Journal of Structural Geology 30 (2008) 431–443

JOURNAL OF STRUCTURAL GEOLOGY

www.elsevier.com/locate/jsg

GIS-based detection of grain boundaries

Yingkui Li [a], Charles M. Onasch [b,*], Yonggui Guo [b]

[a] Department of Geography, University of Missouri-Columbia, Columbia, MO 65211, USA
[b] Department of Geology, Bowling Green State University, Bowling Green, OH 43403, USA

Received 6 July 2007; received in revised form 17 December 2007; accepted 18 December 2007
Available online 27 December 2007

Abstract

The recognition of grain boundaries in deformed rocks from images of thin-sections or polished slabs is an essential step in describing and quantifying various fabric elements and strain. However, many of the methods in use today require labor-intensive manual digitization of grain boundary information. Here, we propose an automated, GIS-based method to detect grain boundaries and construct a grain boundary database in which the shape, orientation, and spatial distribution of grains can be quantified and analyzed in a reproducible manner. The proposed method includes a series of operations and functions to identify grain boundaries and construct the grain boundary database. These processes are integrated into a GIS model using ArcGIS ModelBuilder; thus, little or no operator intervention is required to perform the entire analysis. The method was evaluated using thin section images taken from three sandstone samples. The results indicate that the proposed method can correctly identify >70% of grains recognized manually without any intervention and is especially suitable for analyses where large numbers of grains are required.

© 2007 Elsevier Ltd. All rights reserved.

Keywords: Grain boundary detection; GIS; Thin section; Fabric analysis

1. Introduction

The recognition of grain boundaries is fundamental to any type of analysis where the shape, orientation, and spatial distribution of grains are critical. For example, strain determination methods, such as the Fry (Fry, 1979), enhanced normalized Fry (Erslev, 1988; Erslev and Ge, 1990), and R_f/ϕ (Ramsay, 1967; Lisle, 1985) methods, require the location of specific points along grain boundaries as input. At present, recognition of grain boundaries is most commonly done with thin section images collected with a polarizing microscope (Heilbronner and Pauli, 1993; Starkey and Samantaray, 1993; Ailleres et al., 1995; Lumbreras and Serrat, 1996; Goodchild and Fueten, 1998; Bartozzi et al., 2000; Heilbronner, 2000; Fueten and Goodchild, 2001; Choudhury et al., 2006). Typically, grain boundaries are extracted in two steps: recognition of the boundaries followed by digitization. In polarized light, different grains in a thin section show different interference colors as a function of their mineralogy and optic orientation. If sufficient contrast between adjacent grains exists, the boundaries between them can be easily recognized. If adjacent grains show similar interference colors, the boundaries will be difficult to recognize. In many cases, the contrast can be increased by rotating the thin section relative to the polarizer and analyzer. Thus, full recognition of grain boundaries can be facilitated by using multiple thin section images taken at different angles of rotation (Starkey and Samantaray, 1993; Heilbronner and Pauli, 1993; Goodchild and Fueten, 1998; Heilbronner, 2000; Fueten and Goodchild, 2001). Once the boundaries have been recognized, they must be digitized for additional analysis. Traditionally, manual digitization was used, but due to the labor intensive nature of this process, applications such as strain analysis may not have been used to their full advantage (Choudhury et al., 2006).

To reduce the amount of operator input required, a number of automated grain boundary detection methods have been

[*] Corresponding author.
E-mail address: conasch@bgnet.bgsu.edu (C.M. Onasch).

0191-8141/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jsg.2007.12.007

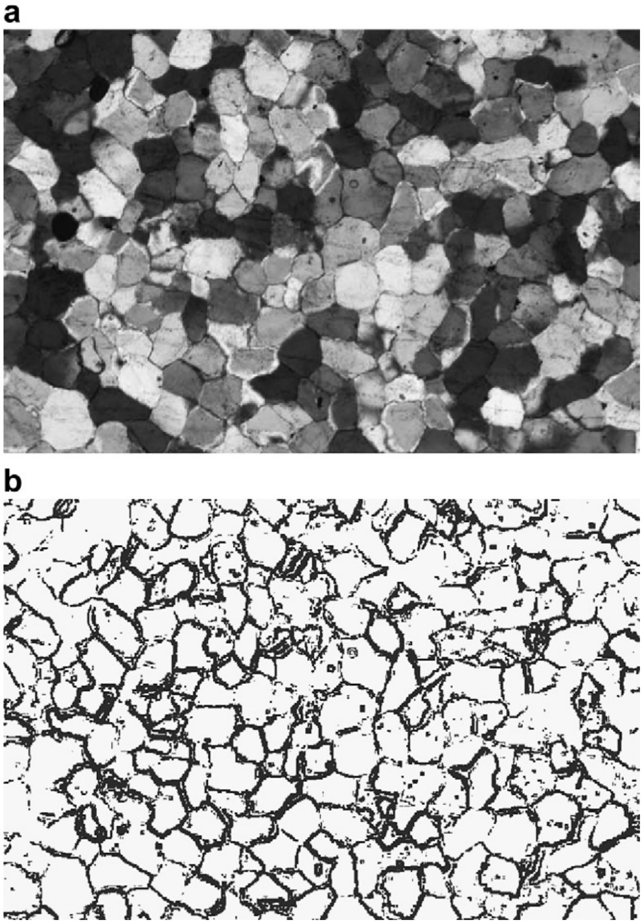

Fig. 1. Extraction of grain boundaries from a thin section photomicrograph. (a) Original thin section image taken with crossed polarizers and gypsum plate inserted. (b) Raster created by the Focal Range function in ArcGIS.

developed (e.g., Heilbronner, 2000; Bartozzi et al., 2000; Zhou et al., 2004; Choudhury et al., 2006), which can be applied to digital images of thin sections or polished slabs. These methods were developed to recognize grain boundaries through automated edge detection (e.g., Heilbronner, 2000; Bartozzi et al., 2000), region detection (e.g., Choudhury et al., 2006), or a combination of both (e.g., Zhou et al., 2004). In all these methods, various algorithms are applied to digital images, which may or may not have been first manipulated with image analysis software, to yield edges (grain boundaries) or regions (grains). While these approaches can greatly speed up the acquisition of grain boundary data and allow for many more samples to be analyzed, they are still limited in terms of their ability to analyze spatial information associated with the grains themselves. First, most of these methods recognize just the grain boundaries, which while very important, may not allow determination of gain shape information (e.g., area, long axis orientation). Second, as mentioned earlier, full recognition of grain boundaries usually requires multiple images. However, most current methods focus on just one image, which limits the amount of grain boundary information that can be extracted. Third, these methods do not allow other information, such as microstructure abundance and orientation, to be easily integrated with grain shape information.

In the last 15–20 years, geographic information systems (GIS) have been increasingly used to address a wide variety of geoscience problems (e.g., Bonham-Carter, 1994; Bishop and Shroder, 2004). The advantages of GIS include its database manner of data management, its capability to overlay and integrate different spatial information, and its powerful tools and functions for image analysis, spatial analysis, and modelling. Thus, it well suited for extracting, manipulating, and analyzing grain boundary information from deformed rocks (Fernandez et al., 2005).

