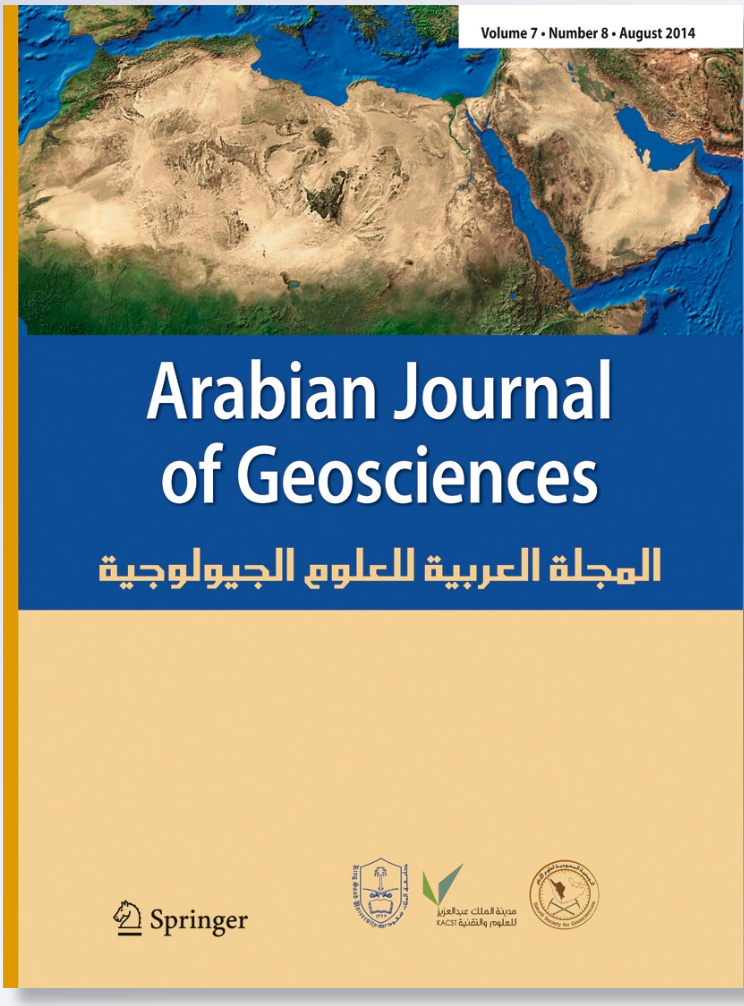
A software solution for point counting. Petrographic thin section analysis as a case study

Martín L. Larrea, Silvia M. Castro & Ernesto A. Bjerg

Arabian Journal of Geosciences

ISSN 1866-7511
Volume 7
Number 8

Arab J Geosci (2014) 7:2981-2989
DOI 10.1007/s12517-013-1032-0



Springer

Arab J Geosci (2014) 7:2981–2989
DOI 10.1007/s12517-013-1032-0

ORIGINAL PAPER

A software solution for point counting. Petrographic thin section analysis as a case study

Martín L. Larrea · Silvia M. Castro · Ernesto A. Bjerg

Received: 5 May 2013 / Accepted: 12 July 2013 / Published online: 28 July 2013

Abstract In the field of geology, as in many other fields, a common problem is the analysis of 2D sections. An example of this is the petrographic study of rock samples, i.e., the quantification of its mineral components and the percentages of each phase present in it. Conventionally, this quantification has been determined by point counting performed on thin sections. Although point counting is very time consuming, it is of common use in several domains including geology, biology, medicine, and materials sciences, among others. Point counting in thin sections is normally conducted through mechanical or electromechanical devices attached to a microscope; such devices are very expensive, offer limited functionality, and are very time consuming. In this contribution, we introduce an interactive visualization application called Rock.AR that reduces the amount of time required to apply this technique and simplify its work flow. It provides visual tools like distortion techniques, overview + detail, and statistics to assist the technique. A comparison between this application and similar tools through users' experiences is provided. We compare the time it takes to complete a session of point counting and the tools' usability.

Keywords Point counting · Petrographic thin sections · Image analysis · Visualization

M. L. Larrea (✉) · S. M. Castro
Departamento de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur, Avenida Alem 1253, Bahía Blanca, Buenos Aires, Argentina
e-mail: mll@cs.uns.edu.ar

E. A. Bjerg
INGEOSUR (CONICET-UNS) and Departamento de Geología, Universidad Nacional del Sur, Avenida Alem 1253, Bahía Blanca, Buenos Aires, Argentina

Introduction

Mineral content, chemical composition, and texture, together with shape and size of mineral, constituents the main rock properties considered by petrologist. The determination of the relative abundances of rock sample's components, i.e., the modal composition, is of research, education, and engineering interest. Modal analysis is one of the most common methods for the classification of rocks, and important for petrologist when establishing evolutionary trends; it provides two types of data series as follows: (a) relative amounts of rock forming minerals and (b) microscopic grain size distribution of the rocks. The knowledge of the relative proportions of mineral phases in a rock sample is of utmost importance for its proper classification.

