RESEARCH ARTICLE

Getting Around the Black Box: Teaching (Geophysical) Data Processing through GIS

Kenneth Kvamme

It is argued that changes need to be made in how students are trained in contemporary computing environments. Focus is given to geophysical archaeology where knowledge of data processing is as important as understanding instrument handling in the field. While powerful and expensive commercial software programs are able to meet processing needs, it is suggested they are poorly suited for student training. In these icon-driven environments complex algorithms may be activated with a single click that requires little or no knowledge of what is actually done to the data yielding a “black box” approach to data processing. This situation contrasts with the days of pre-Windows computing when students often had to write their own code in a high-level computer language, forcing them to learn the intimate details of data processing. Such an approach is generally unacceptable today and may actually deter students from further study, yet an alternative software solution exists that can fulfill training needs. Graphically-driven modeling tools in GIS environments are available that offer diverse functionality, are easily learned, and permit most algorithms commonly applied in geophysical data processing to be constructed. Use of these tools requires complete understanding of necessary operations, their sequences and consequences. The end result is equivalent to a flowchart that shows in icon form all GIS layers input, output, and generated during intermediate steps, the various tools applied to manipulate the data, plus the full sequence of operations. In short, it is possible to teach students the art of data processing in an easy to learn and use graphical environment while imparting true knowledge of algorithms required to achieve desired outcomes. To substantiate these claims, case studies illustrate the processing of magnetic gradiometry and electrical resistance data from lateral surveys, GPR profiles in vertical investigations, and the generation of GPR time-slice maps.

Keywords: data processing; student training; geophysics; GIS graphical programming; TerrSet Macro Modeler

Introduction

This paper confronts a general problem in student training, but in a specific domain. Archaeologists today are rightly enamored with technology and its many benefits that have permitted advances in data acquisition and analysis. This is particularly true in archaeological remote sensing, including ground-based geophysics, with the rise of sub-meter resolution satellites, LiDAR and drone surveys, vehicle-driven carts holding arrays of geophysical sensors, and the like, all coming to fruition within the past decade or so. Yet, it is easy to overlook certain downsides that may arise as technology advances. With the advent of the electronic calculator in the 1970s math skills may have regressed and the same might be said, more recently, of map reading skills caused by reliance on global navigation satellite systems (GNSS). Similar issues exist with modern computer software. In recent decades we have moved to

icon-driven point-and-click environments where, with a single click of a mouse, immensely complicated procedures and algorithms may be applied to our data. For most students these environments offer a “black box” approach to computing and problem solving, where it is difficult to learn and understand exactly what the computer is doing. True, the ambitious student may consult help files when they exist, but explanations frequently are cryptic, terse, unsatisfactorily incomplete, and rarely will one pursue primary sources describing algorithms in references cited. In general, it is difficult for the student to realize exactly what each click causes to be done to the data.

This situation is undesirable in learning environments. It is all too easy for students to click on icon-after-icon with little or no understanding of what they are doing, while powerful, complex, and linked chains of events can occur with each click. Even when the instructor, or a help file, explains that “normalizing transects means forcing their average values to zero” (a procedure common to the processing of magnetic gradiometry data), many students may not completely understand or care to

understand the nature of the required processing steps, their consequences, and why they are performed. This situation contrasts with the days of pre-Windows computing when students often had to write their own code in FORTRAN or some other high-level language. While certainly filled with its own drawbacks (this requirement itself deterred or eliminated many from further study), students nevertheless learned the intimate details of data processing because they had to code them from the ground up themselves.

Recent advances in Geographical Information Systems (GIS) software may offer a solution to this quandary. GIS are collections of software tools for the handling, manipulation, processing, and display of spatially distributed information (Conolly & Lake 2006). They are therefore ideal for application to geophysical data, all of which are spatial in character, our focus here. They have been routinely employed for organizing and visualizing geophysical data and merging results with other information (Neubauer 2004; Gallo et al. 2009; Nowaczinski et al. 2015). Many GIS offer a highly varied suite of complex spatial data handling and processing tools that permit a wide variety of manipulations. Significantly, many GIS offer advanced modeling tools that permit, through a Graphical User Interface (GUI), the chaining together of many tools (forming models) that are applied in a fixed sequence to a variety of data inputs in order to achieve a desired outcome. These models may then be named, stored, sent to other users, and applied to other data sets. The important points are that (1) a series of complex data manipulations and operations may be applied to spatial data sets to achieve desired outcomes, (2) the building of a model *requires* complete understanding of the necessary algorithms, operations, their sequences and consequences, and (3) the construction of a model does not require knowledge of complex and difficult-to-learn computer scripts, languages, or codes (as would be required using common programming solutions found in the R language (R 2018), Python (Python Software Foundation 2018) or Matlab™ (MathWorks 2018). Rather, models are constructed through “drag and drop” operations performed in a GUI environment—users simply grab icons representing data sets (GIS layers) and link them through arrows and other symbols with other icons that represent various GIS tools. The end result is a graphic that is equivalent to a flowchart which shows in icon form all GIS data layers that are input, output, and generated during intermediate steps, the various tools applied to manipulate the data, plus the full sequence of operations. In short, it is possible to teach students the art of data processing through an easy to learn and use GUI environment while imparting true knowledge of the algorithms and processing steps required to achieve desired outcomes.

If GIS modeling tools can be employed to successfully and adequately process geophysical data, then a second pragmatic benefit may also be realized. Professional geophysical software commonly utilized in archaeological applications, such as Geoplot™ (Geoscan Research 2018), TerraSurveyor™ (DW Consulting 2018), or GPR-Slice™ (GPR-Slice & GPRSIM 2018) is expensive. Students and beginning professionals may not be able to afford

costly programs yet wish to work with geophysical data. While a number of geophysical software offerings are freely available, such as Snuffler (SussexArch 2018), the Archaeological Geophysics Toolbox (a series of Python plugins for QGIS at QGIS 2018), and matGPR (MATGPR 2018), each offers limited functionality and is designed for specific kinds of geophysical data (e.g., GPR). A low-cost alternative that offers suitable processing capabilities for a wide range of geophysical data types may be welcome to students and the professional community alike.

My position in this paper is that changes should be made in how students are trained in contemporary computing environments. Knowledge of data processing in geophysical archaeology is as important as knowledge of instrument handling and survey planning in the field. While powerful, expensive, and state-of-the-art software is certainly able to meet processing needs, these programs are poorly suited for student training because complex algorithms are activated with a simple click that requires little or no knowledge of what is actually done to the data. Self-generation of computer code, or its equivalent, is necessary for learning. Although some students are adept at coding or scripting languages or are willing to learn, more of them are not, yet a solution exists that can fulfill training needs in both camps. That solution lies in the use of graphically-driven modeling tools in GIS environments that offer diverse functionality, are easily learned, and permit most algorithms commonly applied in geophysical data processing to be constructed. These are bold statements that I believe must be substantiated and demonstrated.