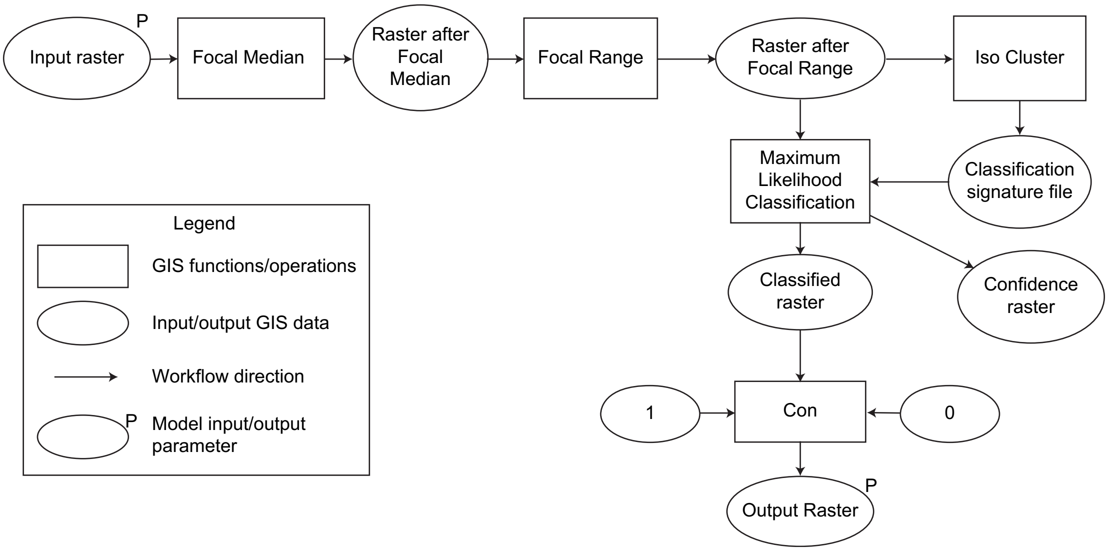

Fig. 2. Model designed using ArcGIS ModelBuilder to extract grain boundaries from a raster image. This model takes a raster as the input and outputs a new raster of 0/1 values representing boundaries (1) and interiors of grains (0). This model also includes a Focal Median operation, discussed in Section 2.3, to reduce noise and increase the contrast of the input raster. Descriptions of different ArcGIS functions used in this model can be found in the Appendix.

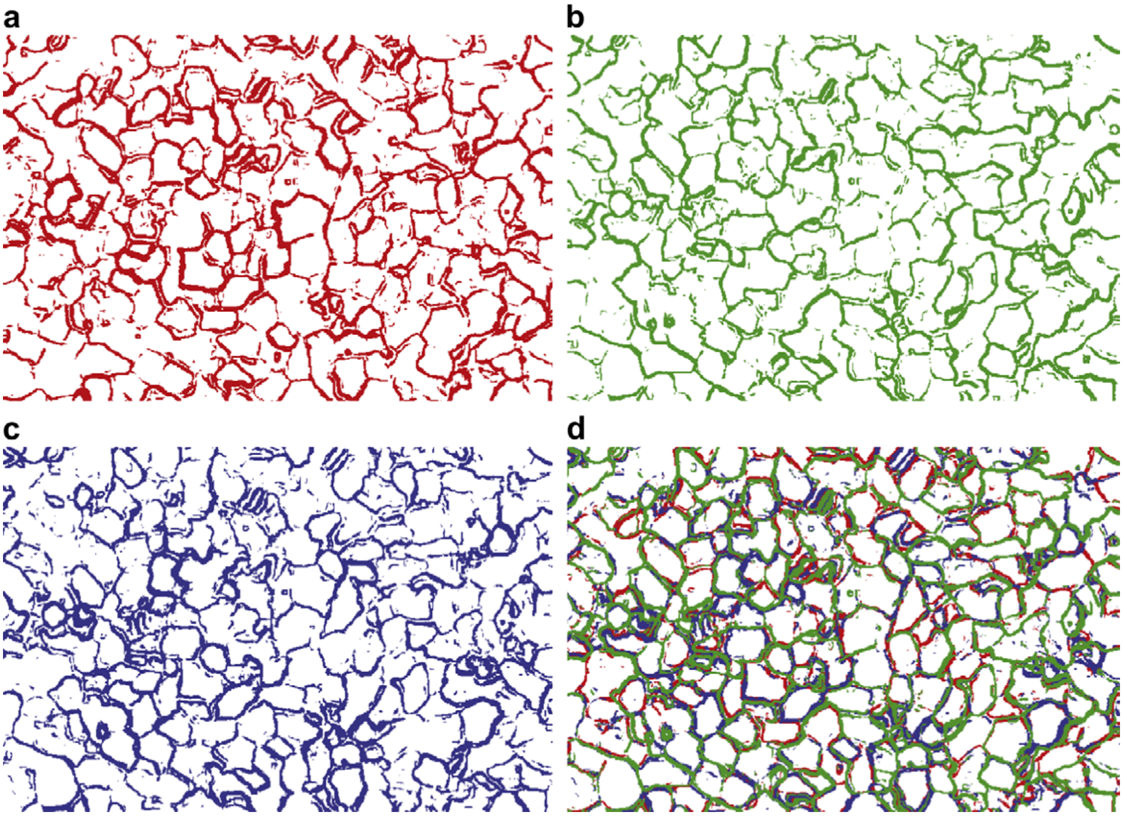

Fig. 3. Combining grain boundaries from the RGB bands of a single color image. Boundaries from red band (a), green band (b), blue band (c), and all three bands combined in a single image (d). Note how no one band contains all the grain boundary information.

Here, we present a GIS-based method to recognize grain boundaries and construct a grain boundary GIS database in which the occurrence and spatial distribution of any attribute (e.g., mineralogy, fabric, microstructure) can be quantified and analyzed in a reproducible way. Our approach requires little or no operator intervention, thereby further increasing its efficiency. This method will be illustrated using three sandstone samples, its accuracy quantitatively evaluated, and its applications discussed.

2. Methodology

As mentioned earlier, several algorithms have been used to detect grain boundaries from a thin section image. The most commonly used algorithm is edge detection (e.g., Heilbronner, 2000; Bartozzi et al., 2000), which detects grain boundaries as sharp changes of interference colors in the thin section image. Another approach is a seeded region growing (SRG) algorithm (Gonzalez and Wintz, 1987; Adams and Bischof, 1994; Zhou et al., 2004; Choudhury et al., 2006), in which the processes is started from a point (or seed) of interest and then extended to a region by adding points that are similar to the seed point. The GIS approach proposed here is based on the edge detection algorithm and includes a noise reduction algorithm to reduce noise introduced during edge detection. In our method, we use ArcGIS and refer to specific ArcGIS functions; however, most GIS packages contain similar functions and can be used to do the analysis. To aid users of other GIS packages, we define each ArcGIS function in the Appendix.

2.1. Edge detection

In a GIS environment, an image is generally represented in a raster format as an array of pixels where the value of each pixel indicates the color/brightness of that pixel. Grain boundaries are represented by abrupt changes of pixel values. If we calculate the range of pixel values (the difference between the maximum and minimum value) within a specified neighborhood (e.g., 3 × 3 window), large values will correspond to grain boundaries. This process can be accomplished in most GIS packages. Here we use the Focal Range function in ArcGIS, which is a type of Sobel edge detection algorithm that performs a 2-D spatial gradient measurement on an image to identify regions of high spatial gradient that correspond to edges. Focal Range function finds the range of pixel values within the specified neighborhood and assigns it to the center pixel in a new raster. In the new raster, small values indicate grain interiors and large values indicate grain boundaries. This raster can be reclassified to just grain boundaries by setting a threshold value or through an automated classification process in which 1 indicates a boundary and 0 a grain interior (Fig. 1). Thus, the grain boundaries can be identified. To minimize operator

