A Tool for the Construction of Synthetic Petroglyphs

Alessandro Buccini and Elisa Crabu(✉)

Department of Mathematics and Computer Science, University of Cagliari,
Via Ospedale 72, Cagliari, Italy
alessandro.buccini@unica.it,
elisa.crabu@unica.it
https://web.unica.it/unica/page/it/alessandro_buccini ,
https://bugs.unica.it/ elisa/

Abstract. Archaeologists are often tasked with the identification and extractions of glyphs and petroglyphs from surfaces. Since this task is usually performed by hand, it is time consuming and prone to human errors. Therefore, in the years several automatic techniques have been developed to solve this problem. However, to the best of our knowledge, a tool for the creation of realistic synthetic data has not be developed and distributed yet. It is the purpose of this paper to fill this gap. We construct a Matlab tool with a graphic user interface to create three-dimensional realistic surfaces where petroglyphs have been carved. The user can select the shape, position, and depths of these petroglyphs. This tool can be used to validate extraction algorithms and to create training data for supervised learning techniques.

Keywords: Glyph creation · Glyph extraction · Matlab graphic user interface

1 Introduction

In archaeology we are often faced with the problem of extracting glyph, or petroglyph, from a surface. This procedure is usually performed manually and, therefore, is prone to errors and extremely time consuming. Several automatic approaches have been proposed throughout the years; see, e.g., [9] for a review. In all cases these methods have been tested on real data. However, in our view, a tool for constructing synthetic, but realistic, data to validate segmentation and extraction methods is missing. It is the purpose of this paper to provide a simple tool to construct synthetic and realistic petroglyphs.

Our main area of interest concerns the documentation of petroglyphs and wall decorations of the so-called *Domus de Janas*, funerary hypogea founded in Sardinia, Italy. In these archaeological sites the glyphs are mostly composed of spirals and jagged lines. Therefore, our software considers this kind of decorations. The extension to different types of petroglyphs is trivial.

To achieve this, we simulated a scenario in which a three-dimensional reconstruction of the surface is obtained by the Photometric Stereo technique; see,

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2026
O. Gervasi et al. (Eds.): ICCSA 2025 Workshops, LNCS 15892, pp. 417–429, 2026.
https://doi.org/10.1007/978-3-031-97638-4_26

e.g., [3,4] and references therein. This method consists in acquiring pictures of the surface of interest with a fixed camera under different light conditions. Combining the pictures it is possible to obtain a three-dimensional reconstruction of the surface. In particular, if the directions from which the light is coming from are known, then three images are enough to compute a reconstruction, while if these directions are unknown, then at least six pictures are needed; see Sect. 2 for more details.

We construct a tool that uses two real reconstructions of a rock with no incisions and "carves" on these synthetic petroglyphs composed of a spiral and a jagged line. The tool provides a graphic user interface that allows the user to select the size of the computed data, the position of the glyphs, their shape, and the depth of the incision. Using this tool one can actually create a myriad of different petroglyphs to either test the performances of segmentation algorithms or train machine learning methods.

This paper is structured as follows: Sect. 2 reviews the photometric stereo technique that was used to recover the real backgrounds. In Sect. 3 we describe in details how the petroglyphs are constructed and in Sect. 4 we provide a guide to the use of our tools and show two examples. We draw our conclusions in Sect. 5.

2 Photometric Stereo

In this section we describe the Photometric Stereo model that was used to construct the data that constitute the starting point of our tool.

Photometric Stereo (PS), [11,12], is a Computer Vision technique which provides a three-dimensional reconstruction of an object. This is an alternative technique of the Shape from Shading (SfS) model, [5,13], that derives shape and color information from a set of images of the target. In the original formulation, a collection of pictures is obtained by fixing a camera in front of the object, and by placing a light source in different known positions, so that each image is obtained under a different lighting direction. Both the camera and the light source are placed at infinite distance. Furthermore, the object is assumed to be a Lambertian reflector, i.e., its surface diffuses the light equally in each direction. Under these assumptions, Kozera [8] proved that two images are needed to find the solution of the problem, which consists on the resolution of a first-order Hamilton-Jacobi partial differential equation system; see [10] for a numerical resolution.