In the following sections elements of GIS technology are presented as vehicles for learning how to process geophysical data in a GUI environment. Case studies illustrate basic processing steps applied to magnetic gradiometry and electrical resistance data from lateral surveys, the analysis and processing of GPR profiles in vertical investigations, and the generation of GPR time-slices resulting from surveys of areas. Each presents limited data sets acquired by students, some of which contain operator, instrument, or other defects in order to illustrate how GIS modeling tools may be used to correct deficiencies in generalized data processing workflows. It is emphasized that the following is not intended as a comprehensive geophysical data processing system. Rather, it is meant to offer a few basic operations primarily for instructional purposes. Yet, as shown below, the software performs many necessary tasks making it suitable for most processing needs.

The TerrSet GIS

Except for down-hole surveys or point measurements nearly all geophysical data collected in the field occur in a *raster* format, a matrix of numbers composed of rows and columns of systematically sampled data. Nearly all GIS routinely handle raster data as “layers” and many offer advanced modeling tools in GUI environments. While several conceivably might function in the capacity advocated here, I employ the TerrSet system, developed by Clark Labs within the Graduate School of Geography at Clark University (Clark Labs 2018), which is available to university researchers at moderate cost and to students at low cost.

TerrSet includes the full Idrisi GIS and image processing suite, plus six next-generation subsystems for modeling earth trends, land and climate change, biodiversity, habitat, and other domains. The GIS contains over 300 tools or modules for data processing that drive the foregoing and which are probably unrivaled in their diversity. Many are unique and unconventional, making geophysical data processing a real possibility. These modules are divided into subsystems that include spatial data input, database query, mathematical, distance, and context operations, decision support, image processing, statistical analysis, and cartographic display.

TerrSet offers two generic modeling tools, the *Idrisi Macro Language* (IML), a command line system similar to old DOS-style batch files, and the *Macro Modeler*. While IML offers a number of advantages, the Macro Modeler is friendlier owing to its drag-and-drop GUI interface which is easier to learn, and it precludes the difficulties of mastering a scripting language.

Macro Modeler

TerrSet's Macro Modeler includes a suite of icons that represent individual raster and vector layers, raster collections (multiple related raster layers of the same data type and dimension), GIS modules, user-created sub-models (models previously created that act as modules), connectors (arrows or pointers that link data layers to modules

and vice versa), "dynalinks" (a link that causes a looping operation, permitting iterative operations), and "dynagroups" (raster collections where every member is sequentially processed) (**Figure 1a**). Although a single outcome may be desired from a GIS model, numerous intermediate and temporary data sets often are generated. These layers are named *tmp00x* in the following, where *x* is a sequential number generated by the model. They may be automatically deleted after a model is run. In the following, all TerrSet modules and generated models are given in upper case while GIS layer names are presented in italics.

Geophysical Data Organization and Input

The following case studies begin with lateral or area surveys, which are conducted in the horizontal plane. In geophysical surveys, instruments are commonly passed over the landscape in a series of parallel transects, typically separated by 0.25–1 m, and in each transect anywhere from 1–50 measurements might be sampled in each linear meter (Gaffney & Gater 2003). In this manner a region is systematically sampled to create a raster of rows and columns, often with rectangular pixels owing to unequal sampling densities along versus between transects. Although cart systems with arrays of multiple sensors are being employed more frequently (Linford, Linford & Payne 2015; Schneidhofer et al. 2016) and instrument position is increasingly determined by GNSS, most surveys still are

Figure 1: Macro-modeling tools applied to magnetic gradiometry data: **a)** the tool set, **b)** example data blocks, **c)** Model DESTRIPE, **d)** raw block *M1*, **e)** average magnetic values per row, **f)** results of subtraction, *d–e*.

conducted manually in small *blocks* (sometimes referred to as “grids”). Survey blocks permit piecemeal coverage of a region and generally measure 20×20 m, but range from 10×10 m to perhaps 30×30 m. Within them instrument position is controlled by meter-marked tapes placed as transect guides on the ground. Owing to instrument characteristics, changes in environmental conditions during surveys, and operator errors, a number of processing steps have evolved for treating each type of geophysical data in order to remove common data defects. Detailed information about geophysical data processing can be found in Ciminale and Loddo (2001), Conyers (2013), Gaffney & Gater (2003), and Kvamme (2006).

Data Input

Input data files organized as rasters should each represent a survey block. Their input may be handled in two ways. ASCII or text files may be read using SSTIDRIS after removal of header information, if any. Binary files are input using GENERICASTER, a highly flexible module that permits importing of a wide range of binary formats. Typically, a binary file is organized in a “band sequential” format. If header information is included, the number of bytes it occupies must be specified, and the type and size of the data must be indicated (4, 8, 16 or 32-bit signed or unsigned integer or single or double precision floating point). In either module the number of rows and columns of the raster must be specified, the real world coordinate ranges on both axes (i.e., minimum and maximum x - and y -axis coordinates), and a geographic reference system (typically “plane” for a standard x - y Cartesian coordinate grid). Accurate coordinate ranges are critical for the correct placement of multiple survey blocks within survey-wide composite data sets. Output is a TerrSet format raster layer.

In instances where survey data are not recorded in blocks of parallel transects that form a natural raster, but where position is recorded by GNSS in transects that may not be perfectly parallel, other TerrSet tools may be employed to generate a uniform raster for subsequent processing. Each geophysical reading or measurement must first be associated with its corresponding x and y spatial coordinate in a text file organized in a space or comma delimited “xyz” format (where z is the geophysical measurement). These data are then brought into TerrSet as a point vector file through the XYZIDRIS module. The user then has the option to generate a raster from these data, whose spatial resolution and coordinate ranges must be specified, utilizing inverse distance weighting, kriging, or triangulated irregular network interpolations (using the INTERPOL, Kriging and Simulation, and TIN modules, respectively).

Case Study 1: Processing Magnetic Gradiometry Data

Magnetic gradiometry is probably the work horse of archaeological geophysics, owing to generally insightful results, speed of survey, high data density, ease of operation, and low data processing requirements. This passive method of prospecting quantifies magnetic variations in *nanoTeslas* (nT), perhaps to a depth of 1.5 m. Many of

those variations are anthropogenic, caused by intense firing (hearths, kilns, burned buildings). Moreover, topsoil tends to be magnetically enriched owing to a number of processes. Consequently, movement of topsoil through mounding, ditch digging, and other constructions tends to generate anomalous indications. Further details about magnetic prospecting may be found in Aspinall, Gaffney, and Schmidt (2008).