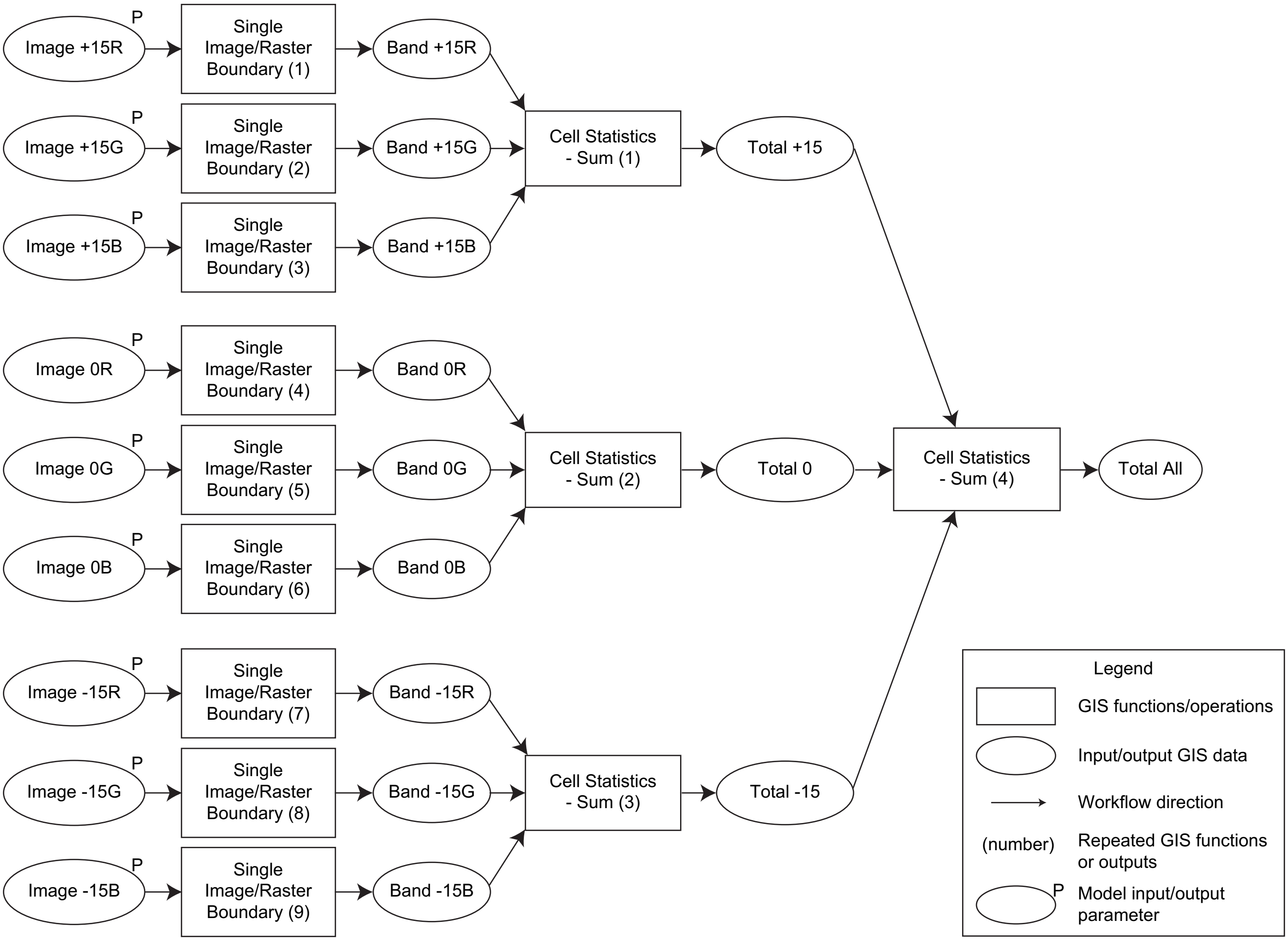
Image +15R
Image +15G
Image +15B
Image 0R
Image 0G
Image 0B
Image -15R
Image -15G
Image -15B
P
Single Image/Raster Boundary (1)
Single Image/Raster Boundary (2)
Single Image/Raster Boundary (3)
Single Image/Raster Boundary (4)
Single Image/Raster Boundary (5)
Single Image/Raster Boundary (6)
Single Image/Raster Boundary (7)
Single Image/Raster Boundary (8)
Single Image/Raster Boundary (9)
Band +15R
Band +15G
Band +15B
Band 0R
Band 0G
Band 0B
Band -15R
Band -15G
Band -15B
Cell Statistics - Sum (1)
Cell Statistics - Sum (2)
Cell Statistics - Sum (3)
Cell Statistics - Sum (4)
Total +15
Total 0
Total -15
Total All
Legend
GIS functions/operations
Input/output GIS data
Workflow direction
(number) Repeated GIS functions or outputs
Model input/output parameter

Fig. 4. A GIS model created using ArcGIS ModelBuilder to process three thin section images taken at three different polarizer/analyzer rotation angles (+15°, 0°, −15°), each of which is separated into three bands (RGB). The outputs of the nine individual edge detection processes (Fig. 2) are then combined into a single raster image. Descriptions of different functions used in this model can be found in the Appendix. This process is easily modified to use any number of original images.

interaction, the processes were automated using ArcGIS ModelBuilder (Fig. 2). So long as the illumination of each image is similar, a single threshold value can be used for all samples.

2.2. Overlaying multiple images

A color image will be transformed as three bands (Red, Green, and Blue) when imported into the ArcGIS environment. Each band has a range of values from 0 to 255. The edge detection processes described above can only be applied to one band at a time. Since each band represents only a portion of the information in the whole color image, the boundaries identified from an individual band can be somewhat different from other bands (Fig. 3). Thus, combining the boundaries identified from each of the three bands yields a composite with more information (i.e., boundaries) than in any single band. To accomplish this, each band was processed by the boundary detection method described above, then a new raster was created where each pixel is the sum of the values in each of the three bands. For each pixel in the new raster, a value of 0 indicates that the pixel is within a grain in all three bands, whereas a value of 1, 2, or 3 indicates the pixel is on a grain boundary in one or more of the band images.

In addition to combining information from the three bands of one thin section image, it was pointed out earlier that full recognition of grain boundaries can be enhanced by combining multiple images taken at different angles of stage rotation. There are two ways in obtaining multiple thin section images from different rotation angles. The first approach is to rotate the analyzer and polarizer together relative to the thin section on a fixed microscope stage (Fueten, 1997). In this way, each point is registered to the same pixel in the image at all positions of the polarizer/analyzer and the overlay of multiple thin section images can be performed directly. The second method is to rotate the thin section relative to fixed polarizer and analyzer. In order to perform the overlay of multiple images, certain types of image analysis algorithms need to be applied to rotate and register the images

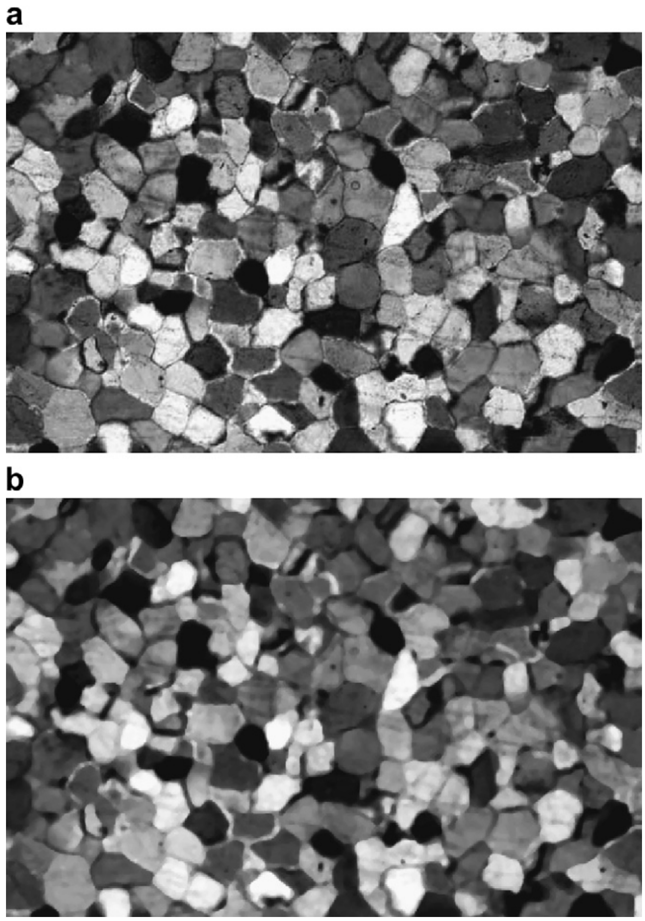

Fig. 5. Effect of applying the Focal Median filter in ArcGIS. (a) Original thin section image (red band). (b) Raster created after applying the median filter showing enhanced boundaries and reduced noise within the grains.

from the different rotation positions (Stöckhert and Duyster, 1999; Obara, 2007). We use the first approach to overlay multiple thin section images so that additional processing steps can be eliminated. The overlay of thin section images can be directly accomplished with the same approach as was used in combining the RGB bands of a single image (Fig. 4). In this way, the maximum amount of grain boundary information is merged into a single raster layer for further analysis.

2.3. Noise reduction

Within some grains, features such as inclusions, cracks, twins, or subgrains can cause abrupt changes in pixel values, thereby complicating the boundary detection process. In our approach, these features are treated as noise and two methods were developed to reduce it: one applied before the edge detection process and one after. The first applies a median filter (Focal Median in ArcGIS) to the input image to enhance the contrast between grains and reduce the noise inside gains (Fig. 2). The median filter searches the neighborhood around a selected pixel for pixels of similar brightness, discarding pixels that differ too much from adjacent pixels, and replaces the selected pixel with the median brightness value of the neighborhood. It does a better job than other filtering methods, such as linear smoothing, for our purpose since it more faithfully preserves the edges of objects in an image (Sangwine and Horne, 1998). Fig. 5 illustrates the effect of the median filter.