Point counting is the standard method to establish the modal proportion of minerals in coarse-grained igneous, metamorphic, and sedimentary rock samples. This requires observations to be made at regular positions on the sample, namely grid intersections. At each position, the domain expert decides to which mineral the respective grid point and its local neighborhood belongs. By counting the number of points found for each mineral, it is possible to calculate the percentages that these values represent of the total counted points. These percentages represent the approximate relative proportions of the minerals in a rock, which is a 2D section of a 3D sample. Concerning the statistically correct number of required counts for quantitative examinations, the image to be manipulated must contain between 1,500 and 5,000 points. This value is established by the domain expert before starting the counting procedure, based on the size of the smallest grain of each mineral phase present in the sample.

The case study in this work is the modal analysis of petrographic thin sections; however, there are other methods to perform point counting on rock samples. For example, Higgins (2000) mention that in coarse-grained rocks, large crystals can be measured directly on the outcrop, as is the case of megacrysts in granites or oikocrysts in mafic rocks. In such cases, crystals can be measured with a ruler, or their outlines traced onto an overlay for later measurement, or a mosaic of photographs can be taken for later analysis. While Higgins (2000) is addressing a slightly different topic in his paper, the concept applies to our work. For the Rock.AR software, the image to be analyzed can be a scanned image of a petrographic thin section, a photo from a rock outcrop, or even a hillside.

Although there are automated image processing techniques for counting features on microscope slides in other fields, particularly in biology (Lamprecht et al. 2007), these techniques cannot be easily adapted to petrographic thin sections. Because of the characteristics of a petrographic slide, the color of each conceptual unit is not homogeneous enough to be interpreted as a single component by an automated image processing technique, based on color or gray scale images. Precisely, images obtained from rock thin sections show inhomogeneous colors for each mineral species due to subtle tone changes which are related to inhomogeneous chemical composition and/or deformation due to stress affecting the rock, just to mention two of the most common causes.

In petrographic thin sections, point counting is normally conducted through an electromechanical or electronic device attached to a petrographic or reflected light microscope, a device that is capable of moving the sample at regular steps. Due to the complexities involved in point-counting methods executed with these ad hoc devices, we propose a methodology that replaces the microscope-dependent methods allowing the domain expert to perform the point counting directly on a digitized image of a normal rock thin section. Our tool also provides visualization techniques for the exploration and analysis of the image data.

Rock.AR is a visualization tool with a user-friendly interface that provides a semiautomatic point-counting method. It increases the efficiency of the point-counting task by reducing the user cognitive workload. This tool automates the creation of the grid used to define the point positions. This grid is overlaid on a predetermined image of a sample, allowing the count of minerals identified at the intersections of the grid lines. This method significantly reduces the time required to conduct point counting, it does not require an expensive ad hoc device to perform the job, and it improves the consistency of counts.

The goal of the application is to alleviate the known problems arising when performing point counting, increasing the methods effectiveness, its navigability, and the interaction among the different representations of the data. There are very few published papers devoted to point-counting methods and even less about the use of computer science related topics to improve this technique. We believe that our contribution to this area is significant and will help geologists, mineralogist, and other scientists in their work.

This paper is structured as follows: In the next section, a summary of the previous work is provided. Section 3 begins with a general description of the characteristics of the application including the basic visualization aspects. Based on these characteristics, the interface and the interactions are described in Sects. 4 and 5. Section 6 provides the implementation details of Rock.AR and the requirements to execute the application. Section 7 describes the user's experience of several scientists in the area using Rock.AR. Finally, Sect. 8 summarizes the work providing some closing remarks and directions for future work.

Previous work

The point-counting technique evolved from the work of Delesse (1848), Rosiwal (1898), Shand (1916), and Chayes (1949), among others. Point counting is normally conducted either by hand only or with the aid of different ad hoc techniques and devices. In order to improve the effectiveness of this type of work, some applications were developed. Gatlin et al. (1993) developed a semiautomated method of point counting. They applied this technique on biological samples, and they pointed out that it could be used on many other types of tissue. This method could only be run on Macintosh computers, and it was not publicly accessible. The electromechanical device attached to a petrographic or reflected light microscope, mentioned earlier, was replaced in 2000 by an electronic device which significantly improved accuracy. Its movement was fully automated, allowing software-controlled slide movements in both horizontal and vertical directions. This device was recently enhanced with the addition of a petrographic data analysis system. This is a rather complicated and expensive device, and its software is rather user unfriendly. The device must remain attached to the microscope during the entire point-counting process, which would be a serious drawback if the microscope is a critical resource. In this case, if the microscope has to be used for other tasks, the device must be detached, forcing the user to restart with the whole process again.