A magnetic gradiometry (MG) data set from the Double Ditch site, in North Dakota, is employed to illustrate a GIS-based data processing sequence. This site is a large prehistoric village that dates between CE 1490–1780. Full geophysical results have been reported that describe nearly 300 blocks (20×20 m) of MG survey (Kvamme & Ahler 2007). Here, only six of those blocks (named *M1–M6*) that illustrate village fortification ditches, bastions, and marked survey defects are utilized. They were acquired by students in 2002 using a Geoscan Research FM-36 fluxgate magnetic gradiometer. In each block, 40 transects in an east-west (horizontal) direction with four samples per meter were acquired, forming rasters of dimensions 80×40 . The six blocks are illustrated in **Figure 1b** along with a map showing their arrangement in a local coordinate space. These data suffer from several defects that include striping, which results from “heading” errors (a slight change in magnetic magnitude as the operator changes direction in zigzag surveys), drift (a change through time of magnetic magnitude common to older instruments), and staggering (a “herringbone” effect along linear anomalies caused by operator timing errors; Kvamme 2006). Furthermore, in block *M2* the operator took a lunch break in the middle of the survey during which time the instrument underwent considerable drift (**Figure 1b**).

In MG data processing one and possibly two “block operations” are commonly required depending on the nature of defects (block operations are performed on survey blocks individually). The processed blocks are then concatenated in correct spatial position to form a composite data set. Subsequent global operations (applied to all blocks simultaneously) are typically in the form of enhancements (e.g., low-pass filters, interpolation), discussed in a section below. For brevity, block operations are illustrated using only *M1*.

De-striping

The correction of a striping defect frequently puzzles students, until they see how easy it is to correct. The approach taken here merely computes the mean of each transect (row) and subtracts it from the initial data set, thereby yielding rows containing deviations from the mean and centered about zero (which is close to the theoretical mean of MG data; see Ciminale & Loddo 2001). In **Step 1**, DESTRIPE accepts raster layer *M1* (**Figure 1d**) and applies the CONTRACT module, which contracts the x -axis by a factor of 80 (the number of columns in the raster, creating a single column) and the y -axis by a factor of 1 (i.e., no contraction, retaining the 40 rows), employing the “aggregation” method which computes the mean of each row in *tmp001* (**Figure 1c**). In **Step 2**, EXPAND is applied to the result to enlarges it by a factor of 80, re-generating the

correct number of columns, but far too many rows (i.e., 3,200) in *tmp002*. **Step 3** therefore applies CONTRACT again, but with a 1× contraction on the x-axis (retaining 80 columns) and an 80× contraction on the y-axis (using the “thinning” option, keeping every 80th row), to produce 40 rows. The result, in *tmp003*, is an 80c × 40r matrix (identical to M1) holding the mean of each row (**Figure 1e**). Map algebra is applied in **Step 4**, using OVERLAY, where *tmp003* is subtracted from M1 (**Figure 1c**), effectively removing the striping (**Figure 1f**). *Note that this algorithm also corrects drift, but that staggering is still present.*

De-staggering

With measurements acquired every quarter-meter along transects there appears to be about a half-meter of stagger (**Figure 1f**), which can be corrected if each even numbered row is shifted one column left and each odd row one column to the right (rows are numbered from the top beginning with even row 0). Students are often stymied by how a raster may be shifted left or right. An easy solution is found through a simple filtering operation. In **Step 1** (**Figure 2a**) custom filter *Left1* (**Table 1**) is applied to M1 via the FILTER module, which causes the measurement to the right to replace the value in each cell (module FILTER permits construction of custom filters). The result in *tmp001* causes

the data to be shifted one column to the left (the right-most column of each row is a duplicate of the second from the right). **Step 2** then multiplies the result, through OVERLAY, by a custom template held in layer *EVEN80x40* which holds ones in even rows and zeros in odd rows, causing the left-shifted data to exist only in even rows (*tmp002* in **Figure 2b**). **Step 3** (**Figure 2a**) again employs FILTER, but with custom filter *Right1* (**Table 1**) applied to M1, which causes all data to be shifted one column to the right. **Step 4** multiplies this result in *tmp003*, through OVERLAY, by a custom template held in layer *ODD80x40* which holds ones in odd rows and zeros in even rows, causing the right-shifted data to exist only in odd rows (*tmp004* in **Figure 2c**). Finally, **Step 5** utilizes OVERLAY to sum *tmp002* and *tmp004*, yielding the corrected data (*tmp005* in **Figure 2d**). Obviously, shifts of greater magnitude may be achieved by modifying the filters shown in **Table 1**.

Table 1: Organization of 3 × 3 filters “Left1” and “Right1.”

Left1	Right1
0 0 0	0 0 0
0 0 1	1 0 0
0 0 0	0 0 0

Figure 2: Macro-modeling tools applied to magnetic gradiometry data: **a)** Model DESTAGGER, **b)** DESTAGGER result for even rows, **c)** DESTAGGER result for odd rows, **d)** combination of *c* and *d*, **e)** complete MAGMODEL for full processing, **f)** Model COMPOSITE, **g)** generic COMPOSMODEL for compositing of unlimited data blocks.

Full model & application to all blocks

Both of the previous user-defined models (**Figures 1a and 2a**) were applied to a single input layer, *M1*, for purposes of example. Real application comes when a model, or series of models, are applied to a group of related layers using the dyna-group function. To do so, all related raster layers, here *M1–M6* (which must have the same dimension—numbers of rows and columns), are first saved in a “Raster Group File” or RGF (using TerrSet’s “Collection Editor”). Then, each model is saved as a “SubModel,” permitting them to be applied in the same way as TerrSet modules. When a RGF is input to a model, the output is also a RGF with the same number of layers as the input, named as specified in the model (with a numeric suffix paralleling the order of layers in the input RGF). A “final” magnetic gradiometry processing sequence might then accept an RGF as input, process that group through DESTRIPE, and finally subject that result through DESTAGGER, as depicted in MAGMODEL in **Figure 2e**. The output, with prefix “procMAG” (**Figure 2e**), will generate six layers, *procMAG_1–procMAG_6*, each with major defects removed.