The second method of noise reduction utilizes characteristic differences in perimeter/area between grain boundaries and noise. Grain boundaries are usually linked together as interconnected networks, whereas noise appears as small isolated areas. Noise can be identified and removed by setting a threshold value on perimeter/area of connected pixels identified as boundaries in the output raster of the edge detection process. This process can be implemented in different GIS packages and requires different operations and strategies corresponding to the functionality of the package. In ArcGIS, this process can be automated through Modelbuilder using a series of functions including Region Group, Zonal Perimeter/Area, Iso Cluster, Maximum Likelihood and Classification (see Appendix for description of GIS functions). The Region Group function scans the whole raster from left to right and top to bottom to identify all connected regions and assign a unique number to each connected region. The Zonal Perimeter/Area function will calculate the perimeter/area for each connected region. Since the grain boundaries are connected together, they will form larger connected regions and have larger perimeter/area values as compared to noise, which will have smaller perimeter/area values. Linking the perimeter/area values to their corresponding pixels allows classification of two types of pixels: grain boundary pixels and noise pixels. The classification can be automated using Iso Cluster and Maximum Likelihood Classification functions in ArcGIS. Selecting too large a threshold value could result in the loss of some very small grains by classifying them as noise. After separating the grain boundary pixels and noise pixels, we can use the Thin function to reduce the number of pixels representing grain boundaries, so each boundary becomes a linear feature one pixel wide. Fig. 6 shows the GIS model for this process and Fig. 7 shows the effect of applying this algorithm to boundaries extracted from a thin section image.

Most operations discussed above could be done with image processing/analysis software. However, by performing them in the GIS environment we can streamline the process and eliminate the need to use multiple software packages. It also makes it easier to pass the data on to subsequent steps.

2.4. Building the grain boundary database

To obtain the shape and other spatial information about the grain boundaries, the raster format of the images must be converted to a vector format in the GIS. This can be done using raster/vector conversion functions in ArcGIS. In our approach, after the noise is removed, the grain boundary raster is converted to polyline (vector) features using the Raster to Polyline function. The Feature class to Coverage function

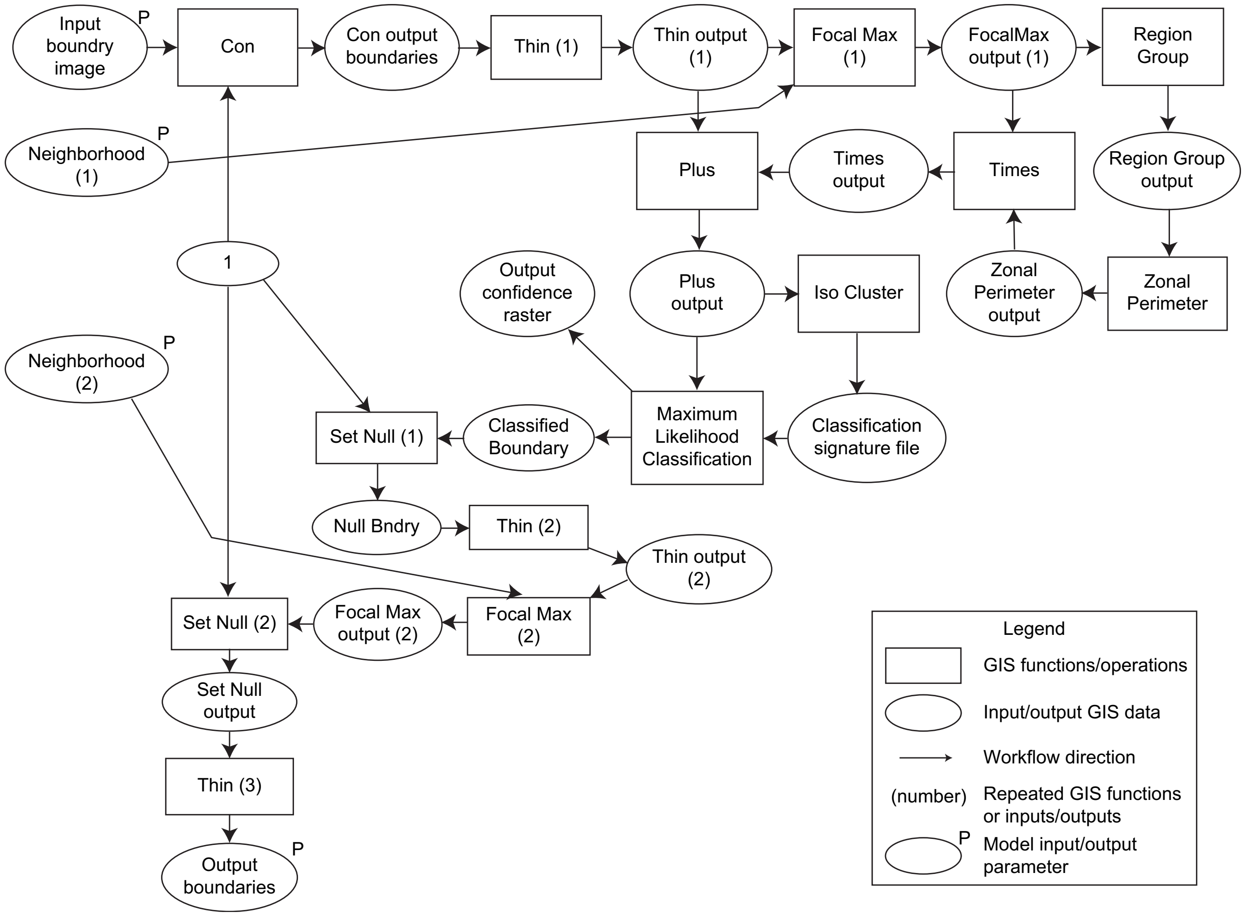

Fig. 6. A GIS model created using ArcGIS ModelBuilder to reduce noise after the edge detection processes. The input boundary image is the combined grain boundary image output from the model in Fig. 4. Descriptions of different functions used in this model can be found in the Appendix.

is then used to build the polygon features (grains) and maintain the topology of the polygons, which automatically generates the shape and other attributes of the grains. This process connects grain boundary segments into continuous boundaries. In closing the gaps in the boundaries, the snap tolerance, which is a threshold distance to snap nodes and vertices together, is the critical parameter and will affect the quality of the polygons, especially for small polygons. If the snap tolerance is set too small, true grain polygons, whose boundaries have gaps larger than the tolerance will not be generated; thus, some grains will be lost (Fig. 8a). If the value is too large, many polyline segments within this distance will snap together creating extraneous or distorted grain polygons (Fig. 8b).

Once the grain polygons are defined, the GIS can automatically add other attributes such as area, perimeter, centroid location, axial ratio, and long axis orientation. In addition, the user may then add other attributes, such as microstructure content, to each grain polygon. This is the primary advantage of using a GIS to identify the grain boundary as compared to other approaches (Fernandez et al., 2005). The GIS approach can not only be used to extract the grain boundary information, but also, and probably more importantly, it can be used to generate a grain GIS database in which the characteristics of grains can be automatically quantified and analyzed.

Each of the steps discussed, including pre-edge detection noise reduction, edge detection, multiple image overlay, post-edge detection noise reduction, raster to vector conversion, and building the grain GIS database, can be linked as a single model (Fig. 9) to automatically identify grain boundaries. Input for the model illustrated in Fig. 9 consists of three images of the same thin section taken at different polarizer/analyzer rotation angles (+15°, 0°, −15°) and output consists of the grain polygons and associated database.