In 2006, JMicroVision[1] developed a software application, available on the Web, to analyze high-definition images of rock thin sections and in other domains. Although this application allows point counting, it is not easy to use. Besides that, it does not allow to overlay an ad hoc regular grid on the image with the result that counting cannot be conducted with the desired step size.

Mock et al. (2012) introduced GeoPixelCounter, an automatic web-based application to estimate the percentage of various minerals in an image file of a rock thin section. The program allows the user to click on a pixel of the image, whereby all other pixels in its neighborhood with similar color

[1] http://www.jmicrovision.com/

will be selected automatically. However, this application is not oriented to support point counting.

Our approach

Traditionally, the term visualization has been used to describe the process of graphically conveying or presenting end results (Card et al. 1999). This concept has evolved and revolutionized the way researchers do science. At present, it is required to support the analytical reasoning with highly interactive interfaces in order to gain insight from the collected data. Based on these concepts, we developed a point-counting visualization application that leads to a natural user–computer interaction and enhances the user's ability to count. Our primary goal was to achieve the highest level of automatization of the point-counting technique.

Using our application, a typical counting session consists of the sample setup, the counting process and the exploration session. For the sample setup, a digital image of a petrographic thin section must be loaded into the application. The user is subsequently informed of its dimensions in pixels. Then, the user has the possibility to input its dimensions in millimeter and also to adjust its brightness and contrast. At this step, the user is also asked to create the regular grid for optimum point density and to configure its visual properties, i.e., the color of the grid and the color of the grid at the selected point. Thereafter, the grid is superimposed onto the image, the counting process can start, points can be classified, and the percentages values of each mineral are calculated and graphically displayed in real time.

Anytime during the exploration session, the information of the so far counted points can be saved and replayed for further analysis. In an effort to provide a user-friendly computer application, the interface and the interactions are kept simple and intuitive. This allows the user to create the grid, overlying it on a selected image, and classifying the minerals at the grid points. The information is organized in multiple tightly coupled views and guides the interactive manipulation of the data (see The interface).

In order to do a meaningful classification, the image to be manipulated must contain between 1,500 and 5,000 points. When interacting with large images on the computer display, two goals must be achieved. First, it is important to convey a general overview of all data. Second, it should be possible to view the data at the point in detail. In the classification stage, it is necessary to have a focus on the point and a local context that helps to classify the mineral being also important to have an overview of the thin section, without distortion. To meet all these requirements, we designed a focus + local context + overview display. It must also be pointed out that the local context heavily relies on zoom in + zoom out interactions performed by the users. The following sections describe the application in detail, going deeply into the interface and the interactions available to the user.

The interface

Kwan-Liu (2004) introduced the concept "the user interface of a visualization system is the visualization displayed." Based on this premise, users are able to concentrate on data exploration and interpretation rather than on user interface artifacts. Our interface is built on this premise and presents four views (Fig. 1) providing different contents to improve the point-counting process. The interface offers the user the image of the petrographic thin section in the sample view and complementary information views organized in a mosaic fashion. In the following subsections, we describe the goal of each view, its interactions, and the interface components.

The views

During the data exploration process, the dynamically linked views show quantitative and qualitative information. There are four linked views on Rock.AR as follows: sample view, table of minerals, chart view, and overview.

Sample view

This is the main view in the application and therefore the greatest. Through this, the user may count, watch, and manipulate the petrographic thin section. It allows to see the sample and a grid overlaid on top of it. To control the precision of the point counting, the user can adjust the distance between vertical and horizontal lines of the overlaid grid before the count begins. The grid intersections, marked with circles, represent the points to count. When the user selects a point, a square is drawn around it; the color of the circle and the intersected grid lines at its center is changed to emphasize the selected point.

Table of minerals

This view represents the list of the minerals the user is looking for in the sample. Each mineral has a key reference color, allowing to easily identifying it in the four views. Different types of minerals correspond to different colors. Each table row represents one mineral, and for each mineral, the following information is displayed: a key value to be referenced, its name, the associated color, the total counted points of that mineral, and the percentage they represent considering the total number of counted points on the sample.

