

Geonetdigitizer: Open source code to digitalize information given in Wulff and Lambert nets

Ludger O. Suarez-Burgoa ^a

^a *Departamento de Ingeniería Civil, Universidad Nacional de Colombia, Medellín, Colombia. losuarezb@unal.edu.co*

Received: May 09th, 2014. Received in revised form: October 16th, 2014. Accepted: February 25th, 2015.

Abstract

The article carries out the theoretical basis, the usefulness, and validation of the computational program geonetDigitizer (licensed as an open-source code as BSD-2) in order to apply as a toolbox for the programming language MATLAB[®]. The program geonetDigitizer was designed for the digitalization of plots expressed in the spherical projections Wulff and Lambert, which is of extensive use in structural geology and geomechanics. The reasons of presenting the program as an open code has the purpose to recover historical data presented in those projections, as also to recover old and modern information presented in that projections. All of these will reduce the possible phenomenon known as Digital Dark Age. To validate the code, it is presented here two examples, where real known values are compared with their corresponding values obtained by the process of digitizing. Finally, in the appendix is presented two application examples.

Keywords: Digitizing, spherical projection, structural geology, geomechanics, open-source code.

Geonetdigitizer: Código abierto para digitalizar información plasmada en mallas Wulff y Lambert

Resumen

El artículo desarrolla el fundamento teórico, el uso y validación del programa computacional geonetDigitizer (licenciado como un código abierto bajo la modalidad BSD-2) para aplicarlo como una caja de herramientas para el lenguaje de programación MATLAB[®]. El programa geonetDigitizer fue diseñado para la digitalización de gráficas plasmadas en las proyecciones esféricas Wulff y Lambert, que es de uso extensivo en geología estructural y geomecánica. Las razones por presentar el programa como código abierto tiene el fin de recuperar información histórica, antigua y moderna presentada en estas proyecciones; y así también reducir el fenómeno conocido como Era Digital Desconocida. Para la validación se presenta aquí dos ejemplos, donde se compara los valores reales conocidos con los correspondientes valores obtenidos del proceso de digitalización. Finalmente, en el anexo se presenta dos ejemplos de aplicación.

Palabras clave: digitalización, proyección esférica, geología estructural, geomecánica, código abierto.

1. Introducción

Es muy común en las Ciencias de la Tierra, que los datos de medidas de planos, lineamientos u otra facción geométrica tridimensional de la roca se haga en gráficas bidimensionales; gráficas que resultados de una proyección estereográfica (malla de Wulff) o de una proyección equiárea Lambert (malla Lambert o malla de Schmidt¹), siendo ambas versiones casos especiales de la proyección esférica. La

proyección esférica es útil para hacer una transformación de \mathbb{R}^3 a \mathbb{R}^2 sobre la superficie de una esfera unitaria, también usada en cartografía; y hasta ahora no fue sustituida inclusive con el actual desarrollo de las tecnologías de información y comunicación, y los diversos métodos de visualización tridimensional (e.g. la visión tridimensional a través de lentes polarizados).

Sin embargo, se tienen dos interrogantes ¿Qué ocurrió con la información desarrollada antes del uso de los

¹ Malla de Schmidt: nombre solo conocido en el ámbito de la geología, pero más se reserva a la gráfica que resulta del método de Schmidt para hacer la

representación de los contornos de concentración de los polos de los planos de discontinuidades, y no a la proyección en sí.

computadores, plasmada en documentos impresos en papel o microfilm? y ¿Qué está ocurriendo con la información que se ha estado produciendo desde el uso de los computadores hasta la actualidad?

Antes del uso extensivo de los computadores y desde la segunda década del siglo XX, toda información de las facciones geométricas tridimensionales de estructuras geológicas fue plasmada en alguna proyección esférica y un medio físico (e.g. papel o microfilm); a esta información la llamaremos información histórica. Durante la transición hacia el uso extensivo del computador, la información se guardó de forma paralela en medio físico y digital, la cual la llamaremos información antigua; y en la actualidad, la tendencia es sólo de guardar la información en medio digital (v.gr. *información actual*), sea como una base de datos estructurada de variables tridimensionales (*información editable*) o como una imagen de la misma proyección esférica (*información no editable*).

Respecto a la conservación en sí de la información (esta vez de cualquier tipo), ésta se hace en dispositivos de almacenamiento personales o en centros de datos a través de las nubes (por ejemplo Dropbox y Google-Drive son servidores de esta naturaleza).

No obstante, la conservación segura de la información digital aún no es garantía para que de aquí a unos pocos años se pueda recuperar algo de aquella. Esto debido a que mucha de esa información está codificada y sólo puede recuperarse mediante el uso de los programas que lo crearon. De no lograr la conservación, la información almacenada padecería de un fenómeno que se ha denominado era digital desconocida (DDA de las siglas del inglés de Digital Dark Age) [3].

Por tal motivo, muchas veces el único y último respaldo que uno tiene es el impreso en papel que se logró conservar en las publicaciones que normalmente están disponibles en las bibliotecas de las universidades o en la Biblioteca Central de cada país; o más reciente, lo único que tiene es la imagen de la representación gráfica en alguna de las proyecciones esféricas usadas en las geociencias. De este modo, cualquier persona podría recuperar la información mediante un proceso de digitalización; y además podrá abarcar la información histórica, antigua y moderna con este procedimiento.

Sin embargo, la digitalización de datos en proyección esférica no es directa como lo sería en el caso sencillo de una digitalización de una gráfica bidimensional en proyección cartesiana, aun siendo esta primera una representación en \mathbb{R}^2 de facciones \mathbb{R}^3 . Por tal motivo es necesario tener a la mano un programa de digitalización especializado.

Lo que observé luego de una búsqueda de paquetes computacionales en la red Internet, concluyo que es muy poco probable que exista un programa que sea un código abierto que evite un DDA y que además logre la tarea de digitalizar los datos de geología estructural y geomecánica representados en la proyección esférica.

De este modo, desarrollé un código abierto tipo caja de herramientas para MATLAB© (más conocido en el contexto internacional de este programa como *toolbox*); que sirve para digitalizar información plasmada en gráficas de la proyección esférica para los casos más usados en geociencias como son

las Wulff y Lambert, para el hemisferio sur. El *toolbox* fue denominado *geonetDigitizer*.

En las siguientes secciones de este artículo, haré una resumida explicación del marco conceptual que involucran los cálculos de este útil grupo de funciones-código, de sus capacidades y limitaciones, y las futuras mejoras que se le puede hacer. Finalmente se presenta: dos ejemplos de validación del programa, donde los datos reales son conocidos; y dos ejemplos de digitalización de información, una histórica y otra antigua.