3. Evaluation of model using natural examples

3.1. Sample selection and preparation

Three sandstone samples were chosen as representative of the range of difficulty that might be encountered in grain boundary recognition. Two samples (MS and WK) are quartz arenites, which is a lithology that is commonly used in previous studies (e.g., Choudhury et al., 2006; Heilbronner, 2000), thereby allowing comparison to other grain boundary detection methods. Of the two quartz arenites, the MS sample has more undulose extinction and some subgrain development,

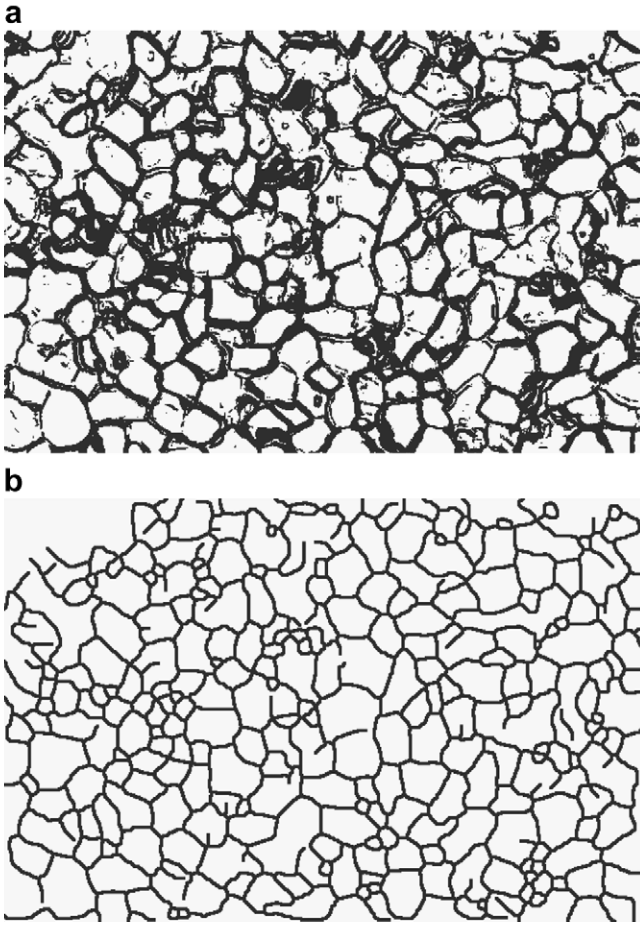

Fig. 7. Effect of applying the noise reduction algorithm shown in Fig. 6. (a) Input raster before the noise reduction. (b) Raster after noise reduction. Grain boundaries have been reduced to one pixel widths and noise within the grains has been greatly reduced.

both of which present challenges to grain boundary identification. The third sample (KF) is a hematite-cemented sandstone and should be among the easiest of lithologies for grain boundary recognition due to the strong contrast in optical properties between the opaque cement and transparent framework grains. Standard, 30 μm-thick thin sections of each sample were used.

For each sample, we start with digital images of the thin section viewed in cross-polarized light. Inserting the gypsum plate further increased the contrast. The images were captured with a digital camera on a petrographic microscope by rotating the analyzer and polarizer together relative to the sample on a fixed stage. This way, each pixel remains registered to the same point within the field of view during the rotation. Three images, with the polarizer and analyzer rotated to 0°, +15°, and −15° positions were taken of the same spot in the sample and saved in a JPEG format with a resolution of 400 × 600 pixels in 24 bit true color. The strategy is similar to that employed by several previous studies (e.g., Starkey and Samantaray, 1993; Heilbronner and Pauli, 1993; Goodchild and Fueten, 1998; Heilbronner, 2000; Fueten and Goodchild, 2001).

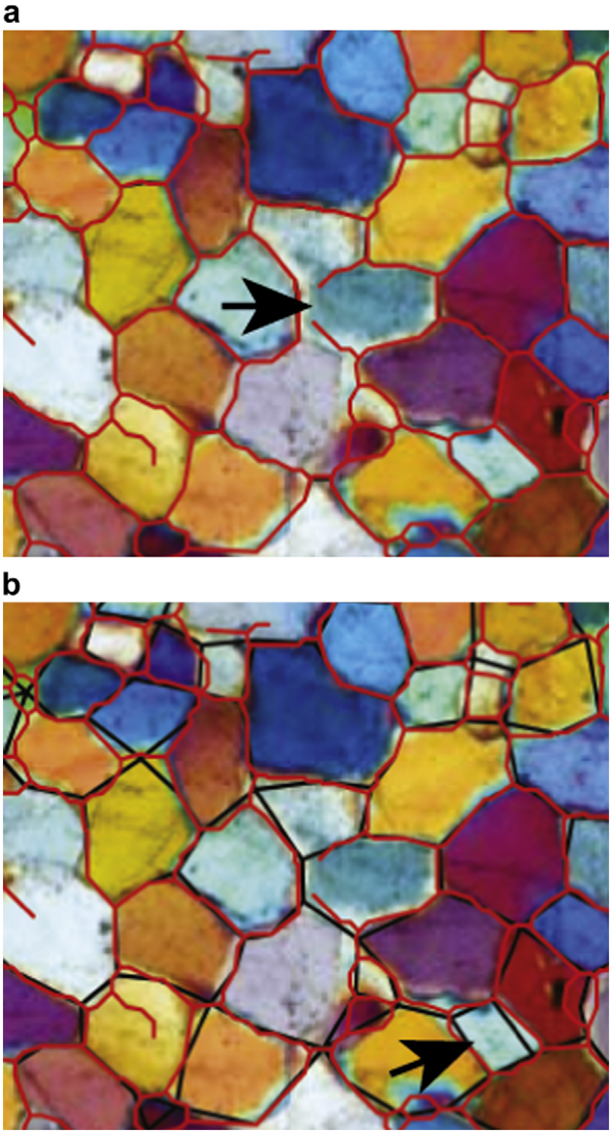

Fig. 8. Effect of different snap distances on the building of grain polygons. The snap distance for (a) is 2 pixels and for (b) is 5 pixels. The red lines are the grain boundaries created by the edge-detection routine and the black lines are the grain boundaries generated by the Feature class to Coverage function. A large snap distance (b) distorts the original grain boundaries and creates erroneous polygons (arrow), whereas a small snap distance (a) may fail to close the gap and generate a grain polygon (arrow).

3.2. Results and comparison with manual-digitized grain boundaries

Once the images were captured, the processing was done using the GIS model (Fig. 9). In order to objectively evaluate the model results, no human intervention (e.g., adjusting processing parameters) was used for individual images. Fig. 10 shows the grain polygons extracted by the GIS model superimposed on the petrographic image for each sample. Within the central portion of each sample, visual inspection indicates good agreement between the grain boundaries and those recognized by the GIS model. Along the edges, there is a strip approximately one average grain radius wide where

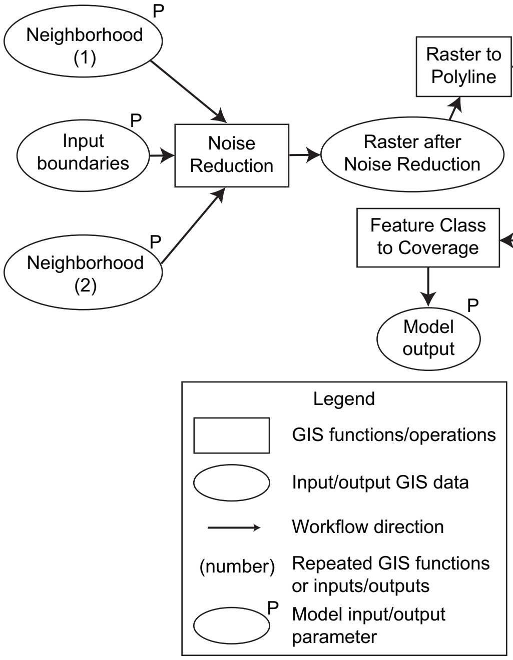

Fig. 9. The complete GIS model using three thin section images. This model is built upon sub-models shown in Figs. 2, 4, and 6.

grains were not recognized. This edge effect results from the way in which the model will not convert grain boundary segments to a polygon if they are not closed, which would be the case for grains at the edge of the image.

In each of the samples, the model failed to recognize some boundaries and drew some boundaries where none exist (Fig. 11). The most common cause for failure to recognize a boundary was lack of sufficient difference in interference color, contrast (Fig. 11a, middle arrow), or both (Fig. 11a, bottom arrow). Boundaries may also be missed when the ends of grain boundary segments from the superposition of the three images are farther apart than the snap tolerance used in linking segments into a complete boundary (Fig. 11a, top arrow). If the snap tolerance was increased, some of these open boundaries would be closed; however, more incorrect boundaries would also be drawn (Fig. 8b).

