An open tool to register landscape oblique images and generate their synthetic model

Authors

- **Timothee Produit**, Ecole Polytechnique Fédérale de Lausanne, *Switzerland*
- **Devis Tuia**, Ecole Polytechnique Fédérale de Lausanne, *Switzerland*

KEYWORDS : Space resection, monoplotter, glacier survey, python

Introduction

Thousands of landscape images stored in archives or in the web are not georeferenced or have a rough georeference (inaccurate 2D location, location name). However, quantitative studies based on images require the precise pose (location and orientation) of the camera. Monoplotting softwares use a Digital Elevation Model (DEM) to assign world coordinates to each pixel of an image. The collinearity equation (Equation 1) transforms the world coordinates in image pixel coordinates. It is used to plot GIS vectors in the image plane. Its inverse function is used to geo-rectify the landscape image. The monoplotter implementation presented in this paper is based on opensource softwares.

User-defined Ground Control Points (GCP) are the common way to register an image. GCP identify corresponding features in the image and in the ortho-image. They are exploited in several ways. First, georeferencing process rectifies the landscape image and maps it on the ortho-image using local transformations between the GCP. However, georeferencing is more suited for aerial nadir images. Its application on landscape image needs a careful and time demanding digitization of many GCP around the study area [1]. Image portions far from the GCP are highly distorted. Second, GCP are exploited as 2D – 3D correspondences, the third dimension is interpolated from the DEM. 2D-3D GCP are used to compute the precise pose of the camera [2]-[7]. Once the pose has been computed, a world coordinate is attributed to each image pixel. Monoplotter softwares have been used to study glaciers [8], mountain landscapes [9] or as teaching softwares for photogrammetry [10].

Monoplotters usually provide the following functions :

- Camera pose estimation based on 2D-3D correspondences;
- Image ortho-rectification;
- Image vector to GIS vector transformation;
- GIS vector to image vector transformation (widely used in Augmented Reality).

The main addition of our software is the computation of a synthetic image, which is the projection of the ortho-image in the image plane. Ortho-image rendering on a DEM is a common 3D-GIS task. Our software offers an open implementation, which does not make use of the graphic card. It focuses on the rendering quality rather than on rendering rapidity.

In the following section we will introduce the mono-plotting concept, the software functions and implementation. Finally, we will show some possible platform usage, based on ancient pictures of the Aletsch glacier (Southern Switzerland).