La recuperación de información obtenida de imágenes digitales (v.gr. información moderna) es similar a los dos ejemplos.

2. Fundamento teórico

Por lo normal, la orientación de las facciones geométricas de interés en geomecánica y en geología estructural (tales como planos o líneas en el espacio) se presenta a través de una proyección de las intersecciones de las facciones con la superficie de una esfera unitaria, que forman finalmente trazas.

Dependiendo de la aplicación, la proyección de la traza de cada facción con la esfera se hace con una línea o un arco circular; y la traza se proyecta desde uno de los polos (sea el sur o el norte) sobre un plano horizontal ubicado en el ecuador de la esfera, o en uno de sus polos. Todas estas variaciones hacen que se tenga distintos tipos de proyecciones esféricas; pero las más usadas en geomecánica y geología estructural son: la proyección Wulff (llamada también *proyección estereográfica*) y la proyección Lambert.

Estas proyecciones pueden representar una transformación donde los ángulos se conservan (v.gr. proyección equiángulo, correspondiente a la proyección Wulff) o donde las áreas se conservan (v.gr. proyección equiárea, correspondiente a la proyección Lambert). Estas transformaciones son de \mathbb{R}^2 y están bien documentadas en diversos textos referentes al tema, por ejemplo en [12].

Sin embargo, para la digitalización de datos, que están ya representados en \mathbb{R}^2 de la proyección esférica, la primera transformación que se necesita es una de \mathbb{R}^2 a \mathbb{R}^2 : aquella de un plano horizontal cartesiano —que representa el plano de la hoja o la imagen que contiene la información— a otro plano horizontal no-cartesiano que representa el plano de la proyección esférica. Luego, la segunda transformación es una inversa de \mathbb{R}^2 a \mathbb{R}^3 .

2.1. Primera transformación

El método que se siguió para la primera transformación de \mathbb{R}^2 a \mathbb{R}^2 se basa en una transformación que se usa en la disciplina de *procesamiento de imágenes* (v.gr. *image processing*) y *visión artificial* (v.gr. *computer vision*) que se denomina *registro de una imagen* (v.gr. *image registration*). En el registro de una imagen existen varios métodos, pero la que se necesita en primera instancia para el objetivo de este código es el recurrir a un caso especial de una *transformación afin no reflectiva*, donde la dirección de los factores de escalaje es cero $\varphi = 0$ (e.g. [12] o [11] pp. 39-40).

Para lograr tal transformación se hace lo siguiente.

1. Se establece un sistema coordenado plano dextrógiro (sistema S_1) concordante con la imagen que se desea digitalizar. Todas las imágenes establecen un sistema coordenado donde el origen está en la parte superior-izquierda de ellas, además que el incremento de los valores de las columnas (eje x) es de izquierda a derecha, y el incremento de los valores de las filas (eje y) es de arriba abajo. La numeración de los píxeles indica que el sistema es un conjunto de pares ordenados de valores discretos, es decir

$$S_1 = \{(x,y): x \in \mathbb{N} \wedge y \in \mathbb{N}\};$$

donde, $x = [1, \infty]$ y $y = [1, \infty]$.

2. Se establece otro sistema coordenado plano dextrógiro (sistema S_2) concordante con el origen del diagrama estereográfico (v.gr. el centro del círculo mayor, donde éste es la intersección de la esfera con el plano horizontal que pasa por el ecuador y uno vertical que pasa por el polo), y ejes cartesianos x y y, donde x apunta del centro hacia arriba y el eje y apunta del centro hacia la derecha, es decir

$$S_2 = \{(x,y): x \in \mathbb{R} \wedge y \in \mathbb{R}\};$$

además, $x = [-1, 1]$ y $y = [-1, 1]$.

3. Se realiza el registro de la imagen del sistema S_1 al S_2 con cinco puntos de control: cuatro puntos ubicados en el contorno del círculo mayor y en las orientaciones Norte, Este, Sur y Oeste; y uno ubicado en el centro de ese círculo.
4. Se hace la lectura en términos de píxeles de los puntos que se desea digitalizar. Esto se hace con un dispositivo de entrada como el ratón (v.gr. *mouse*), pero es más preciso y rápido hacerlo con una tableta digitalizadora con su lápiz incorporado. Asimismo, mientras la pantalla sea más grande y con alta resolución se tiene mejores ventajas en cuanto a la precisión de que el usuario le atine al punto.
5. Con las coordenadas en píxeles de los puntos digitalizados y con H_A se hace la transformación afín $S_1 \rightarrow S_2$.

La transformación tiene cinco grados de libertad, por tanto dos puntos y un ángulo son suficientes para tener una solución cerrada-lineal. Sin embargo, como este código usa cinco puntos de control, existe redundancia; y como existen siempre errores inmersos en la toma de las coordenadas de los puntos de control, la solución se convierte en un problema no-lineal. Que exista redundancia es benéfico para reducir el error durante la digitalización. Por conveniencia de usar un método numérico ya propuesto, se escogió --para resolver este problema no-lineal en el presente código-- aquel denominado por partes (v.gr. *piecewise method*) [8].

2.2. El registro de la imagen

El registro de la imagen se hace digitalizando los cinco puntos de control en la imagen, cuyas coordenadas en el

sistema S_1 son desconocidos; mientras que el sistema S_2 es conocido. El proceso de registro da los elementos de las matrices de transformación, conformada por: una rotación de ejes en un ángulo θ (R_θ); dos escalajes en la dirección x y y, respectivamente λ_1 y λ_2 (v.gr. *escalaje isotrópico*); y un desplazamiento de un origen a otro dado por el vector t . El ángulo φ indica la orientación respecto al eje x de la línea que representa la deformación angular máxima debido a únicamente una deformación de corte puro.

La rotación tiene el objetivo de que el eje Sur-Norte del diagrama sea paralelo a las columnas de píxeles de la imagen; el escalaje en los dos ejes tiene el objetivo de conocer la cantidad de píxeles que hacen una unidad en las direcciones x y y, y el desplazamiento tiene el objetivo de encontrar el par de píxeles coincidente con el centro del círculo mayor.

Luego de hacer operaciones se obtiene una sola matriz de transformación

$$H_A = \begin{pmatrix} A & t \\ O^T & 1 \end{pmatrix}; \quad (1)$$

donde, $A = R(\theta) R(-\varphi) D R(\varphi)$.