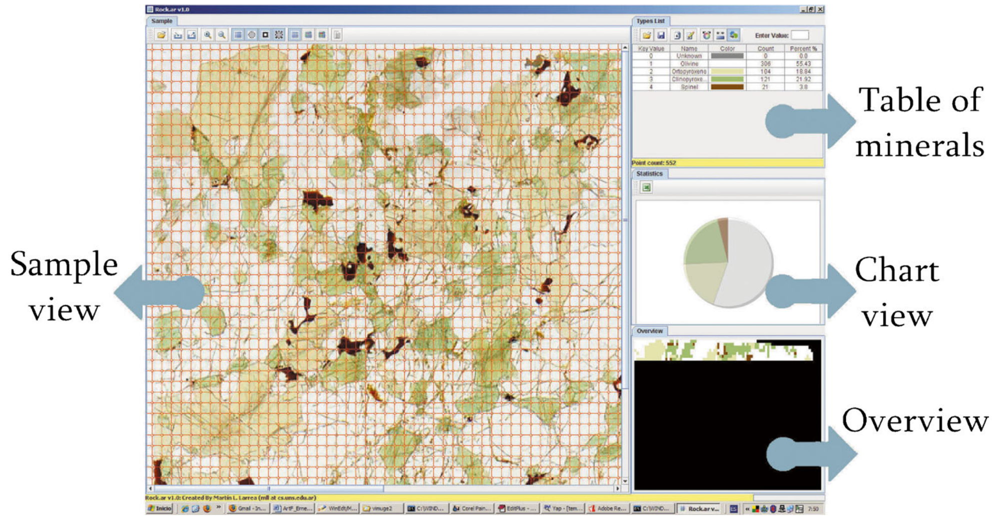


Fig. 1 Screenshot of the Rock.AR application. This application is computer platform-independent. Rock.AR provides a semiautomatic point-counting method for petrographic thin sections. The interface offers the user the image of the petrographic thin section in the sample view and complementary information views organized in a mosaic fashion

Chart view

The most important output of this process is the percentages values of each mineral. These values represent the relative area occupied by each of them. This view shows a graphical representation of those percentages values using the color key references created in the table of minerals.

Overview

Additionally to the sample image, its pixel map is also shown. This map represents each counted point in the thin section with the corresponding mineral color. Pixels are arranged according to the image being counted, offering an intuitive context overview.

The interactions

In this section, we discuss several aspects of interacting with the application. User interfaces should become simple and intuitive, so that they can be operated by people with different levels of knowledge and skills. Users should be able to freely choose and interchange their ways of interaction, depending on their skills and the particular situations. That is the reason why applications will especially benefit from simplified and powerful human–computer interfaces that allow users to choose the type of interaction depending on their background and their preferences (Medyckyj-Scott and Hearnshaw 1993).

All interactions in Rock.AR can be done with the mouse or with the keyboard. We consider mouse interactions to be more suitable for novice users, especially those who are learning the technique. Keyboard interaction is better for expert users because it provides a quick access to the application functionality. The user is free to configure which key will trigger each action.

We identified three interaction tasks as follows: The configuration to begin the procedure, the way the petrographic thin section image will be navigated to perform the point counting, and the removal and modification of any counted point when necessary. In the following paragraph, we describe each of the three interactions in more detail:

1. The configuration stage. When an image has been selected, the user can choose to:
   (a) Adjust the brightness and contrast of the image.
   (b) Set the grid size in order to determine the number of points to count and the step between them.
   (c) Set the colors of the grid elements.

2. Navigation of the petrographic thin section image. We propose a strategy that supports people in this initial interaction stage with the application and a different one for expert users. Users should be able to freely choose and interchange their ways of interaction, depending on their abilities and the situations at hand. The user can select and navigate the points by mouse movement, for novice users, or keyboard input for expert users. The user then proceeds to classify them by selecting the key mineral value; this can also be done by mouse or by keyboard interaction. The navigation over the grid can be unrestricted, accessing freely to any desired point on the grid or predefined, following a preestablished path. In the last case, starting at the top left corner point, when pressing <Enter> on the first row, the next selected point will always be the next at the right. At the end of the row, the next selected point will be the one below the current one, and then the movement will proceed from right to left. This navigation pattern is repeated on a recurring basis until the last point is reached. At the current point, a mineral can be associated to it, and the statistics are updated.

3. Addition, removal, and modification of the different counted points in the sample. To modify a counted point, the user must select the point, press the <Backspace> key, and enter the new value. If no new value is entered, the mineral tied to the point becomes undefined, and the statistics is updated.

Besides these three tasks, in the following, we describe an additional set of actions categorized according to the view they affect.