Extra boundaries can be drawn by the model when the distance between two boundaries is less than the snap tolerance set during the polyline-to-polygon conversion (Fig. 8b). In this case, small spurious polygons will be generated linking two close boundaries. Intragranular strain can also lead to extra boundaries and grains. The three incorrect grains indicated by arrows in Fig. 11b were drawn because the undulose extinction resulted in an interference color and/or contrast difference sufficiently large to trigger drawing a grain boundary. This was more common in the MS sample, which had undergone the most deformation.

The large areas of opaque cement in the KF sample caused some problems for the model, which treated large areas of cement as grains (Fig. 11c). Although the boundaries between cement and grains were correctly identified, most fabric analyses would not want to consider cement as grains. Most of cement areas can be identified and eliminated from the grain population by recognizing that cement areas tend to have very large perimeters relative to their areas. Within the GIS, it is a simple matter to select and eliminate grains whose perimeter/area exceeds a user-set threshold value.

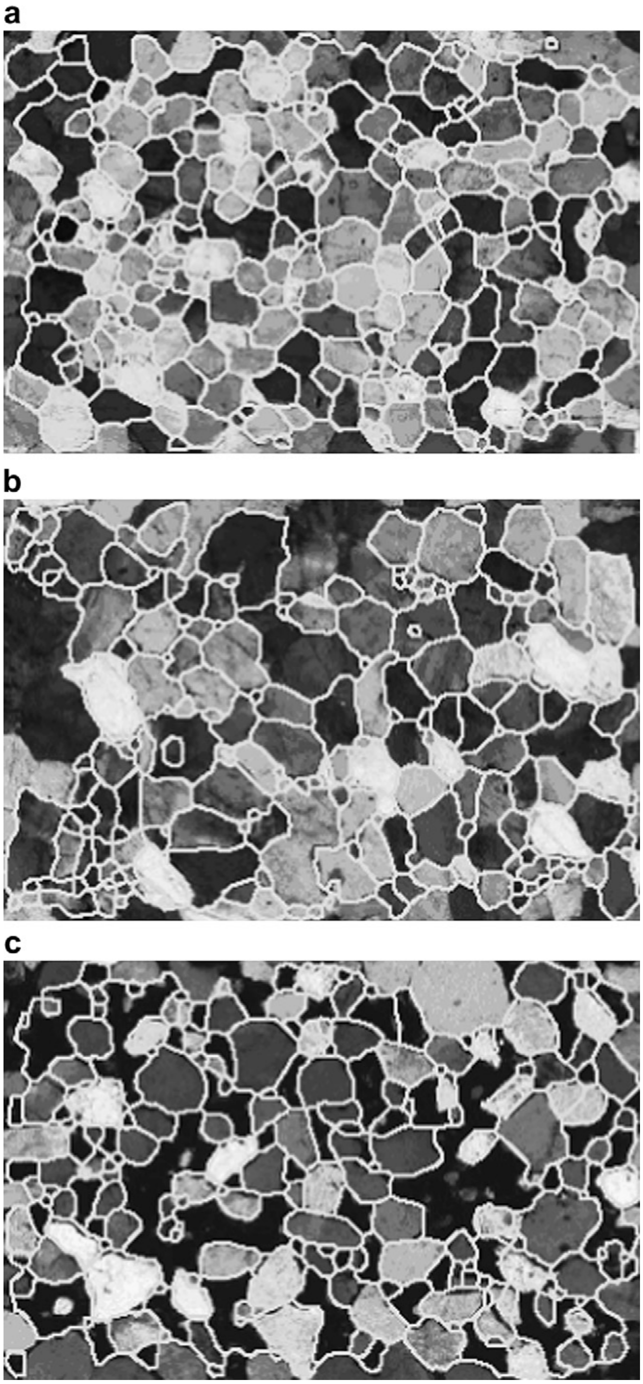

Fig. 10. Grains identified using GIS model (outlined in light gray) overlain on image of thin section for samples WK (a), MS (b), and KF (c). Thin section images were taken in polarized light with gypsum plate inserted. Width of field is 1 mm.

In order to quantitatively evaluate the ability of the GIS model to correctly identify and locate grain boundaries, we compared the grain polygons identified by the model to those

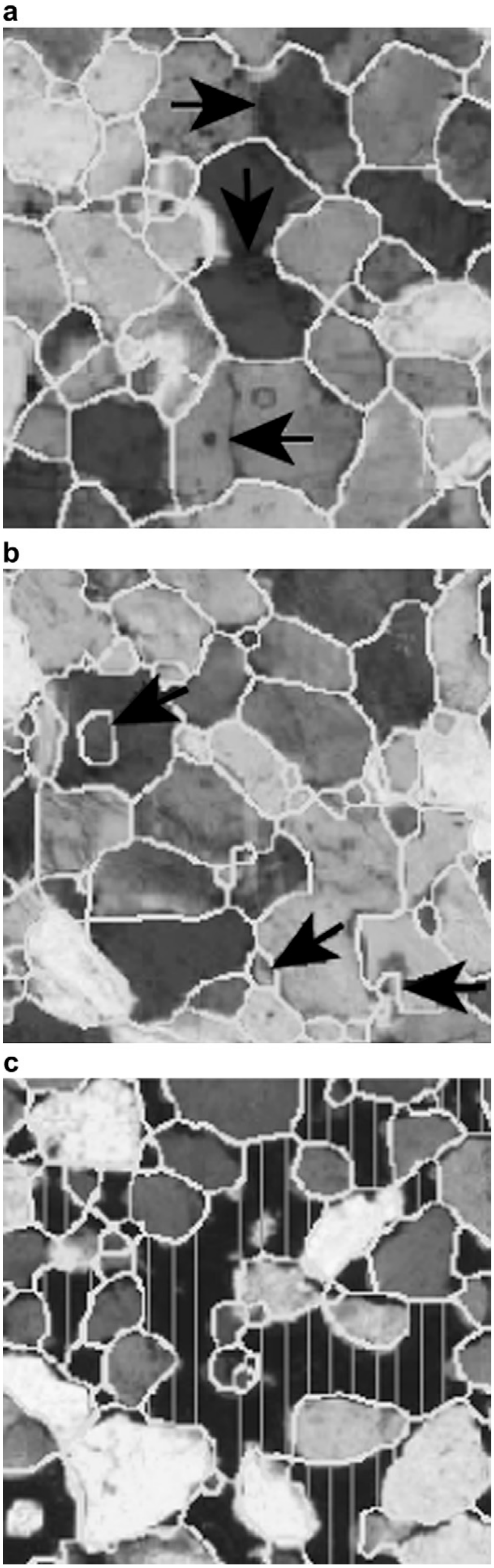

Fig. 11. Detailed comparisons of grains extracted by GIS model (outlined in light gray) and original thin section images showing problems that arose during processing: (a) missing boundaries (arrows), WK sample; (b) small (spurious) polygons (arrows), MS sample; and (c) cement polygons (ruled pattern), KF sample. Field of view is approximately 0.65 mm.