Pero como $\varphi = 0$ (el caso especial), entonces $R(-\varphi) = R(\varphi) = I$; por tanto

$$A = R(\theta) D. \quad (2)$$

La matriz D es una diagonal

$$D = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}. \quad (3)$$

Tanto la existencia de un vector desplazamiento t , como la forma de la matriz H_A , obliga para que las coordenadas sean transformadas de \mathbb{R}^2 a coordenadas homogéneas en \mathbb{R}^3 . Al final de este paso, se tiene como datos conocidos todos los elementos de la matriz de transformación H_A .

2.3. Segunda transformación

Para la segunda transformación, ya no es necesario usar coordenadas homogéneas debido a que los datos ya no sufren un desplazamiento, y todos ellos están referenciados a un origen común. También se observará que existen otras transformaciones secundarias en el proceso. Los pasos para lograr la transformación son.

6. Descifre el tipo de proyección esférica a la que la imagen pertenece: si es equiángulo (proyección Wulff) o si es equiárea (proyección Lambert). Esta información se logra solo si se tiene información explícita en el texto de dónde proviene la imagen o si está dibujada la malla de la proyección, pero este último caso siempre se ha evitado en la presentación de estas gráficas debido a que la convierte en una imagen densa en detalles.
7. Expresé los puntos (x, y) del sistema S_2 en coordenadas polares (θ, ρ) no convencionales (como se ve en las Ecuaciones 4a y 4b), $S'_2 = \{(\theta, \rho): \theta \in \mathbb{R} \wedge \rho \in \mathbb{R}\}$;

donde,

$$\theta = \begin{cases} \theta' & \text{para } x \geq 0 \text{ y } y \geq 0 \\ \pi - \theta' & \text{para } x \geq 0 \text{ y } y \leq 0 \\ \pi + \theta' & \text{para } x \leq 0 \text{ y } y \leq 0 \\ 2\pi - \theta' & \text{para } x \leq 0 \text{ y } y \geq 0 \end{cases}, \quad (4.a)$$

y

$$\rho = \sqrt{x^2 + y^2}; \quad (4.b)$$

para θ comprendido en $[0, 2\pi]$ y ρ comprendido en $[0, 1]$. El ángulo θ' es aquel ángulo absoluto reducido al primer cuadrante

$$\tan \theta' = \left(\frac{|x|}{|y|} \right). \quad (5)$$

- Transforme las coordenadas polares a coordenadas esféricas según el tipo de proyección ya especificada, para tener los valores de acimut (δ_{dir}) y cabeceo (δ) a partir de las coordenadas esféricas [19]

$$\delta_{dir} = \theta. \quad (6.a)$$

y

$$\rho = \begin{cases} \frac{\cos \delta}{1 + \sin \delta} & \text{para proyección Wulff.} \\ \sqrt{2} \cos \left(\frac{\pi + \delta}{2} \right) & \text{para proyección Lambert.} \end{cases} \quad (6.b)$$

Si se quiere el resultado en valores de dirección de buzamiento (ζ_{dir}) y buzamiento (ζ), haga la transformación directa a partir de los datos de acimut y cabeceo antes encontrados; como sigue:

$$\delta_{dir} = \begin{cases} \zeta_{dir} + \pi & \text{para } \zeta_{dir} \leq \pi \\ \zeta_{dir} - \pi & \text{para } \zeta_{dir} > \pi \end{cases} \quad (7.a)$$

y

$$\delta = \frac{\pi}{2} - \zeta. \quad (7.b)$$

3. Materiales

Para usar esta herramienta es necesario que el usuario tenga conocimientos básicos de MATLAB®, al nivel de poder entrar a estos programas y ejecutar las funciones básicas incorporadas en ellos, cargar y guardar variables, importar y exportar imágenes, exportar o importar datos en este entorno. Estos conocimientos pueden adquirirse en los primeros capítulos de por ejemplo [10] que es una guía general de la aplicación de estos programas en las ciencias, pero los capítulos 2 a 4 y el 7 son una excelente introducción para su manejo; [29] que tiene muchas otras aplicaciones de estos programas a la geología, pero el Cap. 2 es una buena introducción para su manejo; o [1], que aunque asume que el lector tiene conocimiento de estos programas, se describe muchas aplicaciones de la ingeniería estructural resueltas en ellos.

También, el usuario tiene que tener conocimiento de: los conceptos de la proyección estereográfica y de la proyección en la esfera unitaria equiárea; del concepto de orientación de

una línea a través de su acimut y cabeceo, orientación de un plano a través de su línea polar (v.gr. polo de un plano) y de su línea de máxima pendiente. Estos conceptos pueden repasarse en [17], que es un clásico en el tema y además que se publicó la traducción al castellano; o [12] que es un libro más reciente, muy didáctico y completo. También los anexos o capítulos específicos de [21] (Cap. 9), [7] (Ap. 5), [3] (Cap. 2), [23] (Cap. 5) y [9] (Sec. 2.3.1); resumen los fundamentos que se necesitan para este fin.

3.1. Software

Para usar el presente código se requiere de un lenguaje intérprete específico que lea el código desarrollado; este es el programa MATLAB®.

3.2. Código abierto

El objetivo principal del código abierto *geonetDigitizer* en su primera versión es el de permitir al usuario hacer una digitalización en pantalla de dos tipos de datos:

- lineamientos (presentados por puntos) y
- por planos presentados por sus polos únicamente.

En el caso de necesitar digitalizar planos a partir de sus trazas, se requiere de un otro grupo de cálculos que logren el ajuste de los puntos digitalizados a un arco de circunferencia para hallar el valor del polo del plano para el mínimo error. Esta capacidad puede ser incorporada en futuras versiones del programa.

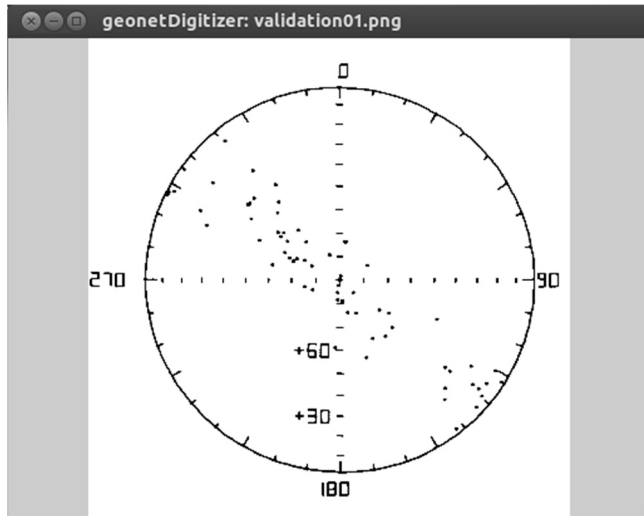
El código abierto está compuesto por 17 funciones que se ejecutan dentro del lenguaje de programación intérprete MATLAB®. Sin embargo, sólo una función, la llamada *geonetdigitizer*, es la que se ejecuta; y esta va llamando a las demás de forma interna. Para lograr la digitalización, el programa ejecuta a una sencilla interface gráfica, la cual va llamando internamente a las demás funciones y a las ventanas de diálogo, esto para lograr la digitalización y el guardar posteriormente la información generada (Fig. 1a).