Interactions on the sample view

- Walk through the points. The user can move through the points by mouse click or by pressing the arrows keys. When the user types a mineral key into the mineral key textbox and presses <Enter>, an automatic movement takes place, and the next point is selected.
- Import/export counted information. The user can save the counted point in a file. This file can be used later to resume the work.
- Remove grid components. The visibility of the following grid elements can be toggled by the user at any moment (Fig. 2): the entire grid, the circles in the grid intersections, the squares and the lines inside it when a point is selected, and the circle at the selected point.
- Zoom in/out. The zoom can be applied to the whole image or just locally to a point in the grid. In the last case, the zoom in allows to view smaller sections inside the square drawn when the point is selected (Fig. 3).
- Overview overlap. It is possible to combine the overview information with the sample view (Fig. 4) with transparency or full opacity. This will allow the user to see which points have been counted and how they were counted.

Interactions on the table of minerals

- Open/save table of minerals. Each list created by the user can be saved as a file to be reused.
- Add new mineral. A new mineral can be added to the list, its key value will be automatically assigned, the user will be asked to enter its name and color references.
- Edit mineral. For each mineral, the user can change its name and color references.

Interactions on the pie chart

- Quantitative values. By default, only the pie chart is visible in the respective view; if the user moves the mouse pointer over the graphic, the quantitative values will appear. However, if she/he wants to keep these values visible, the graphic must be clicked. A second click will make the values invisible again.
- Export data. When the user clicks on the MS Excel® button, a spreadsheet file will be created containing the table of minerals with the number of counted points, its percentages, and color references.

Implementation

Rock.AR was implemented using Java 1.5. Additionally to the Java libraries, three special purpose libraries were used in Rock.AR: JIU—The Java Imaging Utilities for image processing, scaling, copying, and cutting images, JFreeChart to create the pie chart, and Java Excel API to create and write an MS Excel® file.

Rock.AR runs on Microsoft Windows ® (XP and above) and all Linux distributions. The application does require the Java library to be installed. For the moment, Rock.AR is not available on the Web because it is in a beta version, but it can be obtained by contacting the authors. Once a stable version is reached, Rock.AR will be made available as an open-source application.

User experience

We conducted a controlled lab study to measure how our application could improve the point-counting technique as compared to other methods. Our general hypothesis was that,

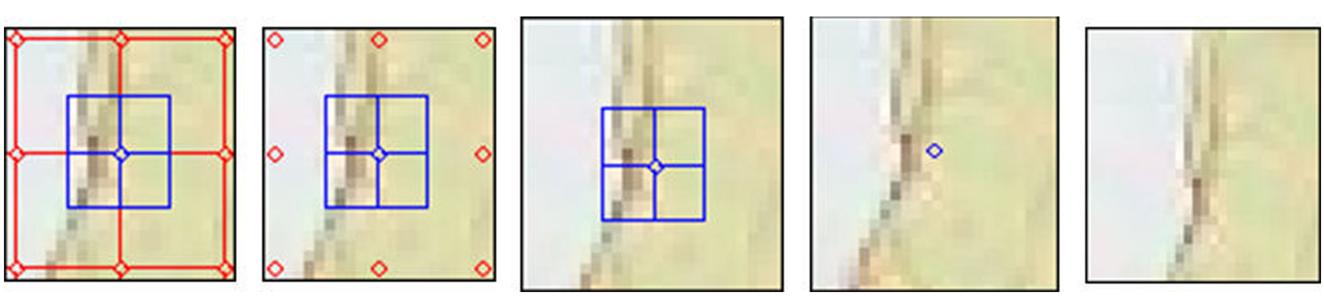


Fig. 2 All graphical elements inside the main panel can be made invisible by user request

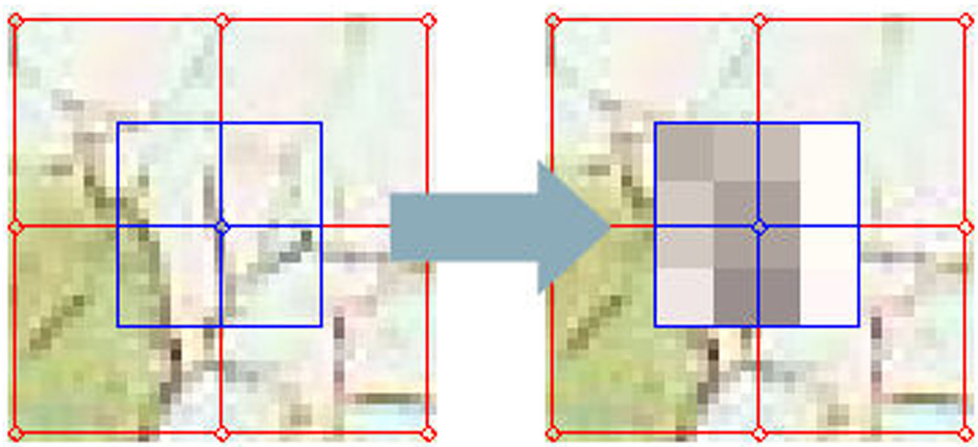


Fig 3 Local zoom applied to the selected point, highlighted in *blue*, on the right

with Rock.AR, the users would be able to count faster, more accurately, and with greater user satisfaction.