However, in real scenarios knowing the positions of the light source during the acquisition of the images is a huge limitation. In the *Domus de Janas* application, since these sites are formed by small and carved rooms, as can be seen in Fig. 1, exact measurement of these positions is impossible. In order to treat this problem, in [3] a method to estimate light position from the data has been proposed, based on the Hayakawa's theorem [7]. In this case at least six pictures are required to find the solution.



Fig. 1. Engravings found in the entrance of the *Domus de Janas* of Corongiu, Pimentel, Italy, placed over the entry to the internal chamber.

Furthermore, in real applications, the assumptions of light and camera at infinite distance and the Lambertianity of the surface may not be satisfied. Indeed, real-world materials generally are not ideal Lambertian surfaces, so that the reflected light may depend on the point of view. Additionally, while in some cases light source and camera can be placed far from the object, as in [3] where a dataset of a seashell is obtained using the sun as light source, in other situations, like the archaeological one described in [4], light source and camera have to be placed close to the object. These non-ideal conditions have been treated in [4], where, by defining two indicators of ideality, starting from a large dataset, one can extract a set of images that will provide the best reconstruction of the observed object.

We will now briefly resume the main steps of this model, which are described in detail in [3].

Let us consider a reference system in $\mathbb{R}^3$, so that the object is placed in the origin and the optical axis of the camera coincides with the z-axis. Each image of the acquired dataset has resolution $(r+2)\times(s+2)$, resulting associated to a rectangular domain $\Omega = \left[-\frac{A}{2}, \frac{A}{2}\right] \times \left[-\frac{B}{2}, \frac{B}{2}\right]$, with A the horizontal size of the image, $B = (s+1)h$ its vertical one, $h = A/(r+1)$.

Assuming that the surface of the observed object is represented by the bivariate function $z = u(x, y), (x, y) \in \Omega$, the normalized normal vector is given by

$$\mathbf{n}(x, y) = \frac{[-u_x, -u_y, 1]^T}{\sqrt{1 + \|\nabla u\|^2}}, \tag{1}$$

where u_x, u_y are the partial derivatives, ∇u the gradient and $\|\cdot\|$ the 2-norm. By discretizing the domain, i.e., considering a grid of points $(x_i, y_j), i = 0, \ldots, r+1, j = 0, \ldots, s+1$, and sorting the pixels lexicographically, the values assumed by u at these points can be indifferently indicated by the symbols $u(x_i, y_j), u_{i,j}, u_k$, with $k = (i-1)s + j$ index of the internal points, which are in number $p = rs$.

Considering a dataset of q images, the t-th light direction propagating from the object to the light source is given by the vector $\boldsymbol{\ell}_t = [\boldsymbol{\ell}_{1t}, \boldsymbol{\ell}_{2t}, \boldsymbol{\ell}_{3t}]^T$, whose norm is proportional to the intensity of the light.

Since the surface is Lambertian, it satisfies the *Lambert's cosine law*

$$\rho(x,y)\langle \mathbf{n}(x,y), \boldsymbol{\ell}_t \rangle = I_t(x,y), \qquad t = 1, \dots, q, \tag{2}$$

where $\rho(x,y)$ is a quantity, called *albedo*, which take into account the partial absorption of the light from the surface, and $I_t(x,y)$ represents the value of the pixel of the t-th image at the point (x,y).

By considering a discretization of formula (2) and the lexicographic order for the pixels, one find a matrix expression of the Lambert's law, as

$$DN^T L = M, \tag{3}$$

by defining the matrices: $D = \text{diag}(\rho_1, \dots, \rho_p) \in \mathbb{R}^{p \times p}$, with $\rho_k, k = 1, \dots, p$, the albedo value at each pixel, $N = [\mathbf{n}_1, \dots, \mathbf{n}_p] \in \mathbb{R}^{3 \times p}$ and $L = [\boldsymbol{\ell}_1, \dots, \boldsymbol{\ell}_q] \in \mathbb{R}^{3 \times q}$ containing the normal vectors and the light directions, $M = [\mathbf{m}_1, \dots, \mathbf{m}_q] \in \mathbb{R}^{p \times q}$ which contains in each column the vectorized images of the dataset, where p and q are the number of pixels and images, respectively.

In our study $q << p$, since usually we consider a small number of images compared to the number of pixels of a picture, that is very large.