El código abierto está almacenado en una carpeta llamada *geonetDigitizer* y puede descargarse según lo especificado en el Anexo B. El manual de uso puede descargarse también en ese sitio.

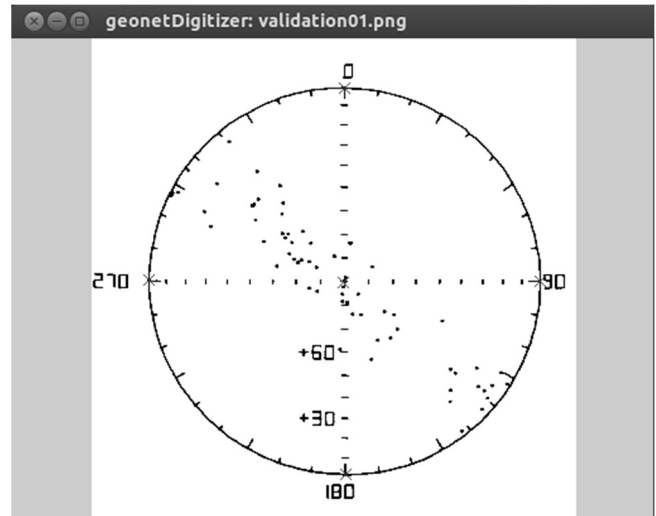
3.2. La imagen a digitalizar

La imagen que contiene la representación en proyección Wulff o Lambert tiene que ser obtenida de un proceso de rastreo digital (v.gr. conocido como *escaneado*, o *escaneo*), de una captura en pantalla de algún texto electrónico (v.gr. lo que se llama en la jerga común pantallazo o impresión de pantalla), o de una transformación de un texto electrónico a una imagen (v.gr. hacer un guardar como..., o un exportar a...).

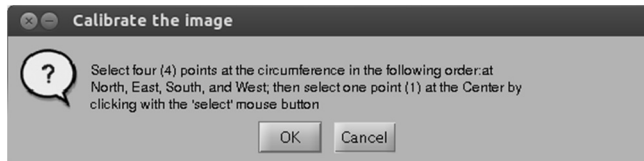
En resumen, antes de hacer cualquier cálculo se tiene que garantizar que el plano de la imagen a digitalizar no tiene deformaciones de alguna proyección distinta de la ortogonal (e.g. si fuera el caso de una imagen tomada con una lente de cámara y no escaneada) o deformaciones por ondulaciones del plano (si fuera el caso de una hoja escaneada pero que haya estado ondulada por humedad o por irradiación solar).



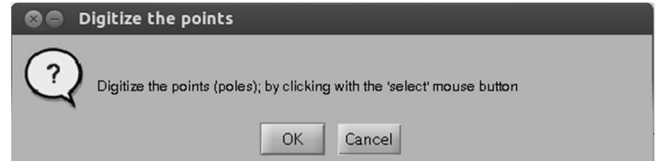
Estado de la ventana al iniciar el código.



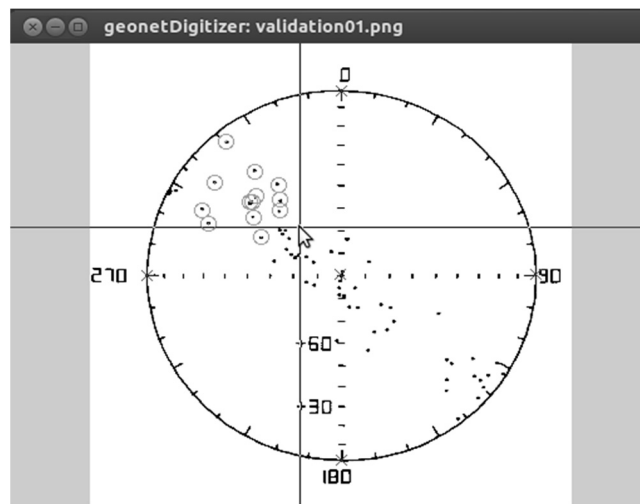
Estado de la ventana después de especificar los 4 primeros puntos de calibración.



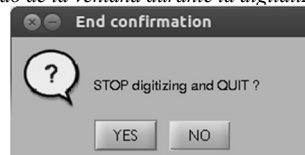
Caja de diálogo que indica qué puntos se usan para calibrar.



Caja de diálogo que indica el inicio de la digitalización.



Estado de la ventana durante la digitalización.



Caja de diálogo que pregunta si finaliza la digitalización.

Figura 1. Diferentes estados de la interfaz gráfica y cajas de diálogo para digitalizar la imagen validation01.png. Fuente: el autor.

4. Validaciones

4.1. Primera validación

En esta validación se tomó la figura 6.24 (página 180) del ejemplo 6.22 del libro de [6]; que aparte de tener la gráfica,

este tiene en su anexo B19 (página 300) los 64 valores que generaron dicha gráfica (que serán los valores reales). Lo mencionado se muestra también en la Fig. 1a. [6] hacen referencia al artículo de [18] de los cuales ellos extrajeron los datos originales. Los datos corresponden a medidas de la

orientación de los planos de bandeamiento de pliegues tipo F_1 en rocas turbiditas del Ordovícico, en Australia.

De este modo, si se digitaliza la imagen conociendo los valores reales es posible definir un error (e_1) para cada par de orientaciones (v.gr. la real y la medida), el error cuadrado total (e_{tot}), el error promedio (\bar{e}), y la desviación estándar del error (σ_e); todos ellos son los valores que se desea encontrar como referencia de los posibles errores inmersos que tiene el procedimiento que usa el código propuesto.