Participants

Twelve participants were involved in this evaluation, all of them from the Geology Department of our university. We screened participants so that all of them have had some experience with point counting through the use of a microscope but none with any point-counting software. They did have experience as PC users. We also screened participants for color blindness. The participants were in the 29–51 age groups, with a mean of 38.3.

Tasks and samples

We broke down the whole counting process into the following three tasks:

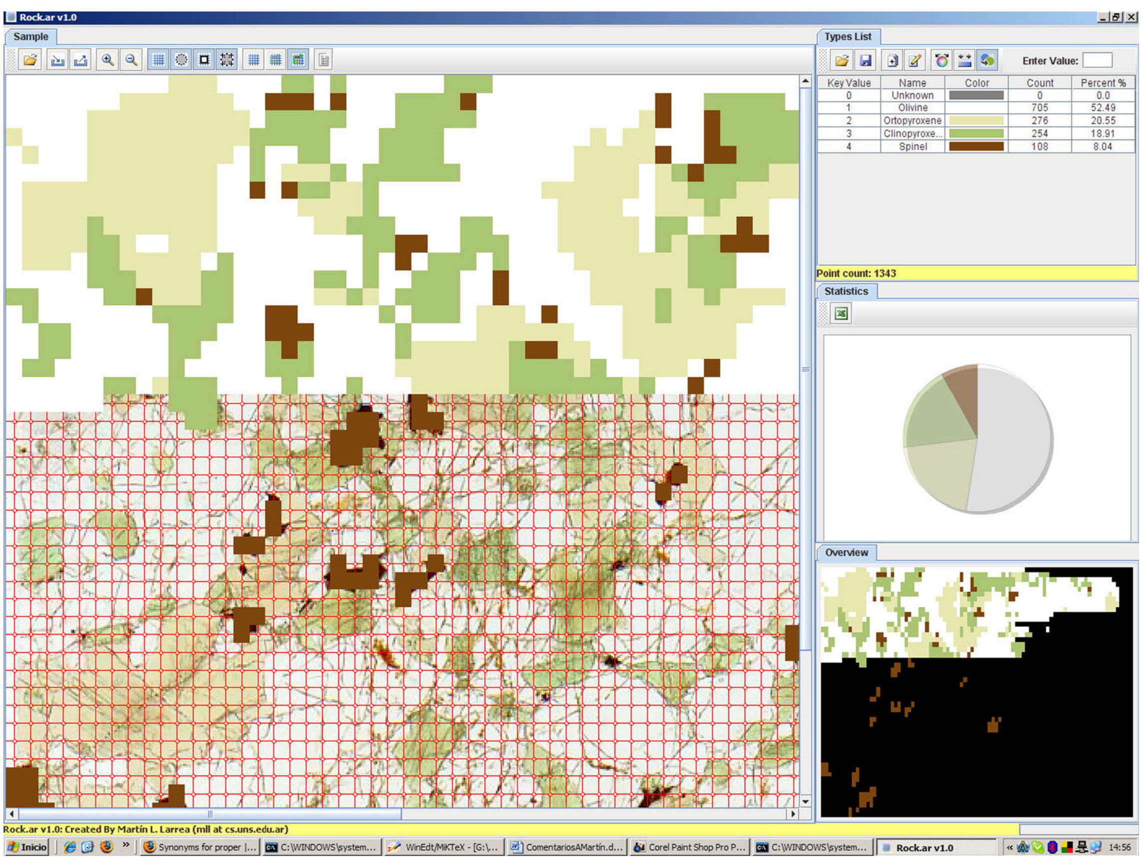


Fig. 4 The overview information with the sample using an overlap layer with full opacity. This allows the user to see which points have been counts and how they were counted

1. Set up: This task is oriented to set the point-counting application on picking up a sample image, adjusting its size, brightness, and contrast, configuring the grid, and finally creating the minerals list.
2. Point counting: This is to associate a mineral from the minerals list to each point in the grid.
3. Create the statistics: This allows creating the sample statistics.

All users were asked to produce a table and a pie chart. To run this evaluation, we used three petrographic thin section images called S1, S2, and S3. In spite of their differences, the three samples had similar complexity. This was checked by a domain expert, who also did the point counting on the three samples, and the results were used as the reference set and will be referred as the trusted results.

Methods

We matched up the Rock.AR against two other point-counting methods, JMicroVision and the manual method. We took into account these two methods because they are clearly different approaches to the same technique. The microscope-based one is the most common manual method, and the JMicroVision is a computer approach to improve the technique. Despite that the goal of Rock.AR is very close to that of JMicroVision, we will show that its design will improve time, accuracy, and user satisfaction because of its interactions and new features.