Under the assumption of known lighting, i.e., the matrix L, by defining $\tilde{N} = ND$ and by using the pseudoinverse of L, [2], we have that $\tilde{N}^T = ML^\dagger$. After finding N and D by normalizing the columns of $\tilde{N}$, the surface of the object is given by the resolution of the Poisson problem $\Delta u(x,y) = f(x,y)$, with f obtained by numerically differentiating the normal field; see [4] for details.

The request of the lighting knowledge is very limiting in real applications, especially in our area of interest. In order to find a solution of the problem, we follow the Hayakawa's technique, of which we will now summarize the main steps, described in detail in [3].

Let us assume that light vectors are normalized, i.e., $\|\boldsymbol{\ell}_t\| = 1, t = 1, \dots, q$. Since the matrix M is expected to be rank-3, to obtain an initial factorization, we consider its "compact" Singular Value Decomposition (SVD), [6],

$$M = U\Sigma V^T, \tag{4}$$

where $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_q)$ contains the singular values $\sigma_1 \geq \cdots \geq \sigma_q \geq 0$, and $U \in \mathbb{R}^{p \times q}, V \in \mathbb{R}^{q \times q}$ contain orthonormal columns $\mathbf{u}_i, \mathbf{v}_i$. Theoretically, $\sigma_4 = \cdots = \sigma_q = 0$, since M is rank-3, but, due to experimental errors, usually these values are non-vanishing. We, therefore, consider the truncating SVD, [2], $M = W^T Z$, by defining $W = [\sigma_1 \mathbf{u}_1, \sigma_2 \mathbf{u}_2, \sigma_3 \mathbf{u}_3]^T, Z = [\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3]^T$. This provides the best rank-3 decomposition respect to both the 2-norm and the Frobenius norm. Following the Hayakawa's theorem, we seek a matrix B such that

$$\|B\mathbf{z}_t\| = 1, \quad t = 1, \dots, q, \tag{5}$$

with $\mathbf{z}_t, t = 1, \ldots, q$ the columns of Z. This problem turns out into a system of equations $H\mathbf{g} = \mathbf{1}$, where H is a $q \times 6$ matrix of rows

$$\begin{bmatrix} z_{1t}^2 & z_{2t}^2 & z_{3t}^2 & 2z_{1t}z_{2t} & 2z_{1t}z_{3t} & 2z_{2t}z_{3t} \end{bmatrix}, \quad t = 1, \ldots, q, \tag{6}$$

which solution $\mathbf{g} = (g_{11}, g_{22}, g_{33}, g_{12}, g_{13}, g_{23})^T$ is unique if and only if the number of available images $q \geq 6$, i.e., the matrix H is full-rank, see [4] for details. The solution $\mathbf{g}$ contains the entries of a matrix $G = B^T B$ that, theoretically, is symmetric positive definite. The matrix B can be obtain by computing the Cholesky factorization of G, from which it follows that the normal vector field is determined minus one unitary transformation Q. This is due to the uncertainty of the reference orientation, known as *bas-relief ambiguity*, [1], for which in [3] a method to find it has been defined. Once the matrix Q has been determined, the solution of the problem (3) will be given by

$$\tilde{N} = QB^{-T}W \quad \text{and} \quad L = QBZ. \tag{7}$$

When the method is applied to experimental dataset, some computational problems may arise. These lead either to an inaccurate reconstruction of the surface or to a breakdown of the algorithm; see [4] for more details. These issues are due to the fact that some of the assumptions of the method may not be met. As we have already observed, in the *Domus de Janas* application, light sources can not be place at infinite distance from the engravings, and, in addition, the rock is a non-ideal Lambertian reflector. This leads into distortion in the data, that lose their ideality. Starting from a large dataset one can find a subset of images which provides an accurate reconstruction of the surface. To do that, in [4], two numerical indicators have been defined as a measure of data ideality. Applying the algorithms described in the paper, one can extract a subset of ideal images from which the reconstructed surface will be the most accurate.

3 Constructions of the Petroglyphs

We now describe how we construct our realistic synthetic petroglyphs. We start from the reconstruction of two rock walls obtained using the photometric stereo technique described above; see Sect. 3.1. We then construct the glyphs which are composed by a spiral and/or a jagged line. The user can specify the shape, position, and depth of these incisions, which are "carved" in the background; see Sect. 3.2.