El error e_1 se define a partir de vectores unitarios, debido a que se quiere encontrar cuán distante está cada medida de su correspondiente real, en orientación tridimensional. De este modo, el error de cualquier vector v_i respecto otro de referencia u_i es

$$e_i = \frac{\text{acos}(v_i \cdot u_i)}{\pi} 100; \quad (8)$$

el error cuadrado total es

$$e_{tot}^2 = \sum_{i=1}^n e_i^2; \quad (9)$$

el error medio es

$$\bar{e} = \frac{1}{n} \sum_{i=1}^n e_i; \quad (10)$$

y su varianza (σ_e^2) es

$$\sigma_e^2 = (n - 1)^{-1} [e_{tot}^2 - n(\bar{e})^2]. \quad (11)$$

Siguiendo el procedimiento recomendado en el manual de usuario del programa que se describe (v.gr. geonetDigitizer), se logró obtener las 64 medidas. Luego, se compara cada medida digitalizada con su correspondiente real y se obtiene para todas las medidas el histograma de frecuencias de los errores relativos dados en porcentaje (Fig. 2). Se observa en esa figura, que cada par de datos tienen un error menor al menos 1% (v.gr. $e_{tot} < 0.01$), y que el error más frecuente está en el intervalo de [0,5%, 0,7%]. Asimismo, el e_{tot}^2 fue 25,40, \bar{e} fue 0,6154 y σ_e igual a 0,1356.

En el Anexo A se muestra el archivo de lotes para el intérprete de MATLAB© que se usó en la validación descrita en esta sección.

4.2. Segunda validación

En la segunda validación se empleó los datos tomados de la Fig. 1a (Pag. 804) del artículo de [27], donde en el Cuadro 2 de ese artículo se tiene las medidas de los 20 puntos que conforman la gráfica.

Los datos se graficaron en una proyección Lambert equiárea (Malla Schmidt) del hemisferio sur. Y en adición, la gráfica en sí tiene una particularidad: los puntos que se observan en la gráfica no son los polos de los planos de las estratificaciones, sino la orientación de la línea de máxima pendiente de esos planos dados por su acimut del buzamiento y el buzamiento; esta información fue leída del texto del mismo artículo.

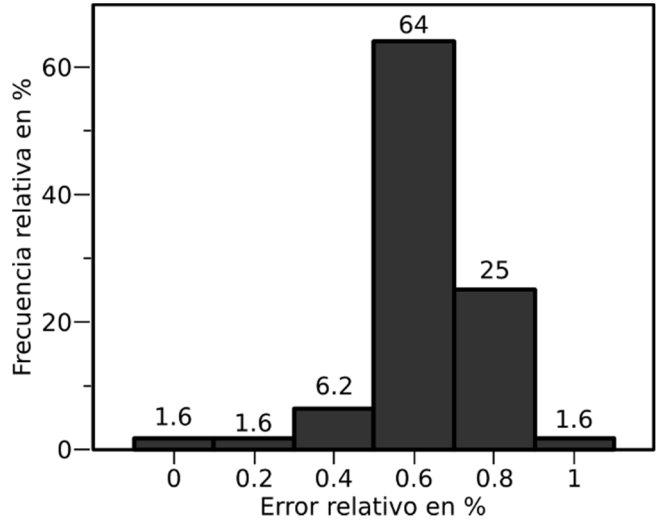


Figura 2. Histograma de los errores relativos encontrados en las 64 medidas. En el eje de abscisas se define el porcentaje del error, mientras que en el eje de ordenadas se define la frecuencia de los errores, también en porcentaje. Fuente: el autor.

Las medidas mencionadas corresponden a los planos de las camadas (*cross-beds*) de la muestra C_1 , planos que son parte de un cuerpo de arenisca del miembro Cathedral Bluffs del Eoceno de la formación Wasatch, localizado al sudoeste del condado Carbón en Wyoming (EE.UU.). Los datos fueron tomados por su respectivo autor para el mismo artículo.

Usando el mismo archivo de rutinas (*script*) de arriba, se obtuvo que el error más frecuente está en el intervalo de [0,25%, 0,75%], y que $e_{tot} = 11.96$, $\bar{e} = 0.6382$ y $\sigma_e = 0.4479$.

5. Conclusiones

Los datos que representan las orientaciones espaciales de ciertos elementos estructurales en el campo de las Ciencias de la Tierra, en su gran mayoría no están disponibles como un conjunto de datos numéricos; sino que están disponibles en forma de impresos o en el mejor de los casos en imágenes digitales bidimensionales provenientes de una de las dos proyecciones esféricas de uso extensivo en Geología Estructural y Geomecánica.

Todos estos datos se pueden recuperar con una herramienta de digitalización, para así convertirla de información analógica a digital y editable a través del programa geonetDigitizer. El programa es uno de código abierto que aplica conceptos de visión artificial a las Ciencias de la Tierra mediante un grupo de funciones de fácil acceso interpretados en MATLAB©, que pueden correrse en distintos sistemas operativos. Todas estas características hacen que el programa sea también potencialmente útil para evitar el fenómeno de la Era Digital Desconocida.

Las pruebas de validación y los ejemplos presentados en este artículo hacen que esta herramienta sea promisoría para los fines pensados.

Los siguientes pasos a seguir con el mencionado programa son de: detectar posibles rupturas del programa (v.gr. *crashes*) ante ciertos datos reales, y la corrección de gusanos (*bugs*); mejorar sus capacidades ante nuevos requerimientos de los usuarios; y crear un software de uso industrial que no se base en funciones para el intérprete MATLAB®, sino sea compilado para su ejecución.