Creating a survey-wide composite

To generate a composite layer holding all six processed data sets in correct spatial position requires use of the model COMPOSITE (**Figure 2f**). It requires a “blank” layer (here named *Blank*) composed entirely of zeros of the correct data type (real or integer) that contains (1) the number of rows and columns and (2) the full coordinate system and ranges of the ultimate composite (which will match the coordinate ranges of the blocks defined during data input). Blank layers of any type and size may be created with TerrSet’s INITIAL module. Model COMPOSITE will correctly place a single magnetic block, here *M1*, within the region of the layer *Blank*. However, when saved as a sub-model (subCOMPOS), a dyna-group such as the fully processed RGF *procMAG* (generated by MAGMODEL, **Figure 2e**), may be input as a substitute for *M1*, as shown in COMPOSMODEL (**Figure 2g**). This model will recognize the number of layers in the RGF and iteratively, through a dyna-link (red arrow), overwrite *Blank* with each member of the group in its proper position (**Figure 2h**). COMPOSMODEL is a generic tool for compositing any form of geophysical data.

Case Study 2: Processing Electrical Resistance Data

Electrical resistance surveys are an active form of prospecting caused by the injection of a current into the ground and the recording of resistance to that current, in *ohms*. Earth resistance largely depends on variations in ground moisture and the density of subsurface materials. Resistance measurements also are a function of electrode configurations. Many practitioners therefore convert to *apparent electrical resistivity*, measured in *ohm-meters*, which is a property of earth conditions alone and differs from ohms only by a constant determined by the nature of the electrode array. Many arrays are utilized, but for lateral surveys the twin-probe array is dominant because the data facilitate interpretation, survey speed is improved because

only two electrodes need to be placed for a measurement (usually affixed in a rigid frame), with two others stationary in a remote place and connected by a long cable. The separation distance between the two roving electrodes controls prospecting depth. Further details about this form of prospecting may be found in Schmidt (2013).

An electrical resistance data set from Fort Pierre-Chateau (FPC) is utilized here to illustrate GIS-based processing of electrical resistance data from lateral surveys. This fort, more properly a trading post, was an important center of the American fur trade from 1832–1854, after which it actually did serve as a military fort until 1857 (Schuler 1990). Six 20 × 20 m blocks of data are employed from much larger surveys conducted in 2007 and 2012 (Patton 2013). The data were acquired with a Geoscan Research RM-15 configured as a twin-probe array with electrode separation of 0.5 m. Data were sampled every half-meter along both coordinate axes, yielding blocks of dimension 40r × 40c.

Two operations are minimally performed when processing electrical resistance data. One is “de-spiking,” which attempts to remove data outliers (extreme values) generally caused by poor probe placements (e.g., on non-conductive rock immediately below the surface). The other is “edge-matching,” which attempts to correct imbalances in mean resistance values between adjacent survey blocks. They frequently occur when ground conductivity changes between surveys due to rainfall, when surveys are made at different times under altered ground moisture conditions, or when electrode configurations are changed (Gaffney & Gater 2003). Raw data from FPC illustrate a number of data spikes, seen as a “salt and pepper” effect (particularly in block *R6*), and major edge imbalances are evident between many blocks, requiring edge-matching of mean values (**Figure 3a**).

De-spiking

Model DESPIKE removes isolated extreme measurements, frequently referred to as “data spikes.” Students are often too liberal in their de-spiking efforts, not only removing unwanted extreme values, but frequently archaeological significant measurements as well, so care should be taken. DESPIKE accepts an input layer to which the FILTER module is applied (**Figure 3b**). FILTER is set as an “adaptive box filter,” which compares a measurement to the average of neighborhood measurements; if it is more than x standard deviations deviant from the neighborhood mean it is an outlier and replaced with that mean. As resistance data spikes are typically isolated, composed of a single measurement, a small 3 × 3 neighborhood containing eight neighbors is usually adequate. Extreme values might be defined as lying more than two standard deviations away from neighborhood means (the analyst must experiment to seek a suitable value). Before and after results for layer *R6* are illustrated in **Figure 3c**, as is their difference achieved through map algebra. DESPIKE may be applied as a block operation to a RGF (after saving as a sub-model) as illustrated in **Figure 3d**, or globally to a composite of many data blocks.

Figure 3: Macro-modeling tools applied to electrical resistance data: **a)** example data blocks illustrating defects, **b)** Model DESPIKE, **c)** input, output, and their difference from DESPIKE, **d)** DESPIKEMODEL for application of DESPIKE to a complete raster group file, **e)** Model EDMATCH-RL, **f)** full data composite after complete processing, **g)** enhancement of **f** after application of high-pass filter.

DESPIKE may also be applied to MG data in efforts to remove extreme measurements generated by recently deposited iron rubbish (nails, etc.) that cause robust dipolar anomalies (a pairing of high positive and low negative measurements). It may also be regarded as a general enhancement tool to clean up “noise” caused by data outliers (see below).

Edge-matching

The many block edge discontinuities seen in **Figure 3a** may be corrected by finding the mean difference between edge rows (or columns) in adjacent blocks and adding the positive difference to the full block whose row (or column) mean is lowest. This is easy for students to grasp, but how to accomplish it is not because a convoluted series of steps are required. In this context focus is on a block to which an adjacent block is matched. Consequently, there may be up to four possible neighbors. While technically four distinct models are required, all vary in minor ways depending on the direction of comparison. **Figure 3e** depicts EDMATCH-RL for the right-to-left comparison. In **Step 1**, WINDOW extracts the *right*-most column of *R1* into *tmp001*, and in **Step 2** it extracts the *left*-most column of *R2* into *tmp002*. **Step 3** computes the difference between these

columns (*tmp001*–*tmp002*) through OVERLAY on a cell-by-cell basis into *tmp003*, and **Step 4** employs CONTRACT with the aggregation option to compute the overall mean difference and generate a layer composed of a single row and column (i.e., a one cell) in *tmp004*. **Step 5** then applies EXPAND to “reconstitute” the dimensionality of this layer by 40× (i.e., to 40r × 40c) in *tmp005*, which permits the mean difference to be added, via OVERLAY, to the second input file, *R2*, producing an edge-corrected result, seen in the lower left two blocks of **Figure 3f**.

Obviously, edge-matching operations must be applied judgmentally and as needed to survey blocks one pair at a time until all apparent discontinuities are removed. The full data set from FPC was subjected as a RGF to DESPIKEMODEL (**Figure 3d**) followed by judgmental use of EDMATCH-RL for right-to-left comparisons and EDMATCH-TB for top-to-bottom comparisons (not illustrated). The result, placed in a RGF, was then submitted to COMPOSMODEL (**Figure 2g**) to generate a full and corrected composite which shows a palisade wall of the fort as well as a number of storage rooms and floors, illustrated in **Figure 3f**. Obviously, this electrical resistance surface is quite dynamic, calling for further enhancements to improve its visualization (undertaken in a section below).