3.1 Backgrounds

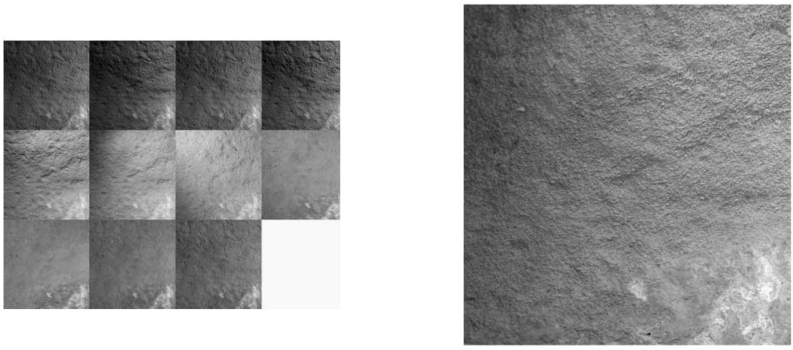
We first describe the two possible backgrounds between which the user can pick.

These backgrounds are obtained from two datasets, collected for another study, which consist in two walls of two different *Domus de Janas*. For the first dataset, only a portion of the wall has been considered, since we require a surface without carvings. The second one, as it does not contain engravings,

is considered totally. For this two surfaces, a set of images has been collected, by fixing a camera in front of them and by placing a light source, a flashlight, around them, in different position, so that each picture is obtained under a different lighting condition. From these images, one can determine the sought surfaces by using the PS technique. Before that, for the observations made in the previous section, it is necessary to preprocess the dataset to exclude non-ideal images. To do that, we choose to use the first indicator, described in [4], to test the available data. Once the ideal dataset has been determined, by applying PS one obtains the reconstruction of the backgrounds available in this tool. The first



Fig. 2. Engravings found in the entrance of the *Domus de Janas* of Corongiu, Pimentel, Italy. To construct our surface, we used the top left portion of these pictures.



(a) Available dataset (b) Reconstructed surface

Fig. 3. On the left, 11 of the available dataset formed by 13 pictures obtained under different lighting conditions. On the right, the reconstructed surface obtained by using the ideal dataset of 9 images.

surface is a portion of the wall of the *Domus de Janas of Corongiu*, Pimentel, Italy, displayed in Fig. 2: the top left portion has been considered to obtain the surface. In this case, 13 pictures have been collected, of which in Fig. 3a we can see some of the portion used to construct the background. Using the first indicator, 9 images have been selected to constitute the ideal dataset. From this, the described technique provides the surface displayed in Fig. 3b.

The second one is a wall at Siligo, Sardinia, shown in Fig. 4. Archaeologists are still debating if this wall belonged to a *Domus de Janas*, however, it is still usable for our end.

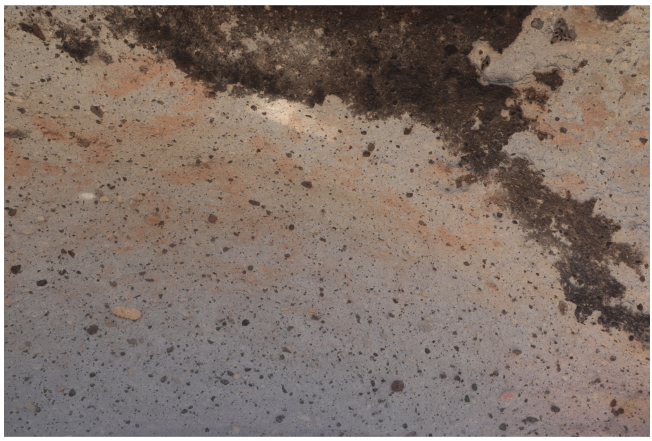
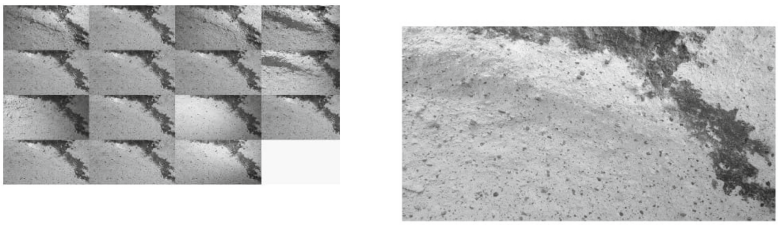


Fig. 4. Wall at Siligo, Sardinia, Italy. To construct our surface, we used the whole area since no carving was present.