6. Anexos

6.1. Archivo de lotes para la primera validación

El archivo de lotes para el intérprete de MATLAB® que se usó en la validación descrita en la sección 5.1 tiene el siguiente contenido.

```
% validation01.m
%% Calling the path of the toolbox
%% Real data
realDipdirDipArray = [ 185, 16; 154, 12; 145, 50;
141, 62; 141, 43; 140, 83; 138, 28; 136, 22; 136,
40; 133, 52; 156, 2; 130, 52; 129, 32; 129, 52;
127, 34; 127, 72; 125, 28; 125, 32; 125, 89; 124,
47; 118, 18; 116, 27; 116, 39; 116, 70; 118, 87;
115, 14; 115, 21; 113, 24; 112, 64; 111, 22; 102,
30; 80, 16; 71, 13; 32, 3; 20, 6; 14, 9; 6, 29;
351, 14; 343, 35; 338, 15; 333, 28; 323, 30; 323,
87; 320, 70; 320, 7; 317, 66; 314, 85; 314, 29;
311, 59; 311, 62; 310, 83; 309, 79; 309, 75; 309,
20; 306, 81; 306, 24; 304, 69; 303, 87; 301, 81;
297, 90; 296, 90; 293, 44; 239, 12; 1, 9 ];
realTrendPlungeArray = dipdir2polearray(
realDipdirDipArray );
%% Start to digitize
% Now start to digitize with function
'geonetdigitizer'
digitTrendPlungeArray = geonetdigitizer(
'000ComparisonRealDigit.png', ...
'equalarea', true, 'NonRigidAffine' );
digitRoundedTrendPlungeArray = round(
digitTrendPlungeArray );
%% Arrange data
arrangedRealTrendPlungeArray
=realTrendPlungeArray;
correspDigitTrendPlungeArray
=digitRoundedTrendPlungeArray;
sideBySideCompArray = [
arrangedRealTrendPlungeArray,
correspDigitTrendPlungeArray ];
%% Transform orientation data from trend/plunge to
unit vector
numData = size( sideBySideCompArray, 1 );
arrangedRealUnitVecArray = zeros(numData, 3);
correspDigitUnitVecArray = zeros(numData, 3);
for i=1 :numData
    arrangedRealUnitVecArray(i,:)
=trendplunge2unitvect(
arrangedRealTrendPlungeArray(i,:) );
end
for i=1 :numData
    correspDigitUnitVecArray(i,:)
=trendplunge2unitvect(
correspDigitTrendPlungeArray(i,:) );
end
%% Calculate the errors
error1Array = zeros(numData, 1);
for i=1 :numData
    v =correspDigitUnitVecArray(i,:);
    u =arrangedRealUnitVecArray(i,:);
    error1Array(i) =acos( dot(v, u) )/pi*100;
```

```
end
error2Array =zeros(numData, 1);
for i=1 :numData
    v =-correspDigitUnitVecArray(i,:);
    u =arrangedRealUnitVecArray(i,:);
    error2Array(i) =acos( dot(v, u) )/pi*100;
end
errorArray =min( [error1Array, error2Array], [],
2);
% the total square error  $e_{\mathrm{tot}}$ 
squareErrorTotal =sum(errorArray.^2);
% the mean error  $\bar{e}$ 
errorMean =sum(errorArray) /numData;
% the variance and standard deviation of the error
 $\sigma_{\mathrm{e}}$ 
% and  $\sigma_{\mathrm{e}}$ 
errorVariance =(squareErrorTotal -numData
*errorMean^2) / (numData -1);
errorStd =sqrt( errorVariance );
%% Calculate the relative frequency of the errors
distribution
xBins =linspace(0,1,6);
[ n, xout ] =hist(errorArray, xBins);
nRel =n/length(errorArray);
%% Plot the distribution of error in a histogram
bar( xout, nRel, 'w', 'BarWidth', 1);
hold on
textHeadings =size( nRel );
for i=1:length(xout),
    textHeadings(i) =text( xout(i), nRel(i),
num2str(nRel(i),2) );
end
% Use the handles TH to modify some properties
set(textHeadings,'Horizontalalignment','center',
'verticalalignment','bottom');
```

6.2. Ejemplo 1: digitalización de datos de estratificación

Este ejemplo es un caso de digitalización de datos históricos. Los datos del presente ejemplo fueron tomados de la Fig. 5a del artículo de [22]. Los datos tienen más de tres cuartos de siglo publicados, por tanto son datos históricos.

La gráfica está en una proyección estereográfica Wulff en el hemisferio sur, donde los puntos son los polos de los planos de estratificación tomados en un afloramiento en el cañón De Chelly, de la formación del mismo nombre, cerca de la localidad de Chin Lee en el estado de Arizona (EE.UU.).

El código que hace correr el programa para esta digitalización en particular es:

```
digitTrendPlungeArray =digitizestereoplots(
'Reiche1938Figure05a.png', 'equalarea' );
```

La Fig. 3 muestra la gráfica a partir de la cual se extrajo los 52 puntos que se muestra en el Cuadro A-1.

Los valores fueron transformados de (acimut, cabeceo) a (dirección de buzamiento, buzamiento) y luego redondeados al entero más cercano con las siguientes expresiones:

```
digitRoundedTrendPlungeArray =round(
digitTrendPlungeArray );
digitRoundedDipdirDipArray =dipdir2polearray(
digitRoundedTrendPlungeArray );
```

La gráfica puede obtenerse con la función `plotplaneorientationdatanum`, función creada para graficar los resultados:

```
plotplaneorientationdatanum(
digitRoundedTrendPlungeArray, [], 'equalarea',
'kx', 'b', true);
```

6.3. Ejemplo 2: digitalización de datos de un tensor de susceptibilidad magnética

Este ejemplo es un caso de digitalización de datos antiguos, tomada de una tesis doctoral de Geología Estructural del año 2004, v.gr. información antigua [31]; cuyos datos sufrieron DDA pero que con el *toolbox* propuesto se logra recuperar.

La anécdota es que sí se logró encontrar los datos originales en forma digital; pero el archivo estaba codificado por el programa que lo originó, que es un programa particular del equipo que hacía las medidas de los datos. Para poder abrir el archivo se hubiera tenido que viajar al sitio donde estaba el equipo (v.gr. Brasil), y ver si después de 10 años aún el equipo tiene el mismo software y hardware (o si otro equipo moderno lo reemplazó). En este caso la recuperación sería una tarea costosa o imposible.

La información que se deseaba recuperar se sacó finalmente después de escanear la figura impresa (Figura 23e) y luego la digitalización con el uso de la presente herramienta.

La figura (en este artículo es la Figura 4) corresponde a la medida de polos de dirección de susceptibilidad magnética, tomadas del Sitio 11 en la muestra B de un Granitoide Permo-Triásico de la Cordillera de Los Andes en la parte Norte del Ecuador [31]. Los datos están en una proyección estereográfica Schmidt hemisferio sur. Los cuadrados llenos pequeños son las direcciones principales mayores, los triángulos pequeños son las principales intermedias, y los círculos pequeños son los menores. Los signos grandes corresponden a los valores medios de las cinco medidas de cada uno, y las elipses alrededor de éstos corresponden a una desviación estándar de esos valores.

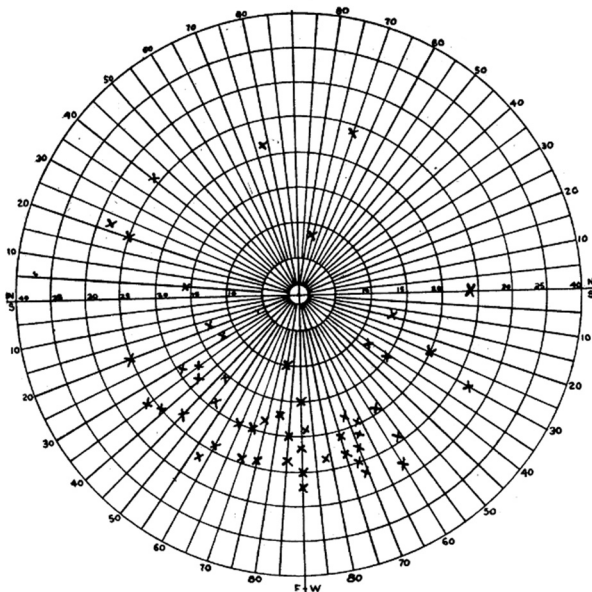


Figura 3. Imagen de medidas de orientaciones de planos de estratificación, datos históricos. Fuente: [22].