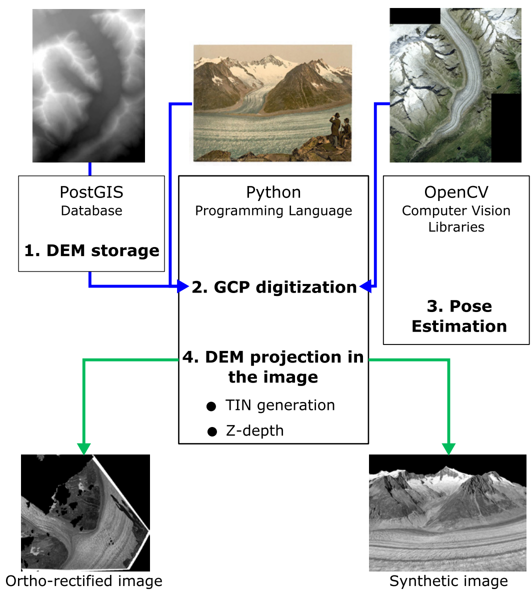


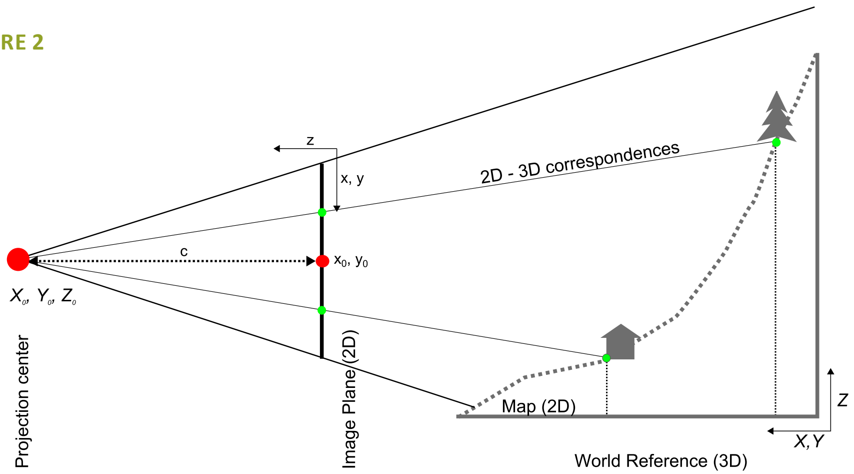
FIGURE 1

Our monoplotting sofware

Mono-plotting

Given a camera pose, the monoplotting software computes the transformations between the image and the 3D model. Figure 2 illustrates the mono-plotting concept. The camera pose is described by the camera location (red point) and orientation. The location (projection center) has coordinates **X0, Y0** and **Z0**. The camera orientation is the angle between the viewing direction along the focal **c** and the **X, Y** plane. In 3D, the camera orientation is described by 3 angles. Each pixel has a correspondence in the 3D model. The projection of the 3D coordinates on the map generates GIS layers (ortho-image and GIS vectors). The projection of an ortho-image in the image plane generates a synthetic image, while projecting GIS vectors generates semantic information.

FIGURE 2



Pose estimation

The pose estimation is the central step of the monoplotting. The 2D – 3D correspondences respectively digitized in the image and in the ortho-image are used to solve the collinearity equation (Equation 1).

In this equation c is the focal, expressed in [mm] or in [pixel]. **x** and **y** are the pixel coordinates. **x0** and **y0** are the coordinates of the principal point (usually the center of the projected image). These parameters describe the camera intrinsic parameters. Extrinsic parameters are **X0**, **Y0** and **Z0**, (the camera coordinates) and orientation of the camera described by a rotation matrix **R**. If the camera intrinsic parameters are unknown, the focal [pixel] is computed from the focal [mm] stored in the image metadata and camera specification found on the web. The principal point is assumed to be at the image center and no image plane deformations are taken into account.

EQUATION 1

$$x - x_0 = -c \cdot \frac{r_{11}\cdot(X-X_0)+r_{21}\cdot(Y-Y_0)+r_{31}\cdot(Z-Z_0)}{r_{13}\cdot(X-X_0)+r_{23}\cdot(Y-Y_0)+r_{33}\cdot(Z-Z_0)}$$

$$y - y_0 = -c \cdot \frac{r_{12}\cdot(X-X_0)+r_{22}\cdot(Y-Y_0)+r_{32}\cdot(Z-Z_0)}{r_{13}\cdot(X-X_0)+r_{23}\cdot(Y-Y_0)+r_{33}\cdot(Z-Z_0)}$$

In computer vision, those parameters are solved in the image coordinates system (**x,y** and **z** along the viewing direction) rather than world coordinates. In our implementation, camera position and orientation are derived from the translation and rotation matrix computed with OpenCV (http://opencv.willowgarage.com). Once pose is known, it can be used in the projective function to compute image coordinates of world objects.

DEM projection in the image

The DEM is projected in the image using Equation 1. This step selects the 3D raster cells which are located in the camera viewing pyramid. In order to populate each pixel with the 3D coordinates of the closest world point, potential 3D raster cells are triangulated with the scientific tool for python (scipy: http://www.scipy.org/). Triangles are projected in the image plane and rasterized in pixel units. The pixels contain a depth value (z-buffer technique [11]), to ensure that the closest world point is the one being attributed at the end of the process. At this time, each pixel consists of a pixel coordinate, an intensity value, a z-depth and a 3D world coordinate.