Case Study 3: Processing GPR Profiles

GPR surveys transmit microwaves into the ground from an antenna positioned at a given locus on the surface. A receiver records the two-way travel time (TWTT), in *nano-seconds* (10^{-9} seconds), of reflected energy and its amplitude, with travel time a proxy for depth below surface. Reflections are caused by discontinuities in the dielectric properties of subsurface materials, with large reflections indicating significant changes in the dielectric constant (which quantifies the ability of a material to hold an electrical charge). The recorded waveform is referred to as a “trace” which is quantized into many distinct measurements of amplitude referred to as “samples.” As the antenna is moved along a transect numerous traces are recorded (e.g., 20–50 per meter) forming a profile, also known as a *radargram*, that permits vertical relationships to be examined, including stratigraphic ones. Further information may be found in Conyers (2013).

GIS-based processing of GPR profiles is illustrated with data from a 1.2 ha survey performed at Pueblo Escondido, a late thirteenth century pit house village located in the Jornada Mogollon culture area of New Mexico. These profiles were obtained in 2005 utilizing a Geophysical Survey Systems, Inc., SIR-2000 GPR with a 400 MHz antenna and survey wheel (Ernenwein & Kvamme 2008). The data were collected with a nominal 40 traces per meter in a 30 ns time window with traces quantized in 512 samples.

The understanding and processing of GPR data presents an extra challenge to students. When numerous adjacent transects are acquired from an area the data become three-dimensional (see below), introducing another dimension of complexity. Second, the data in a vertical profile represent a composite of sinewaves replete with hyperbolic reflectors, very different from other geophysical data commonly employed in archaeology. What is easier to understand is that if reflection strength can be quantified from many adjacent profiles across an area, more interpretable *plan maps* can be generated (discussed in the next section), but only after the individual profiles are “cleaned up” and made more uniform, the task of this section. A single profile, *File70*, is employed here to illustrate GIS-based radargram processing and analysis.

“Flipping” every other transect

GPR surveys covering areas are typically conducted in a zigzag fashion, with alternating transects running in opposite directions. Numbered sequentially, that means the even numbered transects are “flipped,” or appear backwards, relative to the odd numbered ones. Processing, analysis, and interpretation are facilitated when all radargrams appear in the same direction. This issue is corrected with model FLIP, which utilizes TerrSet’s TRANSPPOSE module to reverse the order of columns in even (or odd) transects (Figure 4a). When saved as a sub-model, FLIP may be applied to a RGF that holds all even-number radargrams (Figure 4a).

Distance normalization

Most GPR surveys utilize a “survey wheel” that controls the number of traces per linear meter. While this number is set prior to a survey, over the distance of a transect inaccura-

cies often arise. *File70*, for example, contains 777 traces in its 20 m length, for an average of 38.85 traces per meter, compared to the targeted 40. Distance normalization resamples the image on the distance or *x*-axis through interpolation (Conyers 2012: 40). Model RESIZE accomplishes this through TerrSet’s RESAMPLE module (Figure 4b). User input to RESAMPLE merely changes the number of columns to their correct number (i.e., 800 here), keeping the other parameters for the result the same as the input. RESAMPLE also requires a “correspondence” file (which can be created using RESAMPLE or TerrSet’s text editor) that tells it to keep the radargram’s original coordinate system (i.e., only the number of columns is changed). The correspondence file SAME.COR is illustrated in Table 2. Its first row indicates the number of coordinate points in this file, typically four. Each succeeding row contains arbitrary spatial coordinates in the following order: *old-x*, *old-y*, *new-x*, *new-y* (where “old” refers to the input layer’s coordinates and “new” refers to the output layer’s coordinates). As can be seen, identical coordinates are retained in the output layer, so when SAME.COR is applied by RESAMPLE the input file’s coordinate system remains the same while the number of columns is changed. When saved as a sub-model, RESIZE may be applied to an entire RGF, enabling automatic distance normalization of a large number of radargrams (Figure 4b).

Revealing the waveform

The raw radargram *File70* is portrayed in Figure 4c. TerrSet display tools are employed to superimpose a coordinate grid (in red) showing meter marks on the *x*-axis and TWTT on the *y*-axis. Also shown are corresponding sample numbers for the *y*-axis. A first task in GPR processing is to make a decision about the approximate location of the ground surface, necessary for more accurate estimations of depth. Doing this requires visualization of the waveform. This is easily accomplished in TerrSet through its DIGITIZE tool which permits creation of a line vector down a particular trace (column) of interest (yellow line, Figure 4c). This vector is then input to the PROFILE module along with the radargram of interest. The result is the waveform of that trace calibrated to sample number on the vertical axis and showing reflection amplitude (within the two-byte signed integer range of approximately $\pm 33,000$) on the horizontal axis (Figure 4d).

Clipping profile at ground surface

Ernenwein (2006) describes the lack of consensus about the precise location of the ground surface in a GPR reflection trace, primarily because it can vary depending on transmitted frequencies and the electrical properties of the near-surface. Various studies place it between the first deviation from the mean (zero) and the peak of the first negative wave. The zero cross-point between the first positive and negative waves is employed here, which lies at approximately 1.6 ns TWTT or at sample number $k = 28$ (red circle, Figure 4d). Model TRUNCATE is employed to remove these early samples from a radargram (Figure 4e). It employs the WINDOW module where rows $k-1$ and larger are retained in a new profile with the remainder removed. Incidentally, other rows (samples) may also be

Figure 4: Macro-modeling tools applied to GPR profile: **a)** Model FLIP showing input and output, **b)** Model RESIZE, **c)** *File70* showing TWTT, meter marks, and digitized line, **d)** wave form and sample numbers from profile in **c)**, **e)** Model TRUNCATE for removing unwanted samples (rows), **f)** Model STACK for stacking or averaging traces, **g)** Model BACKGROUND for removal of horizontal banding, **h)** SUPERMODEL that combines all models.

Table 2: Organization of the correspondence file SAME.COR.

SAME.COR	Explanation:			
	4	← Number of data points		
		Old x	Old y	New y
0 0 0 0	0	0	0	0
0 1 0 1	0	1	0	1
1 1 1 1	1	1	1	1
1 0 1 0	1	0	1	0

eliminated in the same procedure, for example, those low in a profile below archaeological deposits of interest or where signal attenuation is obvious (**Figure 4c, d**). When saved as a sub-model, TRUNCATE may be applied to an entire RGF, enabling automatic processing of a large number of radargrams (**Figure 4e**).