Study design and procedures

Originally, we proposed 12 participants, distributed into three groups. Each group was going to work with the three tools. Because the manual method is very time consuming, only two participants carried out this test. The remaining ten participants were divided into two groups. To identify each group, we gave them a call name based on a color, RED and GREEN group; we got a random distribution on each group by assigning the members to each group following their arrival order. The experiment consisted of two sessions, one per day. In each session, each group worked with one tool on one sample. The user's goal was to perform a point count on a given sample using the given method. Each user had to go through the three tasks that we described earlier.

It was very important to ensure the validity of the experiment. Validity is mainly concerned with the question of whether the experiment measures what it is supposed to measure. We considered two types of validity: internal and external. The extent of internal validity is to check that the differences in value of the dependent variables, i.e., the outcome variables, are currently caused by the independent variable, i.e., the variable that is manipulated by the experiment and not by any other source of variation. The external validity of an experiment is concerned with the following question: How well do the results of the experiment generalize beyond the sample of subjects in the experiment?

In our case, the dependent variables were users' satisfaction and the time it took to complete each task; the independent variables were the systems used to carry out such tasks. We have internal validity if we can ensure that the differences in the times of each task on each system are a true representation of the differences between systems. We have external validity if we can guarantee that the selected group of users is an actual representation of the real ones.

We achieved internal validity by assigning subjects to the test groups randomly and by having the same level of expertise among all users. Also, to avoid the effects of external variables on the final results, we used a within-subject design for the experiment. In this design, all subjects are used two times, once with Rock.AR and another with JMicroVision, not necessarily in this order. By doing this, we are canceling any possible carryover effect.

Results

Tables 1, 2, and 3 summarize the time it took users to complete the three tasks with each of the methods. It is important to remark that the second task is the most important one, and it is the most time consuming one. At the end of each task, for both the JMicroVision and Rock.AR, participants had to answer the questions below, providing a number between 1 and 5. These questions were made taking into account the work by

Table 1 Time spent to complete the first task. The set up is faster with Rock.AR

Task 1: The set up
Time is in hours/minutes/seconds

Group	User	Rock.AR	JMicroVision	Microscope
Red	U1	00:00:54	00:03:20	none
	U2	00:01:12	00:02:54	none
	U3	00:01:16	00:03:40	none
	U4	00:01:04	00:02:37	none
	U5	00:01:07	00:03:21	none
Green	U6	00:00:50	00:04:12	none
	U7	00:01:20	00:03:43	none
	U8	00:00:46	00:03:48	none
	U9	00:01:10	00:03:54	none
	U10	00:00:58	00:03:26	none
Microscope	U11	none	none	00:03:13
	U12	none	none	00:02:49
	Average time:	00:01:04	00:03:30	00:03:01
	S. derivation:	00:00:11	00:00:27	00:00:12

Table 2 Time spent to complete the second task. The second task is the most important one, as it is the most time consuming one. Like with task one, users working with Rock.AR accomplished this task in a shorter time.

Task 2: The counting process
Time is in hours/minutes/seconds

Group	User	Rock.AR	JMicroVision	Microscope
Red	U1	01:02:23	01:45:31	none
	U2	01:04:10	01:32:58	none
	U3	00:59:36	02:33:10	none
	U4	00:57:02	01:42:34	none
	U5	01:05:48	01:43:49	none
Green	U6	00:46:10	01:41:45	none
	U7	00:51:32	01:53:32	none
	U8	01:02:09	01:31:16	none
	U9	00:58:01	02:03:08	none
	U10	00:59:31	01:57:13	none
Microscope	U11	none	none	05:17:33
	U12	none	none	04:57:14
	Average time:	00:58:38	01:50:30	05:07:24
	S. derivation:	00:05:39	00:17:05	00:10:10

Shneiderman and Plaisant (2004) who proposes certain models of questionnaire to evaluate various types of applications.

For the set up, task one, the question was:

- *How do you rate the interface*? *Give a number between 1 and 5*, *where 1 is intuitive and 5 is artificial*. For JMicroVision, the average answer was 4, and for Rock.AR, it was 2.

For the counting process, task two, the questions were:

- *How do you rate the interface*? *Give a number between 1 and 5*, *where 1 is intuitive and 5 is artificial*. For JMicroVision, the average answer was 5, and for Rock.AR, it was 2.
- *How do you rate the interactions*? *Give a number between 1 and 5*, *where 1 is intuitive and 5 is artificial*. For JMicroVision, the average answer was 4, and for Rock.AR, it was 1.