The pixels on the image boundary are merged to draw the footprint polygon. During ortho-rectification, pixels are mapped on the XY plane and triangulated. As the triangulated pixels are not regularly distributed, intensity values are interpolated on a regular grid, thus providing the orthophoto. Inversely, the assignation of an ortho-image intensity value to the image pixels generates a synthetic image. The ortho-images used for the synthetic image are generated at different scales and slightly blurred in order to generate the most natural synthetic image.

Results

Two images (Figures 3a and 3b) of the Aletsch glacier taken between 1890 and 1900 are downloaded from the **library of congress** (http://www.loc.gov/pictures/). Those pictures belong to a collection called "View of Switzerland". They are taken from the top of the Eggishorn.

FIGURE 3

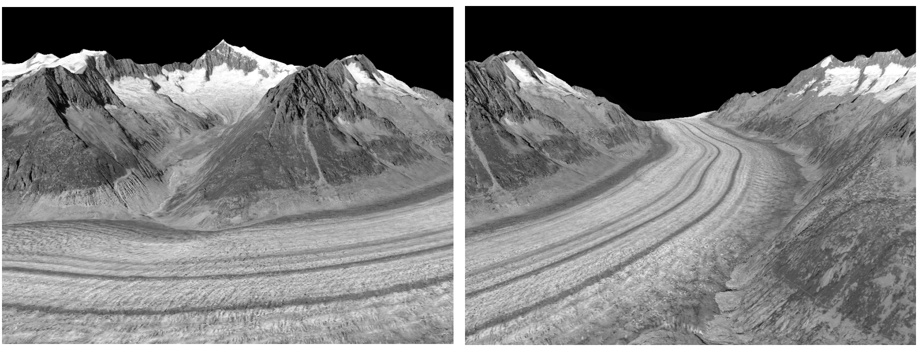
A

B

Ground Control Points are found in the image and in the up-to-date ortho-image. Fortunately, those landscape areas do not change much over 100 years and GCP are stable. However, a trained user can spent between 30 and 60 minutes on each picture. The accuracy and the number of the GCP (7 and 8 respectively) are not sufficient to fully calibrate the camera. The principal point is assumed to be at the image center. The focal that minimizes the reprojection error of each 3D point on the image plane is used. This focal generates a camera location close to Eggishorn peak, which is expected to be the photographer location. The mean reprojection errors are 6.2 and 3.3 pixels respectively.

The camera pose computed is used to project the DEM in the image plane. Synthetic images (Figures 4a and b) are generated, where we can appreciate the changes occurring in the last century. In Figure 4a, the lateral glacier size is greatly reduced and do not reach the main glacier anymore. In Figure 4b, in the lower right corner, the glacier retreats and its height decreases. Note that the mountains in the background are not represented because they are out of the DEM considered for the study area.

FIGURE 4



A B

These ortho-rectified-images are used to evaluate the right bank change between 1900 and 2008. The glacier bound is drawn on the ortho-rectified ancient image and on the current ortho-image. Figure 5 expresses those differences with 4 profiles. The 1900 glacier level is drawn with a green line, the 2008 one is in red. Table 1 summarizes the difference found in the ancient and current glacier state.

TABLE 1

Profil	Horizontal difference [m]	Vertical difference [m]
A	311	82
B	191	120
C	202	83
D	130	44