(a) Available dataset (b) Reconstructed surface

Fig. 5. On the left, 15 of the available dataset formed by 18 pictures obtained under different lighting conditions. On the right, the reconstructed surface obtained by using the ideal dataset of 17 images.

For this site, 18 photos have been collected, see Fig. 5a, of which 17 have been selected by the indicator. From these, the surface reconstructed by the the PS is the one in Fig. 5b.

3.2 Glyphs

Once the user selects the desired background, they need to select the glyphs they wish to "carve". We provide the user with two different shapes: a spiral and a jagged line. We report in Fig. 6 an example of spiral and one of jagged line produced by our tool.



(a) (b)

Fig. 6. Examples of spiral (a) and jagged line (b).

Spiral. We first describe the spiral our tool constructs. Let us first define the following parameters

- $r \in \mathbb{R}^+$, number of rounds that the spiral makes;
- $w_S \in \mathbb{R}^+$, width of the spiral;
- $d \in \mathbb{R}^+$, diameter of the spiral.

We define the spiral S as the area of the plane included between the curves

$$\gamma_{\text{int}}^S(t) = \begin{bmatrix} \frac{dt}{2r\pi}\cos(t) \\ \frac{dt}{2r\pi}\sin(t) \end{bmatrix} \quad \text{and} \quad \gamma_{\text{ext}}^S(t) = \begin{bmatrix} \left(\frac{dt}{2r\pi} + w_S\right)\cos(t) \\ \left(\frac{dt}{2r\pi} + w_S\right)\sin(t) \end{bmatrix} \quad \text{for } t \in [0, 2r\pi], \tag{8}$$

and the segments joining $\gamma_{\text{int}}(0)$ and $\gamma_{\text{ext}}(0)$, and joining $\gamma_{\text{int}}(2r\pi)$ and $\gamma_{\text{ext}}(2r\pi)$.

It is trivial to see that the spiral S belongs to the square $\Omega_S = \left[-\frac{s}{2} - w_S,\ \frac{s}{2} + w_S\right]^2$.

Jagged line. We now describe the jagged lines constructed by our tool. We first set the following parameters

- $n \in \mathbb{N}$, number of peaks;

- $h \in \mathbb{R}^+$, height of the peaks;
- w_J, width of the jagged line;
- $l \in \mathbb{R}^+$, length of the jagged line.

We define, for $j \in \{0,\ 1,\ 2, \ldots, 2n-1\}$, the curves

$$\begin{aligned} \gamma^J_{\text{low}}(t,j) &= \left[t, (-1)^j h \left(t - j\frac{l}{2n}\right) + h\,(\text{mod}(j,2))\right]^T, \\ \gamma^J_{\text{high}}(t,j) &= \left[t, (-1)^j h \left(t - j\frac{l}{2n}\right) + h\,(\text{mod}(j,2)) + w_J\right]^T, \end{aligned} \tag{9}$$

for $t \in \left[j\frac{l}{2n}, (j+1)\frac{l}{2n}\right]$, where $\text{mod}(j,2) = 0$, if j is even, and $\text{mod}(j,2) = 1$, if j is odd. We define the jagged line J as the area of the plane included between the curves $\gamma^J_{\text{low}}(t,j)$, $\gamma^J_{\text{high}}(t,j)$, the segment connecting $\begin{bmatrix}0\\0\end{bmatrix}$ and $\begin{bmatrix}0\\h\end{bmatrix}$, and the segment connecting $\begin{bmatrix}l\\0\end{bmatrix}$ and $\begin{bmatrix}l\\h\end{bmatrix}$.

Similarly as above, it is trivial to see that $J \subset \Omega_J = [0,\ l] \times [0, h + w_J]$.