Cuadro A-1. Resultado de las orientaciones obtenidas del proceso de digitalización.

ζ_{dir}	ζ	ζ_{dir}	ζ	ζ_{dir}	ζ
199	51	059	40	337	42
168	46	051	38	339	45
193	18	043	33	345	44
130	55	055	56	340	49
112	60	051	53	340	57
110	53	046	49	342	53
095	33	033	57	345	49
270	51	030	52	352	50
283	28	039	40	000	58
295	42	026	42	360	53
300	58	020	42	005	50
306	25	016	38	000	46
306	31	010	35	359	40
073	27	000	31	005	42
063	25	327	40	015	51
011	21	327	52	021	52
070	54	330	59		
056	36	341	39		

ζ_{dir} : dirección de buzamiento (dip direction); ζ , buzamiento (dip). Fuente: el autor.

Aquí se tuvo el inconveniente de que la gráfica carecía de las marcas de los puntos cardinales principales, sin embargo como ella tenía texto, se logró encontrar de forma manual los píxeles que definen los puntos cardinales y el centro. Estos están en el siguiente orden: Norte, Este, Sur, Oeste y Centro;

$$\left\{ \begin{array}{l} (578, 319), (1080, 812), \dots \\ (578, 1309), (76, 812), (578, 812) \end{array} \right\}$$

El código que hace correr el programa para esta digitalización en particular es:

```
digitTrendPlungeArray =geonetdigitizer(
'VinascoVallejo2004Figure23e.png', 'equalarea',
true, 'NonRigidAffine', true );
```

Los Cuadros A-2A a A-2C muestran los puntos de medida recuperados con el uso de la presente herramienta; y el Cuadro A-3 muestra los puntos medios de esos datos, también obtenidos por digitalización.

Cuadro A-2. Resultado de las orientaciones principales del tensor de susceptibilidad magnética obtenidas del proceso de digitalización.

A: Vector mayor k_1 .

Dato	Ac.	Cab.
1	246	72
2	284	85
3	071	76
4	071	87
5	059	89

Ac., acimut (trend); Cab., cabeceo (plunge).

B: Vector mayor k_2 .

Dato	Ac.	Cab.
1	166	77
2	192	64
3	330	66
4	343	70
5	341	74

Ac., acimut (trend); Cab., cabeceo (plunge).

C: Vector mayor k_3 .

Dato	Ac.	Cab.
1	118	27
2	154	24
3	176	17
4	021	27
5	297	20

Ac., acimut (*trend*); Cab., cabeceo (*plunge*).

Fuente: el autor.

Cuadro A-3.

Las orientaciones principales-promedio del tensor de susceptibilidad magnética obtenidas también del proceso de digitalización.

Eje Principal	Ac. Promedio	Cab. Promedio
k_1	255	89
k_2	346	88
k_3	096	07

Ac., acimut (*trend*); Cab., cabeceo (*plunge*).

Fuente: el autor.

6.3. Alojamiento del programa

El presente código está alojado en el sitio anfitrión de proyectos informáticos de Matlab, llamado MatlabCentral.

<http://www.mathworks.com/matlabcentral/fileexchange/49817-geonetdigitizer>

A través de este sitio se hace las respectivas descargas, aportes y peticiones de participación en el proyecto.

6.4. Licencia

Copyright© 2015 en adelante, Universidad Nacional de Colombia y Ludger O. Suárez Burgoa.

Este código abierto es software libre: usted puede redistribuirlo y/o modificarlo bajo los términos de la Licencia BSD-2:

<http://opensource.org/licenses/bsd-license.php>

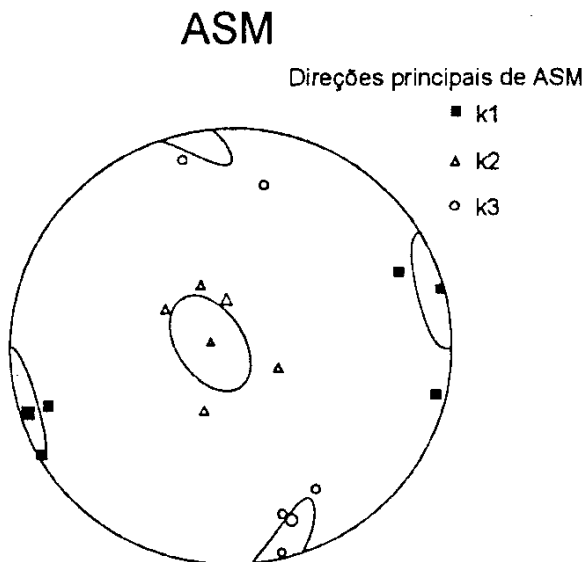


Figura 4. Imagen de medidas de orientaciones del tensor de susceptibilidad magnética, datos antiguos.

Fuente: [31].

6.5. Descargo de responsabilidades

El presente código computacional se distribuye con la esperanza de que sea útil, pero sin ninguna garantía; sin la garantía implícita en su comercialización o idoneidad para un propósito particular.