Stacking

GPR data may be densely sampled in the field by acquiring a large number of traces per meter along a transect. Often, the data may appear “noisy” on the horizontal axis due to uneven ground or other circumstances, in which case the data may be “stacked,” meaning that an average of several

adjacent traces is computed, creating a smoothing effect to improve the apparent signal and clarify reflections (Conyers 2012: 40). At Pueblo Escondido 40 trace per meter (every 2.5 cm) were acquired for a total of 800 along the 20 m profiles. Stacking by a factor of two will average every pair of adjacent traces yielding 5 cm resolution and 400 traces while stacking by four produces 10 cm resolution and 200 profile traces. Model STACK (**Figure 4f**) employs the CONTRACT module with the aggregation option for averaging with contraction only along the x-axis by 2× (for stacking of two traces), 3× (stacking of three traces), and so on. When saved as a sub-model, STACK may be applied to an entire RGF, enabling automatic processing of a large number of radargrams (**Figure 4f**).

Background removal

The pronounced horizontal banding commonly seen in GPR profiles (**Figure 4c**) greatly detracts from their interpretation by obscuring reflections of interest. They are commonly removed by computing the average reflection amplitude along a particular row (sample) and subtracting it from the raw profile, thereby removing the average amplitude of that row (i.e., the horizontal banding) and leaving local reflections apparent (Conyers 2012: 40–41). This is accomplished in model BACKGROUND which utilizes the FILTER module (**Figure 4g**). It requires a pre-established custom designed filter (that can be created prior to a run using FILTER). The widest possible filter is

desired to isolate the overall mean of each input row, so the maximum filter size of 255 columns is employed. TerrSet requires all filters to possess at least three rows (defining a filter of $255c \times 3r$), but the first and third are ignored by filling them with zeros while the second row is populated with ones. When FILTER is applied each filter element is multiplied by a corresponding cell in a row of the radargram; the “normalize” option then divides by the number of ones to yield a mean, which isolates the banding (*tmp001* in **Figure 4g**). This result is then subtracted from the input through OVERLAY to yield a “clean” profile (*tmp002* in **Figure 4g**). When saved as a sub-model, BACKGROUND may be applied to an entire RGF enabling background removal in a large number of radargrams (**Figure 4g**).

Putting it all together

A typical processing sequence for a collection of GPR profiles places all even-numbered profiles in an RGF and submits them to FLIPMODEL (**Figure 4a**). The reversed files then are combined with the original odd-numbered profiles in a second RGF, perhaps named *allGPR*. Several individual profiles might then be examined to estimate “ground zero,” the sample or row number to be regarded as the approximate ground surface, as portrayed in **Figure 4c, d**. Finally, depending on needs (whether distance normalization or stacking are required), the foregoing models may be combined into a single “supermodel” combining them and permitting nearly all operations to be undertaken in a single run, with processed layers held in an RGF (here, *finalGPR*; **Figure 4h**).

Case Study 4: Generating GPR Time Slices

Adjacent, closely-spaced, parallel transects permit entire areas to be covered in GPR surveys. In such contexts, GPR data become three-dimensional, with traces forming the *z* or vertical axis, the transects the *x*-axis, and the orthogonal transect separation the *y*-axis (Conyers 2013). Horizontal plan maps of GPR reflection amplitudes across an area may be generated at a variety of TWTT, a proxy for depths below the surface (and convertible to approximate depths if GPR wave velocity is known or estimated). Time-slices are created by extracting samples (rows) from a “time window” in an identical range of TWTT from each trace along each profile, and after a few manipulations assembling them in correct spatial position. Conceptually, this task seems to be the most foreboding to students, but they soon learn that it is nearly a trivial operation.

The following employs 40 radargrams from Pueblo Escondido, each processed by elements of SUPERMODEL (**Figure 4h**). These data, held in east-west running profiles named *File41–File80*, were obtained from a 20×20 m area with each separated by a half-meter (**Figure 5a**). In the following the processed *File70* is again employed for primary illustration.

Extracting a slice from one profile and aggregating the data

Many investigators pursue slicing operations by arbitrarily generating time-slices at equal intervals through the depth (time window) of a profile, for example every

10 ns TWTT. It is also possible to examine the nature of reflections and their amplitudes in processed profiles for clues that permit judgmental slices to be generated. In this case, top and bottom TWTT are determined that hopefully isolate archaeological features of interest. This process, admittedly with some foreknowledge of the site (Ernenwein & Kvamme 2008), is illustrated in **Figure 5b** where a series of pronounced reflections that consistently occur in a time window between 5–8 ns TWTT are isolated.

Model EXTRACTSLICE first employs the WINDOW module to extract the time window of interest from a profile. The three nanoseconds of the windowed result (**Figure 5c**, upper right) translates to 51 rows or samples while the 40 traces per meter over 20 m yields 800 columns (unstacked). As this result (in *tmp001*) contains positive and negative amplitudes, absolute or even squared amplitudes are computed in Step 2 with the TRANSFORM module in order to ultimately permit reflection maxima to be isolated (*tmp002* in **Figure 5c**, middle right). The final step down-samples the data vertically, from the many rows of the windowed result to a single row (which will contribute one row of *x*-axis data to an ultimate plan map, as in **Figure 5a**). The data are also down-sampled horizontally from many traces per meter (even after stacking) to a spatial resolution more commensurate with the half-meter transect separation that forms the plan map’s *y*-axis (**Figure 5a**). Here, the desired outcome will contain one row and, selecting quarter-meter spatial resolution, 80 columns, depicted as the yellow “boxes” in **Figure 5c**. That means that in each of those boxes, $51r \times 10c$ of data (510 data points) are contracted to a single pixel. This is achieved through module CONTRACT, where an appropriate *x*-axis contraction (here $10\times$) and *y*-axis contraction (here $51\times$) is undertaken, with “maximum value” selected for each output cell. The outcome is a single row of data, $80c \times 1r$, where each cell holds the maximum absolute amplitude of the quarter-meter space and three ns time range that it represents (**Figure 5c**, lower right). When saved as a sub-model, EXTRACTSLICE may be applied to an entire RGF, enabling slice extraction and compaction in a full set of radargrams (**Figure 5d**).