3.3 Carving of the Glyphs

We assume that we can represent the background as a function

$$b : [0,a] \times [0,b] \to \mathbb{R}.$$

We set the following constants

- τ_S, relative size of the spiral with respect to the whole background;
- τ_J, relative size of the jagged line with respect to the whole background;
- $\delta_S > 0$, depth of the carving of the spiral;
- $\delta_J > 0$, depth of the carving of the jagged line;
- x_S and y_S, horizontal and vertical shift of the spiral;
- x_J and y_J, horizontal and vertical shift of the jagged line.

We define, for $(x,y) \in [0,a] \times [0,b]$, the functions

$$\iota_S(x,y) = \begin{cases} -\delta_S & (x - x_S, y - y_S) \in \tau_S \dfrac{\min\{a,b\}}{s + w_S} S, \\ 0 & \text{else}, \end{cases} \tag{10}$$

and

$$\iota_J(x,y) = \begin{cases} -\delta_J & (x - x_J, y - y_J) \in \tau_J \dfrac{\min\{a,b\}}{\max\{l, h + w_J\}} J, \\ 0 & \text{else}. \end{cases} \tag{11}$$

The functions ι_S and ι_J represent the carvings of the two glyphs.

If we define ι_S and ι_J as above, then the edge of the carving would be extremely "sharp". Therefore, we wish to smooth it out. Let $k : [0,a] \times [0,b] \to \mathbb{R}$ be a Gaussian kernel with standard deviation σ and let $*$ denote the convolution operator, we define the carved surface as

$$c = b + k * \iota_S + k * \iota_J. \tag{12}$$

4 Description of the Tool

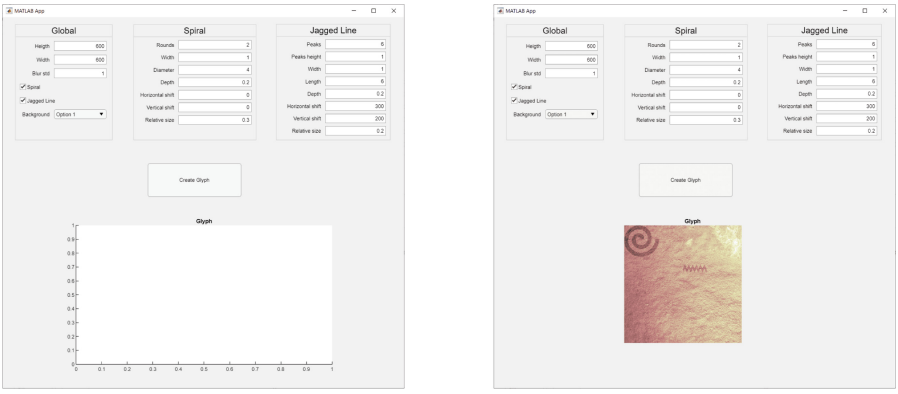
We now describe our tool, which is available at the web page https://github.com/alessandroBuccini/GlyphCreator.
It runs through a Matlab GUI that can be run using the command

```
GlyphCreator
```

in the folder where the package was installed. The package is formed by five files

- `GlyphCreator.mlapp` that contains the Matlab application;
- `createGlyph.m`, `createSpiral.m`, and `createZigZag.m`, that contain auxiliary Matlab functions for running the method;
- `background.mat` that contains the two backgrounds on which we "carve" the petroglyphs.

When launched the application starts as shown in Fig. 7(a). In the top part of the window the user can set the parameters for the creation of the glyphs. The petroglyph is created by using the button `Create Glyph`. This shows, as an image, the petroglyph in the lower part of the window and creates the file `Glyph.mat` containing the sampling of the carved surface as a matrix. This file is stored in the directory where the package is installed. In Fig. 7(b) we show the result obtained with the default parameters. In Table 1 we describe all the parameters that is possible to set in our tool and their default values.



(a) (b)

Fig. 7. GUI for the tool: (a) Opening screen, (b) Default Glyph.

We show two examples of petroglyphs created using our tool in Fig. 8. In the first column we show an example obtained with the first background with both a spiral with many rounds and a thick jagged line. In the second column we did not carve the jagged line and inserted only the spiral. In this case we selected a quite large value for σ, namely $\sigma = 10$. We can observe that in the first example the edges of the glyphs are sharp, while in the second one they are "blurred". We can expect extraction algorithms to struggle more in the latter case.

Table 1. Description of the parameters in the tool.

