blobs. This visualization mode makes the check of the segmentation relative easy. The user can quickly move from blob to blob with the mouse without the need to even click on them, and one can easily recognize the improperly detected boundaries.

The other solution GrainAutLine provides for the simultaneous inspection of the blobs and the original image below it is the supplementary canvas. This is a separate, small image area showing the current neighbourhood of the mouse cursor. This



Fig. 3 The user interface of GrainAutLine. (1) basic tools, (2) processor selection, (3) processor settings, (4) main view, (5) supplementary view, (6) main view layer settings.



Fig. 4 Alternative selection visualization: only the blob under the mouse cursor is transparent.

image does not show any other layers, only the original image and the mouse location in its center marked with a red cross. Anytime the user needs to clearly see the original image, they can look at this supplementary canvas as well. This is especially useful if one needs to manually slice a blob into two pieces by drawing the missing part of the real boundary on the aux image. In this case, the blob may occlude the original image which may be disturbing even if the transparency setting of the blobs is relative high. Instead of changing the transparency all the time, the boundary can be drawn by looking at the supplementary canvas only, which proved to be very convenient.

5 Semi-automatic image processing tools

In this section, we present the details of the most important tools the GrainAutLine system provides for image segmentation and for the correction of possible segmentation mistakes.

5.1 Automatic segmentation tools

The work with GrainAutLine usually starts with a fully automatic segmentation which will be refined later by the user. The available methods are widely used image processing approaches: the Canny edge detector, the Maximally Stable Extremal Regions, and a combination of double thresholding and adaptive thresholding [15].

The Canny edge detector is a commonly used edge detector searching for significant brightness changes while trying to maintain the continuity of border lines. The corresponding processor takes the original image, converts it to grayscale, and applies the Canny edge detector implementation of the OpenCV library. The blobs are retrieved as the connected components of the image between the detected edges.

The Maximally Stable Extremal Regions (MSER) method is a technique searching for closed polygons instead of boundaries,

and the main characteristic of the detected polygons is that they have significant brightness changes everywhere around their boundaries. This means that MSER polygons do not have boundaries where the brightness changes slowly along a gradient. Applicability for marble thin section images depends on the contrast, but in most cases results are good in terms of the number of additional correction steps required.

The third solution is a combination of double and adaptive thresholding. In this approach, we utilize the fact that grain boundaries are much darker than the inner parts of the grains. Adaptive thresholding means that areas significantly darker than their neighbourhood are detected. This eliminates the need to define a threshold brightness which is very useful if there is a slow change in brightness caused by illumination inhomogeneities. Double thresholding means the application of two thresholds: a strict and a loose one. The result consists of the connected components which were marked by the loose threshold, but only if they have at least one pixel marked by the strict threshold as well. This means that boundaries do not need to be very dark everywhere, as that would cause many discontinuities, but they need to be sufficiently dark everywhere and have enough very dark pixels too. The combination of these two thresholding methods leads to a processor which has convenient parameters to fine-tune for the contrast and darkness values of the image. Extended with the manual correction functions, this became the mostly used segmentation method in GrainAutLine.

All these approaches work on brightness values, so that twin crystal lines usually remain in the segmentation. If these lines are brighter than the boundaries, the adaptive double thresholding may be configured to omit them, but usually the brightness difference is not enough to do this.

5.2 Filling holes

Based on the observation that calcite grains usually do not contain holes, the need for an operation eliminating the holes inside blobs is straightforward. This processor takes the selected blobs and fills all internal holes. In most cases, this operation is applied at the end of the segmentation corrections, because it eliminates also the fragments of scattered grain boundaries which could be useful during the manual corrections.

5.3 Merging blobs

Merging blobs which have been separated due to twin crystal lines is one of the most common operations used in Grain-AutLine. When applied, it takes every stroke (connected component) of the aux image drawn by the user and merges the blobs overlapping with that stroke.

The user only needs to draw a simple stroke over the blobs to be merged, which can be done very fast. Much faster than drawing an exact boundary. The operation needs to remove the remaining of the incorrectly detected boundaries between these blobs shown in green in Fig. 5. Otherwise the result would be mathematically connected, but fragments of the former boundaries would scatter the blob and make the retrieved statistics distorted. To avoid this, the operation applies image morphologic dilation to the merged blobs, and subtracts the neighbouring non-merged blobs from the result. This way, we get a smooth merge without the danger of merging further neighbouring blobs (Fig. 6).



Fig. 5 Merging blobs 1 and 2 by filling the green area. Points C and D cannot be connected directly like A and B, because the red area needs to be avoided so that blob 3 is not merged as well.



Fig. 6 Marking the blobs to merge with a single stroke and the result.