Concatenating multiple profile slices for regional plan map

The final task in the preparation of a slice map merely involves the concatenation of the extracted and compressed data from each radargram in correct spatial position. Unfortunately, there is no easy way to fully automate this process because the output of each result from EXTRACTSLICE retains the coordinate systems of the individual GPR profiles (**Figure 5b**), not of the desired plan map (**Figure 5a**). Two options exist. One is to employ the CONCAT module to manually concatenate each extracted slice to their position in the desired plan map by specifying the desired row number each should occupy. The second option is to manually alter the *y*-coordinates in the metadata of each extracted slice to the spatial position it will occupy in the 20×20 m

Figure 5: Macro-modeling tools for GPR time-slicing: **a)** plan map showing layout, dimensions, and file positions in survey block, **b)** processed *File70* with TWTT, meter marks, and locus of 5–8 ns time slice, **c)** Model EXTRACTSLICE applied to *File70* showing consequences of each step at right, **d)** EXTRACTMODEL that permits slice extraction from a raster group of GPR profiles, **e)** initial time slice, **f)** time slice in *e* after de-spiking, **g)** time-slice in *f* after interpolation to uniform 25 cm pixels, **h)** time-slice in *g* after application of a low-pass filter.

plan map (Figure 5a) and then employ COMPOSMODEL (Figure 2g) to automatically place each in correct position. The second option was employed here with the result shown in Figure 5e. Clearly, a pit house measuring approximately eight meters square is indicated along with a number of smaller structures to the north and a series of linear structures to the west.

Post-Processing: Image Enhancements

After the “considerable” work demonstrated in the foregoing (according to students), they are frequently disappointed in the appearance of their results with data appearing “too blocky” (pixilated), too dark, or too noisy, for example. Each of the processed composite data sets (Figures 2h, 3f and 5e) represent initial views that can be considerably enhanced through low-pass, high-pass, and other filters, additional de-spiking, and interpolation to higher spatial resolutions in order to reduce pixilation.

De-spiking

Model DESPIKE (Figure 3b) may be employed again in post-processing to reduce the effects of outlying or extreme data points that detract from the result. The initial GPR time-slice (Figure 5e) illustrates a number of isolated extreme measurements, so DESPIKE was applied to it using a 1.5 s. d. threshold which eliminates many of them (Figure 5f).

Interpolation

Interpolation introduces a greater number of measurements or pixels to a finite coordinate space, thereby reducing pixilation and creating a more continuous appearing image. It is accomplished with model RESIZE (Figure 4b) where, typically, an extra row (or column) is added between extant rows (or columns) through a cubic convolution algorithm that guarantees smooth estimated transitions between them. The de-spiked GPR time-slice

(**Figure 5f**), with quarter-meter sampling on the *x*-axis and half-meter sampling on the *y*-axis, was resampled to a uniform quarter-meter spatial resolution (**Figure 5g**).

Low-pass filters

Low-pass filters are “averaging” or smoothing filters that tend to consolidate anomalous features in an image and reduce noise, “passing” low (spatial) frequency image elements to the result (Lillesand, Kiefer & Chipman 2008: 509–512). They do this by taking a simple average of cell (pixel) values in a neighborhood for strong smoothing. Weaker smoothing is accomplished with smaller neighborhoods or by employing “Gaussian” weighting, where a cell’s weight in a neighborhood decreases with distance from its center (mimicking a normal or bell-shaped curve). TerrSet offers a variety of default neighborhood sizes for either form in its FILTER module, or a user-defined option permits neighborhoods of any shape, size, or arrangement of weights. A small 3×3 Gaussian weighting scheme was applied to the interpolated GPR data that consolidates anomaly definition (**Figure 5h**).

High-pass filters

High-pass filters are differencing filters where the value of a cell is differenced from the average of its surrounding neighborhood, highlighting its uniqueness and thereby passing high spatial frequency elements to the result (Lillesand, Kiefer & Chipman 2008: 509–512). They are particularly useful to clarify features in electrical resistance data that exhibit highly dynamic variation (e.g., **Figure 3f**). FILTER permitted application of a large 10-cell radius high-pass filter to these data clarifying much of the apparent pattern in an image with a more uniform spread of gray-values (**Figure 3g**).

Conclusions

The foregoing has demonstrated that basic GIS tools can be made to accomplish powerful results in applications as specialized as geophysical data processing. It is shown that complex operations may be broken down into a series of individual tasks or steps, each of which may seem innocuous and meaningless, but when combined in a logical sequence yield powerful outcomes. By taking a complex problem and breaking it down into a series of easy to understand stages comprehension is enhanced. Moreover, icon-driven drag-and-drop GUI interfaces aid in this process because the graphical element is more intuitive and easier to “read,” in contrast to complicated coding and scripting languages. That is why such an approach aids the learning process, facilitating understanding of what each tool or model accomplishes. In short, a programming environment is offered that is relatively easy to learn and understand, and nearly a decade of experience at the University of Arkansas has shown that students can work in this environment without prior programming experience. Indeed, many students enjoy it and some even become passionately excited by this process. Better still, all come away with rare knowledge of “how things work,” a rare circumstance in our increasingly technological world, but nevertheless important

for future scientists and engineers. Such a result is not necessarily the case in windows-driven, point-and-click, black-box software environments.

It is again emphasized that the foregoing is not intended as a comprehensive geophysical data processing system, although a great diversity of tools is offered. More specialized and difficult-to-implement processing routines have not been pursued, such as “reduction to the pole” in magnetometry, GPR migration routines, or hyperbola fitting for GPR velocity estimation. Users will have to go elsewhere for such capabilities. Nevertheless, the basic building blocks for geophysical data processing can be found here. A number of modules also have not been presented for lack of space, such as ones for “gaining” GPR data, plow scar removal in plan maps through fast Fourier transforms, variations for data importing, or the avoidance of “edge effects” that may arise in various filtering operations. Interested parties may contact the author for further information about these capabilities.

The foregoing has shown that GIS can do nearly as well as specialized commercial software in the processing of geophysical data and it often exceeds the capabilities of such software in some operations. Processing geophysical data in a GIS environment offers additional advantages. Geophysical results may be co-registered to a common coordinate base permitting ready combining or overlaying with other geophysical, archaeological, topographic, aerial, or satellite data layers, facilitating interpretation and greater understanding of site-specific information. On-screen digitizing permits anomalous indications in geophysical data to be vectorized, enabling interpreted features to be explicitly defined as points, lines, or polygons. Moreover, the GIS environment facilitates other common research endeavors in geophysical archaeology. For example, map algebra methods or principal components analysis, common within GIS, may be utilized to integrate geophysical data from multiple survey modalities, permitting “fusion” of their information for greater subsurface insights (e.g., Nowaczinski et al. 2015). Coupled with the insights it offers in student training, these advantages further underscore the importance of GIS to geophysical endeavors in archaeology.

Acknowledgements

Many thanks go to the students of my archaeological prospecting class at the University of Arkansas over the past decade who participated in TerrSet programming environments and, through questions and difficulties, encouraged me to improve and enhance the various tools shown here. I am also grateful to the various students who helped me collect the various geophysical data sets employed as examples in the foregoing. The comments of two anonymous reviewers who suggested a number of additions and changes helped clarify this presentation.

Competing Interests

The author has no competing interests to declare.