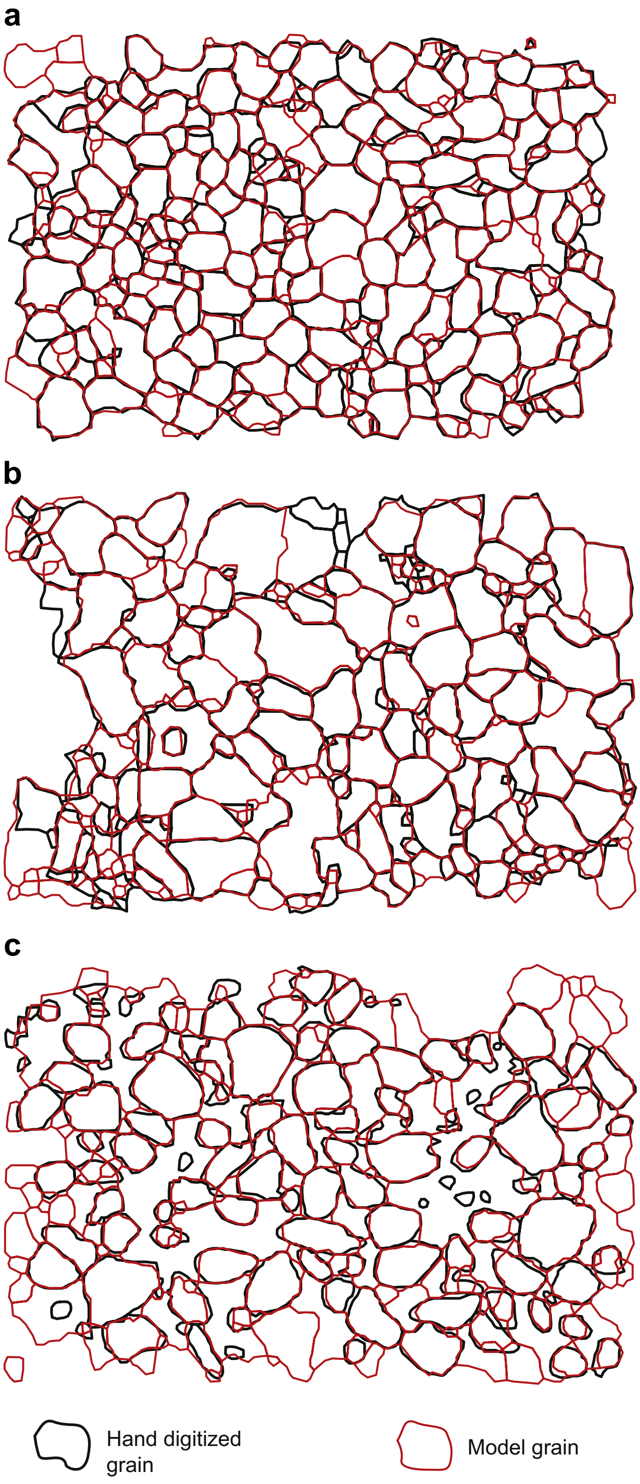

Fig. 12. Comparison of grains identified using GIS model to those digitized manually. For samples WK (a), MS (b), and KF (c). Width of each grain aggregate is approximately 1 mm.

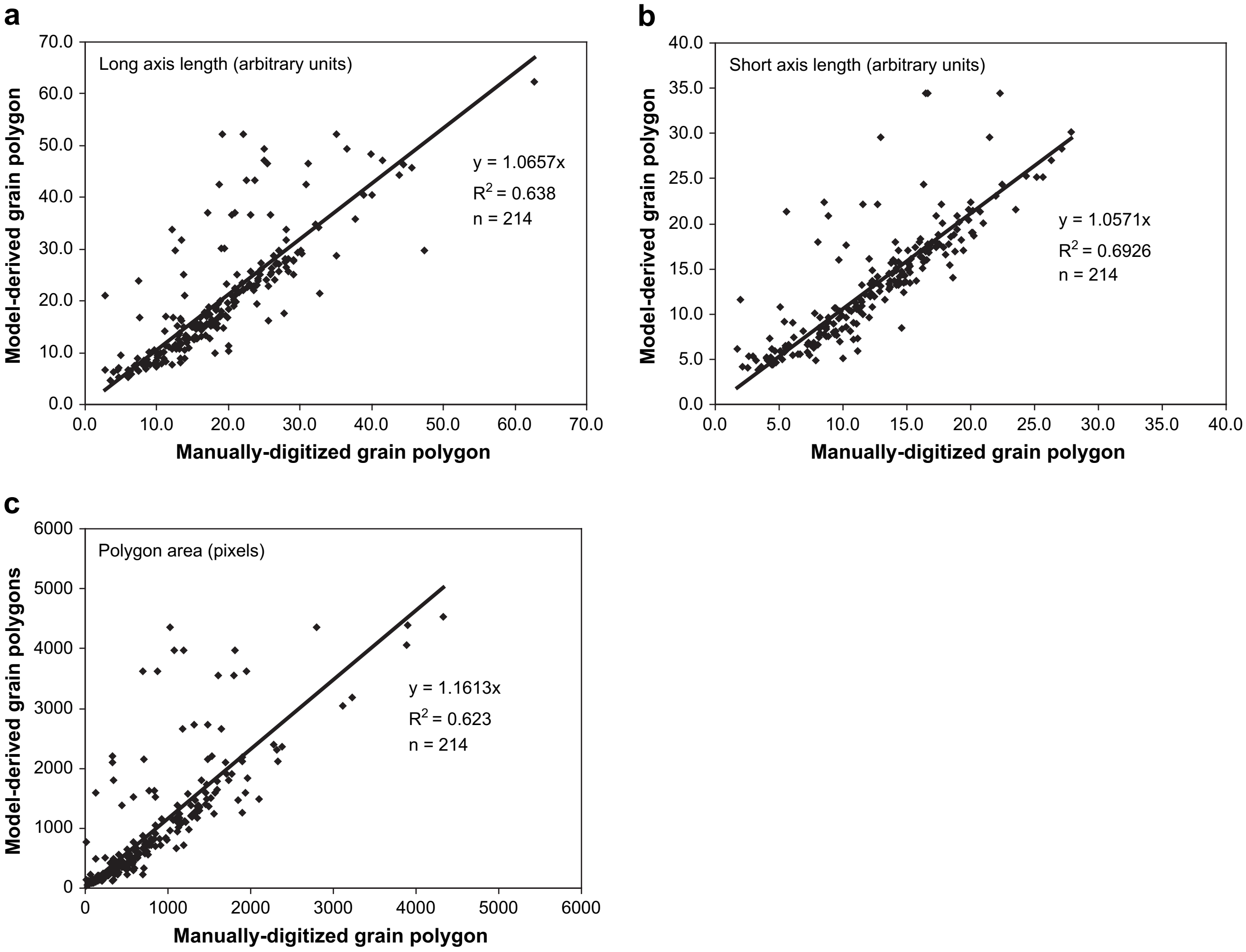

Fig. 13. Relationships between model-derived and manually-digitized grain polygons for long axis (a), short axis (b), and area (c) for WK sample.

digitized by hand (Fig. 12). Using a comparison of the locations, at least 70% of the hand-digitized grains were correctly identified by the model. To see how the geometry of the model-derived grains compared to the hand-digitized grains, we compared their areas, and the lengths of the long and short axes in the WK sample by linking these two grain features based on their locations (Fig. 13). All the parameters show a close relationship ($R^2 > 0.60$) between model-produced and manually-digitized polygons, which is statistically significant at the 95% level of confidence. The other two samples gave similar results. We did observe a number of points that fell above the regression lines in all three plots, which indicate that some model-produced grains polygons are larger than the manually-digitized polygons. This indicates that missing boundaries, which will lead to larger grains, are a bigger problem than extra boundaries, which would produce smaller grains.

4. Discussion

Of the methods developed to recognize grain boundaries, using a GIS offers several advantages. First, it is an automated process with little or no operator intervention. Once the model and parameters were set up, it only requires several minutes (depending on the type and memory of the computer used) to perform the whole analysis. Using the GIS model built by ModelBuilder allows users to visualize and understand the whole analytical process better than with other grain boundary recognition methods. This visual process can also provide opportunities for users to adjust the model parameters, adding human interventions to the process in order to increase the accuracy of the model output. Second, the GIS method not only can extract the grain boundaries, but also create grain polygons in GIS and build a GIS grain database, which has attributes such as position, area, perimeter, long axis length, short axis length, and long axis orientation, automatically. This is difficult for other methods using image processing software, which lacks these capabilities. Third, a GIS is a powerful tool for spatial analysis and modeling. For example, strain analysis (Fernandez et al., 2005), statistical analyses such as Kriging (Guo and Onasch, 2001), and deforming/retrodeforming modeling are easily accomplished in a GIS. Finally, GIS software, such as ArcGIS, is widely available and models such as the one presented here are easily customized for individual use.

4.1. Accuracy versus efficiency

No grain or grain boundary recognition method developed to date is 100% accurate in all rock types. The GIS method proposed here yielded grain boundaries in quartz arenites that matched hand-digitized boundaries with accuracies greater than 70% without any human intervention. With more operator intervention, the method could have yielded greater accuracy, but at the cost of being less efficient. However, even with 70% accuracy, the population statistics of model-derived and manually-digitized grains are comparable (Fig. 13).