Block	Field	Description of the field
Global	Height	Height of the carved surface. Since the surface is stored as a matrix this value corresponds to the number of rows of the matrix (default value 600).
	Width	Width of the carved surface. Since the surface is stored as a matrix this value corresponds to the number of columns of the matrix (default value 600).
	Blur std	Standard deviation of the blurring kernel k, i.e., σ, in (12) (default value 1).
	Spiral	Toggle box to determine if the spiral is carved (default value "checked").
	Jagged Line	Toggle box to determine if the jagged line is carved (default value "checked").
	Background	Drop down menu to select between the two possible backgrounds provided with the tool (default value "Option 1").
Spiral	Rounds	Number of rounds of the spiral, i.e., r in (8) (default value 2).
	Width	Width of the spiral, i.e., w_S in (8) (default value 1).
	Diameter	Diameter of the spiral, i.e., d in(8) (default value 4).
	Depth	Depth of the carving of the spiral, i.e., δ_S in (10) (default value 0.3).
	Horizontal shift	Horizontal shift of the spiral, i.e., x_S in (10) (default value 0).
	Vertical shift	Vertical shift of the spiral, i.e., y_S in (10) (default value 0).
	Relative size	Relative size of the spiral with respect to the whole surface, i.e., τ_S in (10) (default value 0.3).
Jagged Line	Peaks	Number of peaks in the jagged line, i.e., n in (9) (default value 6).
	Peaks height	Height of the peaks in the jagged line, i.e., h in (9) (default value 1).
	Width	Width of the jagged line, i.e., w_J in (9) (default value 1).
	Length	Length of the jagged line, i.e., l in (9) (default value 6).
	Depth	Depth of the carving of the jagged line, i.e., δ_J in (11) (default value 0.2).
	Horizontal shift	Horizontal shift of the jagged line, i.e., x_J in (11) (default value 300).
	Vertical shift	Vertical shift of the jagged line, i.e., y_J in (11) (default value 200).
	Relative size	Relative size of the jagged line with respect to the whole surface, i.e., τ_J in (11) (default value 0.2).