References

- Aspinall, A, Gaffney, C and Schmidt, A.** 2008. *Magnetometry for archaeologists*. Lanham, Maryland: AltaMira Press.
- Ciminale, M and Loddo, M.** 2001. Aspects of magnetic data processing. *Archaeological Prospection*, 8(4): 239–246. DOI: <https://doi.org/10.1002/arp.172>
- Clark Labs.** 2018. *TerrSet Geospatial Monitoring and Modeling Software*. Available at URL: <https://clarklabs.org/tererset/> [Last accessed 16 May 2018].
- Conolly, J and Lake, M.** 2006. *Geographical information systems in archaeology*. Cambridge: Cambridge University Press. DOI: <https://doi.org/10.1017/CBO9780511807459>
- Conyers, LB.** 2012. *Interpreting ground-penetrating radar for archaeology*. Walnut Creek, California: Left Coast Press.
- Conyers, LB.** 2013. *Ground-penetrating radar for archaeology*, 3rd ed. Latham, Maryland: Alta Mira Press.
- DW Consulting.** 2018. *TerraSurveyor*. Available at URL: <https://www.dwconsulting.nl/TerraSurveyor.html> [Last accessed 16 May 2018].
- Ernenwein, EG.** 2006. Imaging in the ground-penetrating radar near-field zone: a case study from New Mexico, USA. *Archaeological Prospection*, 13(2): 154–156. DOI: <https://doi.org/10.1002/arp.286>
- Ernenwein, EG and Kvamme, KL.** 2008. Data processing issues in large-area GPR surveys: correcting trace misalignments, edge discontinuities and striping. *Archaeological Prospection*, 15(2): 133–149. DOI: <https://doi.org/10.1002/arp.331>
- Gaffney, C and Gater, J.** 2003. *Revealing the buried past: geophysics for archaeologists*. Stroud (UK): Tempus Publishing.
- Gallo, D, Ciminale, M, Becker, H and Masini, N.** 2009. Remote sensing techniques for reconstructing a vast Neolithic settlement in Southern Italy. *Journal of Archaeological Science*, 36(1): 43–50. DOI: <https://doi.org/10.1016/j.jas.2008.07.002>
- Geoscan Research.** 2018. *Geoplot 3.0 for Windows*. Available at URL: <http://www.geoscan-research.co.uk/page9.html> [Last accessed 16 May 2018].
- GPR-Slice & GPRSIM.** 2018. *GPR-Slice Ground Penetrating Radar Imaging Software*. Available at URL: <http://www.gpr-survey.com/> [Last accessed 16 May 2018].
- Kvamme, KL.** 2006. Data processing and presentation. In: Johnson, J (ed.), *Remote Sensing in Archaeology: An Explicitly North American Perspective*, 235–250. Tuscaloosa: University of Alabama Press.
- Kvamme, KL and Ahler, SA.** 2007. Integrated remote sensing and excavation at the Double Ditch State Historic Site, North Dakota. *American Antiquity*, 72(3): 539–561. DOI: <https://doi.org/10.2307/40035860>
- Lillesand, TM, Kiefer, RW and Chipman, JW.** 2008. *Remote sensing and image interpretation*, 6th ed. New York: John Wiley.
- Linford, N, Linford, P and Payne, A.** 2015. Chasing aeroplanes: developing a vehicle-towed caesium magnetometer array to complement aerial photography over three recently surveyed sites in the UK. *Near Surface Geophysics*, 13(6): 623–631. DOI: <https://doi.org/10.3997/1873-0604.2015044>
- MATGPR.** 2018. *MATGPR R-3.5*. Available at URL: <http://users.uoa.gr/~atzanis/matgpr/matgpr.html> [Last accessed 16 May 2018].
- MathWorks.** 2018. *Matlab*. Available at URL: <https://www.mathworks.com/products/matlab.html> [Last accessed 16 May 2018].
- Neubauer, W.** 2004. GIS in archaeology—the interface between prospection and excavation. *Archaeological Prospection*, 11(3): 159–166. DOI: <https://doi.org/10.1002/arp.231>
- Nowaczinski, E, Schukraft, G, Rassmann, K, Reiter, S, Müller-Scheeßel, N, Hecht, S, Eitel, B, Bubenzer, O and Batora, J.** 2015. A multidimensional research strategy for the evaluation of settlement pits: 3D electrical resistivity tomography, magnetic prospection and soil chemistry. *Archaeological Prospection*, 22(4): 233–253. DOI: <https://doi.org/10.1002/arp.1510>
- Patton, M.** 2013. *Geophysical surveys and archaeological insights at Fort Pierre Chouteau a frontier trading post on the Middle Missouri*. Unpublished thesis (MA), University of Arkansas, Fayetteville.
- Python Software Foundation.** 2018. *Python*. Available at URL: <https://www.python.org> [Last accessed 16 May 2018].
- QGIS.** 2018. *QGIS Python Plugins Repository*. Available at URL: <https://plugins.qgis.org/plugins/AGT/> [Last accessed 16 May 2018].
- R.** 2018. *The R Project for Statistical Computing*. Available at URL: <https://www.r-project.org/> [Last accessed 16 May 2018].
- Schmidt, A.** 2013. *Earth resistance for archaeologists, geophysical methods for archaeology*. Lanham Maryland: AltaMira Press.
- Schneidhofer, P, Nau, E, Hinterleitner, A, Lugmayr, A, Bill, J, Gansum, T, Paasche, K, Seren, S, Neubauer, W, Draganits, E and Trink, I.** 2016. Palaeoenvironmental analysis of large-scale, high-resolution GPR and magnetometry data sets: The Viking Age site of Gokstad in Norway. *Archaeological and Anthropological Sciences*, 9(6): 1187–1213. DOI: <https://doi.org/10.1007/s12520-015-0312-x>
- Schuler, HH.** 1990. *Fort Pierre Chouteau*. Vermillion, South Dakota: University of South Dakota Press.
- SussexArch.** 2018. *Snuffler – Freeware Geophysics Software*. Available at URL: <http://www.sussexarch.org.uk/geophys/snuffler.html> [Last accessed 16 May 2018].

How to cite this article: Kvamme, K. 2018. Getting Around the Black Box: Teaching (Geophysical) Data Processing through GIS. *Journal of Computer Applications in Archaeology*, 1(1), pp. 74–87, DOI: <https://doi.org/10.5334/jcaa.14>

Submitted: 07 March 2018

Accepted: 18 May 2018

Published: 30 May 2018

Copyright: © 2018 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <http://creativecommons.org/licenses/by/4.0/>.

]u[*Journal of Computer Applications in Archaeology* is a peer-reviewed open access journal published by Ubiquity Press.

OPEN ACCESS