Our GIS method has a number of operator-specified settings/parameters. These can be optimized for a given suite of similar samples to yield consistent results between samples. For example, two problems encountered in the grain boundary processing are large grains resulting from missing boundaries and small grains resulting from extra boundaries. Both could be reduced by adjusting various parameters such as image resolution, median filter values, and snap tolerances. For example, in a deformed sample with undulose extinction and subgrains, a lower image resolution, stronger median filter setting, and/or larger dangle length tolerance for snapping boundary segments would emphasize detrital grains at the expense of subgrains. Once suitable settings were found, the entire suite of samples could be processed with no further intervention. The most labor-intensive intervention would be to edit the model results from individual samples by adding or deleting boundaries or spurious polygons; however, this would greatly reduce the number of samples that could be processed, thereby partly defeating the advantages of our approach.

5. Conclusions

The GIS-based grain detection method presented here is able to accurately recognize grain boundaries without operator intervention. The real power of the method is its ability to construct a GIS database that contains information about the shape and location of grains, which can then be used to perform analyses of shape and location (e.g., strain) and model geologic processes in an objective and reproducible way. When optimized for a given set of samples, the automated nature of the method allows for the processing of large numbers of samples thereby opening opportunities to address new questions.

Acknowledgements

Funding for this project was provided by the National Science Foundation (EAR-0087607). We thank Dr. Frank Fueten and an anonymous reviewer for providing constructive suggestions that improved the manuscript.

Appendix

ArcGIS functions used in the grain boundary detection model listed in alphabetical order. The complete model described in this paper is freely available from the authors.

Cell statistics – sum

Cell statistics are a series of functions to perform a cell by cell-based statistic from multiple input rasters. These functions compute an output raster where the output value at each location is a function of the value associated with one or more input rasters at that location without the consideration of the neighborhood pixels/cells. The Sum function calculates the sum of all input raster values at each pixel/cell to create an output raster.

Con

The Con function performs a conditional if/else evaluation on each of the input pixels/cells of an input raster and create an output raster that meet the specified condition. An example of this conditional evaluation is to create an output raster that has a value of 100 where the input raster is greater than 5, and a value of 50 where the input raster is less than 5.

Feature class to Coverage

This function creates a single coverage (ArcInfo format) from the input feature class. The purpose of this conversion is to create and maintain the shape information (area, perimeter) of the polygons.

Feature to polygon

This function creates polygon features from polyline features.

Focal Max

Focal Max is a neighborhood function that assigns the maximum value of the neighborhood to the specified pixel/cell. For each pixel/cell in the input raster, the function calculates the maximum value within the specified neighborhood and assigns that value to the corresponding pixel/cell on the output grid.

Focal Median

Focal Median is a neighborhood function that assigns the median value of the neighborhood to the specified pixel/cell. For each pixel/cell in the input raster, the function calculates the median value within the specified neighborhood and assigns that value to the corresponding pixel/cell in the output raster.

Focal Range

Focal Range is a neighborhood function that assigns the range (maximum-minimum) value of the neighborhood to the specified pixel/cell. For each pixel/cell in the input raster, the function calculates the range of the values within the specified neighborhood and assigns that value to the corresponding pixel/cell on the output raster.

Iso Cluster

Iso Cluster applies the isodata clustering algorithm to determine the characteristics of the natural groupings of pixels/cells and records the results in an output ASCII signature file.

Maximum likelihood classification

This function performs a maximum likelihood classification on the input raster to create a classified output raster. The signature file created using Iso Cluster function will be used as a valid entry to perform the classification.

Neighborhood

Neighborhood is pixels/cells around the specified pixel/cell. A neighborhood can be a rectangle, circle, annulus (a donut) or a slice of a circle. In this paper, we use rectangle for most of neighborhood functions.

Plus

This function creates an output raster where each cell is the sum of the corresponding cells in two input rasters.

Raster to polyline

This function converts a raster to polyline features.

Region Group

The Region Group function scans the whole raster from left to right and top to bottom to identify all connected regions and assign a unique number to each connected region.

Set null

This function returns NoData if a conditional evaluation is true and the value specified by another raster if the conditional evaluation is false. This function can be used to change all values that meet a certain condition to NoData. This can be used for processing the remaining selected cells, eliminate certain cells for future consideration within a model, or create a mask.

Thin

This function thins rasterized linear features by reducing the number of cells representing the width of the features. After running the Thin function, each linear feature will be represented as a linear feature with a single cell width. This is a necessary step if converting the linear raster to vector polyline features.

Times

This function creates an output raster where each cell is the product of the corresponding cells in two input rasters.

Zonal Perimeter

The Zonal Perimeter function calculates the perimeter value for each connected region (zone) and assigns the value to each pixels/cells within the corresponding region/zone.

References

Adams, R., Bischof, L., 1994. Seeded region growing. IEEE Transactions on Pattern Analysis Machine Intelligence 16, 641–647.

Ailleres, L., Champenois, M., Macaudiere, J., Bertrand, J.M., 1995. Use of image analysis in the measurement of finite strain by the normalized Fry method: geological implications for the 'Zone Houillere' (Brianconnais zone, French Alps). Mineralogical Magazine 59, 179–187.

Bartozzi, M., Boyle, A.P., Prior, D.J., 2000. Automated grain boundary detection and classification in orientation contrast images. Journal of Structural Geology 22, 1569–1579.

Bishop, M.P., Shroder Jr., J.F. (Eds.), 2004. Geographic Information Science and Mountain Geomorphology. Praxis-Springer-Verlag, UK.

Bonham-Carter, G.F., 1994. Geographic Information Systems for Geoscientists. Pergamon, New York.

Choudhury, K.R., Meere, P.A., Mulchrone, K.F., 2006. Automated grain boundary detection by CASRG. Journal of Structural Geology 28, 363–375.

Erslev, E.A., 1988. Normalized center-to-center strain analysis of packed aggregates. Journal of Structural Geology 10, 201–209.

Erslev, E.A., Ge, H., 1990. Least-squares center-to-center and mean object ellipse fabric analysis. Journal of Structural Geology 12, 1047–1059.

Fernandez, F.J., Menendez-Duarte, R., Aller, J., Bastida, F., 2005. Application of geographical information systems to shape-fabric analysis. In: Bruhn, D., Burlini, L. (Eds.), High-Strain Zones: Structure and Physical Properties, vol. 245. Geological Society of London Special Publication, pp. 409–420.

Fry, N., 1979. Random point distributions and strain measurement in rocks. Tectonophysics 60, 806–807.

Fueten, F., 1997. A computer controlled rotating polarizer stage for the petrographic microscope. Computers & Geosciences 23, 203–208.

Fueten, F., Goodchild, J.S., 2001. Quartz c-axis orientation determination using the rotating polarizer microscope. Journal of Structural Geology 23, 895–902.

Gonzalez, R.C., Wintz, P., 1987. Digital Image Processing. Addison-Wesley, Reading, Massachusetts.

Goodchild, J.S., Fueten, F., 1998. Edge detection in petrographic images using the rotating polarizer stage. Computers & Geosciences 24, 745–751.

Guo, Y., Onasch, C.M., 2001. Analysis of grain-scale deformation using a thin section GIS. Geological Society of America Abstracts with Programs 33, 324.

Heilbronner, R., 2000. Automatic grain boundary detection and grain size analysis using polarization micrographs or orientation images. Journal of Structural Geology 22, 969–981.

Heilbronner, R.P., Pauli, C., 1993. Integrated spatial and orientation analysis of quartz c-axes by computer aided microscopy. Journal of Structural Geology 15, 369–382.

Lisle, R.J., 1985. Geological Strain Analysis. A Manual for the R_f/ϕ Technique. Pergamon Press.

Lumbreras, F., Serrat, J., 1996. Segmentation of petrological images of marbles. Computers & Geosciences 22, 547–558.

Obara, B., 2007. An image processing algorithm for the reversed transformation of rotated microscope images. Computers & Geosciences 33, 853−859.

Ramsay, J.G., 1967. Folding and Fracturing of Rocks. McGraw-Hill, New York.

Sangwine, S.J., Horne, R.E.N., 1998. The Color Image Processing Handbook. Chapman & Hall, London.

Starkey, J., Samantaray, A.K., 1993. Edge detection in petrographic images. Journal of Microscopy 172, 263−266.

Stöckhert, B., Duyster, J., 1999. Discontinuous grain growth in recrystallised vein quartz-implications for grain boundary structure, grain boundary mobility, crystallographic preferred orientation, and stress history. Journal of Structural Geology 21, 1477−1490.

Zhou, Y., Starkey, J., Mansinha, L., 2004. Segmentation of petrographic images by integrating edge detection and region growing. Computers & Geosciences 30, 817−831.