Referencias

- [1] Allmendinger, R.W., Cardozo, N., Fisher, D.M., et al., Structural geology algorithms: vectors and tensors (1 ed.). Cambridge: Cambridge University Press, 2012.
- [2] Bortoluzzi, G. and Ligi, M., DIGMAP: A computer program for accurate acquisition by digitizer of geographical coordinates from conformal projections, Computers and Geosciences, 12 (2), pp. 175-197, 1986. [http://dx.doi.org/10.1016/0098-3004\(86\)90005-1](http://dx.doi.org/10.1016/0098-3004(86)90005-1)
- [3] Brand, S., Escaping the digital dark age. Library Journal, 124 (2), pp. 46-49, 1999.
- [4] Chica-Sánchez, A., Aspectos mecánicos e hidráulicos de las rocas (1^{ra} ed.), Universidad Nacional de Colombia, Medellín, Colombia, 1979.
- [5] De Paor, D.G., Stereonet applications for Windows and Macintosh, Computer Methods in the Geosciences, Structural geology and personal computers, 15 (1), pp. 233-235, 1996.
- [6] Fisher, N.I., Lewis, T. and Embleton, B.J.J., Statistical analysis of spherical data (1 ed.). Cambridge: Cambridge University Press, 1993. 0-521-45699-1.
- [7] Goodman, R., Introduction to rock mechanics (2 ed.). New York: John Wiley & Sons, 1989. 0-471-81200-5.
- [8] Goshtasby, A., Piecewise linear mapping functions for image registration. Pattern Recognition, 19 (6), pp. 459-466, 1986. [http://dx.doi.org/10.1016/0031-3203\(86\)90044-0](http://dx.doi.org/10.1016/0031-3203(86)90044-0)
- [9] Groshong, R.H., Jr., 3-D structural geology: A practical guide to quantitative surface and subsurface map interpretation (2 ed.). Berlin: Springer, 2006. 3-540-31054-1.
- [10] Hahn, B.H. and Valentine, D.T., Essential MATLAB for engineers and scientists (4 ed.). Burlington, MA: Academic Press, 2010.
- [11] Hartley, R. and Zisserman, A. Multiple view geometry in computer vision. Geometría multivisual en visión computacional, 2 ed. Cambridge: Cambridge University Press, 2003
- [12] Lisle, R.J. and Leyshon, P.R., Stereographic projection techniques for geologists and civil engineers (2 ed.). London: Cambridge University Press, 2004. 0521828902. <http://dx.doi.org/10.1017/CBO9781139171366>
- [13] Loudon, T.V., Computer analysis of orientation data in structural geology, Technical Report No. 13 Northwestern University - Office of Naval Research, Geography Branch, 1964, 13 P.
- [14] Loudon, T.V., Wheeler, J.F. and Andrew, K.P., A computer system for handling digitized line data from geological maps, Computers and Geosciences, 6, pp. 299-308, 1980. [http://dx.doi.org/10.1016/0098-3004\(80\)90036-9](http://dx.doi.org/10.1016/0098-3004(80)90036-9)
- [15] Loudon, T.V., Wheeler, J.F., Andrew, K.P., et al., Affine transformation for digitized spatial data in geology. Computers and Geosciences, 6, pp. 397-412, 1980. [http://dx.doi.org/10.1016/0098-3004\(80\)90015-1](http://dx.doi.org/10.1016/0098-3004(80)90015-1)
- [16] Pereira-Leite, E. (Ed.), Scientific and engineering applications using MATLAB (1 ed.). Rijeka, Croatia: InTech, 2011. <http://dx.doi.org/10.5772/1531>
- [17] Phillips, F.C., La aplicación de la proyección estereográfica en geología estructural (2 ed.). Rosario: Editorial Blume, 1975.
- [18] Powell, C. McA., Cole, J.P. and Cudahy, T.J., Megakinking in the Lachlan Fold belt. Journal of Structural Geology, 7, pp. 281-300, 1985. [http://dx.doi.org/10.1016/0191-8141\(85\)90036-7](http://dx.doi.org/10.1016/0191-8141(85)90036-7)
- [19] Priest, S.D., Hemispherical projection methods in rock mechanics (1^{ra} ed.). London: George Allen & Unwin, 1985. 0046220070.
- [20] Priest, S.D., Discontinuity analysis for Rock Engineering. London: Chapman & Hall, 1993. 0-412-47600-2.
- [21] Ramsay, J.G. and Huber, M.I., The techniques of modern structural geology (1 ed.). London: Academic Press. Strain Analysis, 1, 1983. 0-12-576901-6.
- [22] Reiche, P., An analysis of cross-lamination: The Coconino sandstone. Journal of Geology, 46 (7), pp. 905-932, 1938.

- <http://dx.doi.org/10.1086/624709>
- [23] Rowland, S.M., Duebendorfer, E.M. and Schiefelbein, I.M., A laboratory course in structural geology (3 ed.). Malden: Blackwell Publishing, 2007. 1-4051-1652-8.
- [24] Sánchez-Villar, N.F., Desarrollo de patrones de fracturamiento y mecanismos de deformación del anticlinal de Monterralo, Pie de Monte Llanero, Cordillera Oriental de Colombia, Tesis, Departamento de Geociencias, Universidad Nacional de Colombia, Bogotá, Colombia. 2011.
- [25] Schuenemeyer, J.H., Koch, G.S. and Link, R.F., Computer program to analyze directional data, based on the methods of Fisher and Watson. *Mathematical Geology*, 49A (3), pp. 177-202, 1972. <http://dx.doi.org/10.1007/BF02311717>
- [26] Starkey, J. and Paulitsch, P., A computer programme to construct spherical projections, *Proceedings of the International Symposium: Experimental and natural rock deformation*, Darmstadt, Springer. pp. 38-74, 1970.
- [27] Steinmetz, R., Analysis of vectorial data. *Journal of Sedimentary Petrology*, 32 (4), pp. 801-812, 1962.
- [28] Thibault, P.A. and Klink, K., Analog-to-digital conversion of circular char data. *Computers and Geosciences*, 23 (3), pp. 329-332, 1997. [http://dx.doi.org/10.1016/S0098-3004\(96\)00093-3](http://dx.doi.org/10.1016/S0098-3004(96)00093-3)
- [29] Trauth, M.H., *MATLAB recipes for earth sciences* (2 ed.). Heidelberg: Springer Verlag, 2007.
- [30] van Everdingen, D.A., van Gool, J.A.M. and Vissers, R.L.M., Quickplot: A microcomputer based program for processing of orientation data. *Computers and Geosciences*, 18 (2-3), pp. 183-287. 1992. [http://dx.doi.org/10.1016/0098-3004\(92\)90081-2](http://dx.doi.org/10.1016/0098-3004(92)90081-2)
- [31] Vinazco-Vallejo, C.J., *Evolução crustal e histórica tectônica dos granitoides permo-triásicos dos Andes do Norte*, Tesis, Instituto de Geociências, Universidade de São Paulo, São Paulo, Brasil, 2004.
- [32] Xu, Y. and Xu, T., An interactive program on digitizing historical seismograms. *Computers and Geosciences*, 63, pp. 88-95, 2014. <http://dx.doi.org/10.1016/j.cageo.2013.11.001>

L.O. Suarez-Burgoa, es graduado como Ingeniero Civil en 2001, de la Universidad Mayor de San Andrés, La Paz, Bolivia, MSc. en Ingeniería Geotécnica en 2008 de la Universidad Nacional de Colombia, Medellín, Colombia y Dr. en Geotecnia en 2012 de la Universidad de Brasília, Brasil. En la actualidad es Profesor Asistente en dedicación exclusiva del Departamento de Ingeniería Civil de la Facultad de Minas de la Universidad Nacional de Colombia, Medellín, Colombia. Ha desarrollado actividades como consultor e investigador por más de diez años en el campo de la geotecnia civil y minera en Bolivia, Chile, Brasil y Colombia. Actualmente dirige el Grupo de Investigación de Geología Matemática y Computacional y es miembro de la Asociación Internacional de Geociencias Matemáticas.