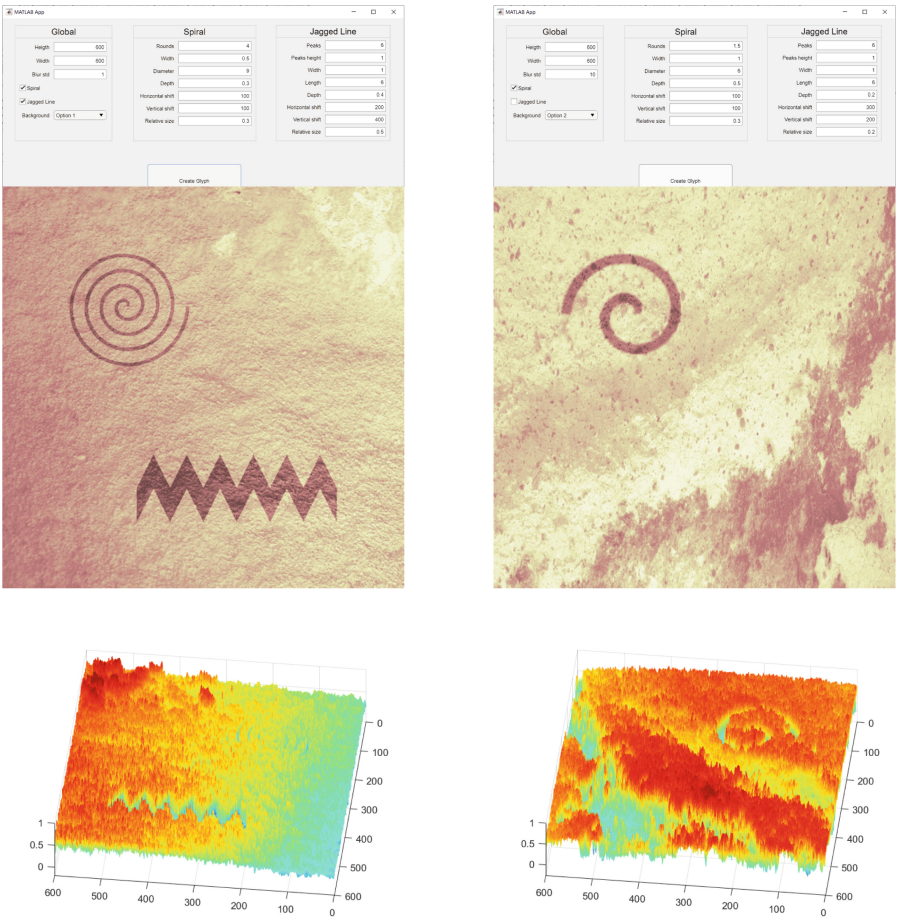


Fig. 8. Examples of petroglyphs created with our tool. Each column collects one example. The first column presents a surface created using the first background with both a spiral and a jagged line with sharp edges, while the second one reports an example obtained using the second background with only a spiral with "blurred" edges. For each example we report in the first row the parameters used, in the second one the obtained surface showed as an image, and in the third row the `mesh` of the surface.

5 Conclusions

We have constructed a Matlab application to construct realistic surfaces engraved with glyphs. The user can select many parameters and create a large quantity of different examples. This tool can be easily applied to validate segmentation algorithms and to generate data to train supervised learning methods. The current version of the tool is tailored towards the petroglyphs commonly present in the Sardinian territories, therefore, we considered only jagged lines

and spirals. In other applications different types of glyphs may be engraved. Future versions of the tool will include the possibility to engrave different types of glyphs and the extension is straightforward.

Acknowledgments. The authors are member of the GNCS group of INdAM that partially founded this work with the GNCS Project "Algebra lineare numerica per problemi di grandi dimensioni: aspetti teorici e applicazioni". Both authors are supported by the PRIN 2022 PNRR project no. P2022PMEN2 "A mathematical approach to inverse problems arising in cultural heritage preservation and dissemination" financed by the European Union - NextGenerationEU and by the Italian Ministry of University and Research (MUR).

References

1. Belhumeur, P.N., Kriegman, D.J., Yuille, A.L.: The bas-relief ambiguity. Int. J. Comput. Vis. **35**(1), 33–44 (1999)
2. Björck, Å.: Numerical Methods for Least Squares Problems. SIAM, Philadelphia (1996)
3. Concas, A., Dessì, R., Fenu, C., Rodriguez, G., Vanzi, M.: Identifying the lights position in photometric stereo under unknown lighting. In: 2021 21st International Conference on Computational Science and Its Applications (ICCSA), pp. 10–20. IEEE (2021)
4. Crabu, E., Pes, F., Rodriguez, G., Tanda, G.: Ascertaining the ideality of photometric stereo datasets under unknown lighting. Algorithms **16**(8), 375 (2023)
5. Durou, J.D., Falcone, M., Sagona, M.: Numerical methods for shape-from-shading: a new survey with benchmarks. Computer Vis. Image Underst. **109**, 22–43 (2008)
6. Golub, G.H., Van Loan, C.F.: Matrix Computations. The John Hopkins University Press, Baltimore, third edn (1996)
7. Hayakawa, H.: Photometric stereo under a light source with arbitrary motion. J. Opt. Soc. Am. A-Opt. Image Sci. Vis. **11**(11), 3079–3089 (1994)
8. Kozera, R.: Existence and uniqueness in photometric stereo. Appl. Math. Comput. **44**(1), 1–103 (1991)
9. Maddalena, L., Gregoretti, F., Antonelli, L.: Petroglyph segmentation: An overview. Tech. Rep. RT-ICAR-NA-2024-01, Institute for High-Performance Computing and Networking, National Research Council of Italy (2024)
10. Mecca, R., Falcone, M.: Uniqueness and approximation of a photometric shape-from-shading model. SIAM J. Imaging Sci. **6**, 616–659 (2013)
11. Woodham, R.J.: Photometric stereo: a reflectance map technique for determining surface orientation from image intensity. In: Image Understanding Systems and Industrial Applications I, vol. 155, pp. 136–143. SPIE (1979)
12. Woodham, R.J.: Photometric method for determining surface orientation from multiple images. Opt. Eng. **19**(1), 139–144 (1980)
13. Zhang, R., Tsai, P.S., Cryer, J., Shah, M.: Shape from shading: a survey. IEEE Trans. Pattern Anal. Mach. Intell. **21**, 690–706 (1999)