5.4 Automatic slicing

Grain boundaries may have brighter pixels which may cause gaps that connect two grains. In these cases, the automatic segmentation methods cover two grains with the same blob. Often, slicing these blobs only requires a line of a few pixels in the nearest passage inside the blob. The automatic slicing tool aims to find this passage and cut it automatically. This way, the user only has to select the affected blobs and apply the operation.

The automatic slicing works in three phases: detection of big interior areas, searching for the narrowest passages separating them, and marking the passages on the aux image. We have decided not to allow the operation to cut the blobs immediately as its result may contain mistakes, so it only draws a set of slicing lines on the aux image as a suggestion. This suggestion can be modified by the user if necessary, and applied with a simple operation which subtracts the contents of the aux image from the blobs and re-calculates the connected components.

The first step is to detect the big interior areas. It is based on the assumption that the real grains can be found inside the faulty blobs at locations which are far away from all boundaries of the blob. After applying a distance transform which calculates for every pixel the distance of the nearest boundary pixel, high peaks of the resulting distance map are considered grain center candidates, the middle points of the true grains.

The second step applies a max-flow-min-cut graph search locating the narrow passages. This is done by searching for disjunct paths between every pair of grain center candidates. If the maximal number of disjunct paths is lower than a threshold, a narrow passage has to be between the starting and ending points (Fig. 7). It should be noted that the passage may not be continuous, it may consist of multiple, even narrower passages.

In the third step, the narrow passages are located along the found paths (assuming these pixels to have only boundary neighbours or pixels belonging to a found path), and they are marked in the aux image as suggestion for slicing locations. If the user accepts this suggestion, it is used to separate the grains (Fig. 8).

5.5 Ismuth-aware erosion

Despite all automation effort, there are cases when the user has to draw the exact boundary between two grains. For example if a blob needs to be sliced along a curve which is too long to be detected by the automatic slicing tool. In this case, precise mouse movement would be required. To ease this requirement, GrainAutLine allows the user to draw a wider line which covers the boundary continuously but may contain other pixels as well.

Ismuth-aware image morphologic erosion is an erosion operation which makes the line narrower, but it does not allow it to be fragmented: if the removal of a pixel during the classic erosion operation would split the line into multiple connected components, that pixel is preserved (Fig. 9).



Fig. 7 The automatic cut operation locates narrow passages between possible grains.



Fig. 8 Automatic cutting of two grains having a small gap in their boundary.



Fig. 9 Ismuth-aware erosion allowing the user to draw thick lines and shrink them by taking the image into account.

In this way, drawing the boundary lines only needs a wider line drawn by the user on the aux image, and this operation to remove the unnecessary pixels by taking the original image brightness values into consideration.

6 Results and their application

The primary output of the GrainAutLine application is the final processing state descriptor. As many of our target users are familiar with geoinformatic systems (GIS) providing a huge amount of spatial data analysis tools, the grain boundaries can be exported into standard Shape Files supported by all GIS systems, or simply into a grain boundary image.

We started to develop this system to extract grain size histograms and other grain size related statistics to support archaeometric marble classification. Fig. 10 presents the final grain boundary image, and Fig. 11 shows the grain size histogram for the same sample. From the user experience point of view, using the application is significantly faster and less tedious than the manual approach aiming for the same level of accuracy. Based on our experiences, most of the processing time is taken by manual merge operations which require rough and thus quick mouse strokes only.



Fig. 10 Grain boundary image of the marble thin section in Fig. 1.



Fig. 11 Grain size histogram

The initial concept of the application was to provide a semiautomatic solution where the user has full control and supervision over the whole process. This way, the accuracy of the results cannot be measured in terms of the segmentation quality, as that will be up to the user. The key measure of usefulness is the amount of work automated by the system. And in this sense, the application performed well.

7 Conclusions

This paper presented the internal architecture and operation of the GrainAutLine application. Archaeometric applications like marble provenancing can significantly benefit from image processing technologies in terms of process automation. Interdisciplinary areas like this need software solutions which can utilize domain specific knowledge. These tools need to meet usability requirements of the end users, they need to provide proper visualization, and tools guided by a priori knowledge about the processed input data. Generic image processing and GIS systems can partially fulfil this requirement, but an application dedicated to the task can achieve much more.

GrainAutLine is designed specifically for the user supervised image processing of marble thin sections. Based on our experiences, the application can significantly reduce the time required for drawing a grain boundary image, which is then a starting point of many data analysis tasks. The acceleration of this process opens new opportunities for the analysis of massive amount of microscopy images which may be unsuitable with manual solutions. Based on our knowledge, currently there is no similar software solution available for marble thin section microscopy image analysis. GrainAutLine is a novel approach to semi-automatic image processing in this domain.