For the statistics, task three, the questions were:

- *Which is your overall reaction to this method*? *Give a number between 1 and 5*, *where 1 is frustrating and 5 is satisfying*. For JMicroVision, the average answer was 2, and for Rock.AR, it was 4.
- *How do you rate the amount of information on screen*? *Give a number between 1 and 5*, *where 1 is inadequate and 5 is adequate*. For JMicroVision, the average answer was 4, and for Rock.AR, it was 4.

This set of questions reveals that Rock.AR got better results than JMicroVision. All participants were able to finish all the tasks. According to the users, the most negative issue about JMicroVision was its complex interface, while in the case of Rock.AR, they pointed out that the language used in the menus was not appropriate for the geology domain.

The success rates for the counted points with Rock.AR were found to be over 97 % and with JMicroVision about 91 %. In the after sessions discussions, all users agreed how helpful it was to locally zoom in a point while keeping the context. We believe that this feature, in conjunction with an easy and intuitive set of interactions, is the major cause of success for the low number of wrongly counted points. With regard to user satisfaction, participants consistently rated the Rock.AR software higher than JMicroVision.

Table 3 Time spent to complete the third task. Users accomplished this task in less time with Rock.AR

Task 3: The statistics
Time is in hours/minutes/seconds

Group	User	Rock.AR	JMicroVision	Microscope
Red	U1	00:00:54	00:03:31	none
	U2	00:01:12	00:02:58	none
	U3	00:01:16	00:03:10	none
	U4	00:01:04	00:02:34	none
	U5	00:01:07	00:03:49	none
Green	U6	00:00:50	00:04:45	none
	U7	00:01:20	00:03:32	none
	U8	00:00:46	00:03:16	none
	U9	00:01:10	00:03:08	none
	U10	00:00:58	00:03:13	none
Microscope	U11	none	none	00:02:27
	U12	none	none	00:03:11
	Average time:	00:01:04	00:03:24	00:02:49
	S. derivation:	00:00:11	00:00:33	00:00:22

Conclusions and future work

Rock.AR is a visualization tool mainly designed to do point counting on images, particularly those obtained from petrographic thin sections. It is implemented in the Java programming language, has a user-friendly graphical interface, uses information visualization techniques, and is available for Microsoft Windows ® (XP and above) and all Linux distributions. The software will also be freely available for download. These attributes qualify it to be used as a research and educational tool for various users and especially to provide geologists, and other scientists, with a valuable educational and research tool.

Because the point counting is a very tedious and time-consuming task, we are exploring alternate methods to provide a more automatic way to do it. Our goal is to provide the application with an automatic count of the points. We assume that, on the basis of a number of points classified from the user, the application can learn how to classify most of the sample points with a minimal error rate and much less commitment of time.

Acknowledgments This research is partially supported by the PGI 24/N020, 24/H108, and PIP 112-200801-02306. We would like to acknowledge the support from Secretaría General de Ciencia y Tecnología, Universidad Nacional del Sur and CONICET, Argentina.

References

Card SK, Mackinlay JD, Shneiderman B (1999) Readings in information visualization: using vision to think. Morgan Kaufmann Publishers Inc

Chayes F (1949) A simple point counter for thin section analysis. Am Mineral 34:1–11

Delesse MA (1848) Procédé méchanique pour determiner la composition des roches. Annales des Mines 13:379–388

Gatlin CL, Schaberg ES, Jordan WH, Kuyatt BL, Smith WC (1993) Point counting on the Macintosh. A semiautomated image analysis technique. Anal Quant Cytol Histol 15(5):345–350

Higgins M (2000) Measurement of crystal size distributions. Am Mineral 85:1105–1116

Kwan-Liu Ma (2004) Data Visualization. Berkshire Encyclopedia of Human-Computer Interaction. Berkshire Publishing Group

Lamprecht MR, Sabatini DM, Carpenter AE (2007) Cell Profiler™: free, versatile software for automated biological image analysis. Biotechniques 42(1):71

Medyckyj-Scott D, Hearnshaw HM (1993) Human Factors in Geographical Information Systems. Halsted Press

Mock K, Amato J, Bertmaring J (2012) GeoPixelCounter (Version 1.0) [software]. Retrieved from http://www.math.uaa.alaska.edu/~afkjm/GeoPixelCounter/

Rosiwal A (1898) Über geometrische gesteinsanalysen, ein einfacher weg zur ziffermässigen festellung des quantitätsverhältnisses der mineralbestandtheile egmengter gesteine. Verhandlungen der Kaiserlich-Königslich geologischen Reichsanstalt 143–175

Shand SJ (1916) A recording micrometer for rock analysis. J Geol 24:394–403

Shneiderman B, Plaisant C (2004) Designing the User Interface: Strategies for Effective Human-Computer Interaction (4th Edition). Pearson Addison Wesley