Many of our results aim for archaeometric domains, here we presented those ones relevant to the software engineering community as well. Although the application was developed for the automation of a very specific task, its tools can be applied in several other domains of material technology like sandstone and concrete structure analysis, which form subjects of further research.

Acknowledgements

The authors would like to thank Lilla Barancsuk, Ákos Szőts, and Anna Váry for their valuable research and software engineering work during the development of the GrainAutLine application.

This work is connected to the scientific program of the "Development of quality-oriented and harmonized R+D+I strategy and functional model at BME" project. This project is supported by the New Széchenyi Plan (Project ID: TÁMOP-4.2.1/B-09/1/KMR-2010-0002).

References

- Székely, B., Zöldföldi, J. "Fractal analysis and quantitative fabric analysis database of West Anatolian white marbles." In: ASMOSIA VII, pp. 719-734, 2009.
- [2] Zöldföldi, J., Hegedűs, P., Székely, B. "MissMarble, an Interdisciplinary Data Base of Marble for Archaeometric, Art History and Restoration Use." In: Turbanti-Memmi, I. (ed.) Proceedings of the 37th International Symposium on Archaeometry, Springer-Verlag, Berlin, Heidelberg, pp. 355-361, 2011.

https://doi.org/10.1007/978-3-642-14678-7_51

- [3] Cramer, T. "Multivariate Herkunftsanalyse von Marmor auf petrographischer und geochemischer Basis." (Multivariate provenancing of marbles on petrographic and geochemical basis.) PhD Thesis, Fakultät VI Bauingenieurwesen & Angewandte Geowissenschaften, Technischen Universität Berlin, 2004. (in German)
- [4] Attanasio, D., Brilli, M., Ogle, N. "The Isotopic Signature of Classical Marbles." L'Erma di Bretschneider, Roma, 2006.
- [5] Zöldföldi, J. "Petroarchaeological Study on the Provenance of White Marbles in West Anatolia." PhD Thesis, Eberhard Karls Universität Tübingen, 2011.
- [6] Green, W. A., Young, S. M. M., van der Merwe, N. J. Herrmann, J. J. "Source tracing marble: trace element analysis with inductively coupled plasma-mass spectrometry." In: ASMOSIA V, pp. 132-142, 2002.
- [7] Pentia, M., Herz, N., Turi, B. "Provenance determination of classical marbles: a statistical test based on ⁸⁷Sr/⁸⁶Sr, ¹⁸O/¹⁶O and ¹³C/¹²C isotopic ratios." In: Lazzarini, L. (ed.) *Interdisciplinary studies on ancient stones*. (pp. 219-226), Bottega D´Erasmo, Padova, 2003.
- [8] Oesterling, N., Heilbronner, R., Stünitz, H., Barnhoorn, A., Molli, G. "Strain dependent variation of microstructure and texture in naturally deformed Carrara marble." *Journal of Structural Geology*. 29(4), pp. 681-696, 2007.

https://doi.org/10.1016/j.jsg.2006.10.007

- Burger, W., Burge, M. J. "Principles of Digital Image Processing: Fundamental Techniques." Springer Verlag, London, 2009. https://doi.org/10.1007/978-1-84800-191-6
- [10] Choudhury, R. K., Meere, P. A., Mulchrone, K. F. "Automated grain boundary detection by CASRG." *Journal of Structural Geology*. 28(3), pp. 363-375, 2006. https://doi.org/10.1016/j.jsg.2005.12.010
- [11] Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S. "SLIC Superpixels Compared to State-of-the-Art Superpixel Methods." *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 34(11), pp. 2274-2282 2012. https://doi.org/10.1109/TPAMI.2012.120
- [12] Nguyen, T. N. A., Cai, J., Zhang, J., Zheng, J. "Robust Interactive Image Segmentation Using Convex Active Contours." *IEEE Transactions on Image Processing*. 21(8), pp. 3734-3743. 2012. https://doi.org/10.1109/TIP.2012.2191566
- Jungmann, M., Pape, H., Wißkirchen, P., Clauser, C., Berlage, T. "Segmentation of thin section images for grain size analysis using region growing competition and edge-weighted region merging." *Computers & Geosciences*. 72, pp. 33-48, 2014. https://doi.org/10.1016/j.cageo.2014.07.002
- [14] Zhou, H., Zheng, J., Wei, L. "Texture aware image segmentation using graph cuts and active contours." *Pattern Recognition*. 46(6), pp. 1719-1733. 2013.

https://doi.org/10.1016/j.patcog.2012.12.005

[15] Nixon, M. S., Aguado, A. S. "Feature Extraction and Image Processing for Computer Vision." 3rd Edition, Academic Press is an imprint of Elsevier, 2012.