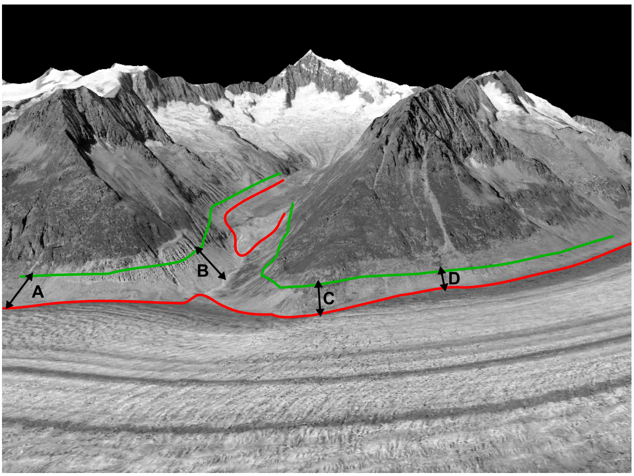


FIGURE 5

Conclusion

Ancient pictures are often used to assess glacier reduction or any environmental change during the last century. The repeated photography technique [12], [13] is widely used to compare those images with a recent image taken from the same location. However, with this process the comparison cannot be quantified. Monoplotting softwares allow such comparison without going to the field, find the right location and take a new picture, which is very challenging for some pictures. Moreover, monoplotters assign a world coordinate to each pixel, thus allowing the quantification of the change.

The python implementation proves to be fast and robust. Scipy and Numpy libraries bring Python close to common mathematic programming languages (Matlab, Octave) that scientist are used to work with. However, since Python is also a general programming language, it is easy to link to open software as openCV. OpenCV is a very active project in computer vision and cannot be substituted by any commercial one. Moreover, Python is widely used in the GIS community and its choice will facilitate interaction. Our own 3D rendering software was needed to support extended and very specific functions. Further development will investigate faster processing on Graphics Processing Units (GPU).

[1] BJORK, A. A., KJAER , K. H., KORSGAARD, N. J., AND KHAN, S. An aerial view of 80 years of climate-related glacier fluctuations in southeast Greenland. *Nature Geoscience* (2012), 427–432.

[2] BOZZINI, C., CONEDARA, M., AND KREBS, P. A new tool for obtaining cartographic georeferenced data from single oblique photos. Proceedings of the *23rd International CIPA Symposium* (Prague, Czech Republic, 2011).

[3] MITISHITA, E. A., MACHADO, A. L., HABIB, A. F. AND GONCALVES, G. 3D monocular restitution applied to small format digital airphoto and laser scanner data. Proceedings of *Commission III, ISPRS Congress* (Istanbul, Turkey, 2004).

[4] JAUREGUI, M., VILCHEZ, J., AND CHACON, L., A procedure for map updating using digital mono-plotting. *Computer & Geosciences* (2002), 513-523.

[5] DOYSTER, Y. AND HALL, J. K., FORTRAN programs for coordinate resection using an oblique photograph and high-resolution DTM. *Computer & Geosciences* (1995), 895-905.

[6] MAKAROVIC, B. Data base updating by digital mono-plotting. *Environmental assessment and resource management* (1983), 433-442.

[7] MAKAROVIC, B. Digital mono-plotters. *ITC Journal* (1973), 583-599.

[8] CORRIPIO, J. G., Snow surface albedo estimation using terrestrial photography. *International Journal of Remote Sensing* (2004), 5705 – 5729.

[9] ASCHENWALD, J., LEICHTER, K., TASSER, E. AND TAPPEINER, U. Spatio-temporal landscape analysis in mountainous terrain by means of small format photography: a methodological approach. *Geoscience and Remote Sensing* (2001), 885-893.

[10] FLUEHLER, M., NIEDEROEST, J., AKCA, D., Development of an educatioal software system for the digital monoplotting. Proceedings of *ISPRS on Tools and Techniques for E-Learning*. (Potsdam, Germany, 2005).

[11] COOK, R. L., CARPENTER, L., AND CATMULL, E. The Reyes image rendering architecture. *ACM SIGGRAPH Computer Graphics* (1987), 95-102.

[12] ROUSH, W., MUNROE, J. S., AND FAGRE, D. B. Development of a spatial analysis method using ground-based repeat photography to detect changes in the alpine treeline ecotone, Glacier National Park, Montana, USA. *Arctic, Antarctic, and Alpine Research* (2007), 297-308.

[13] KULL, C. Historical landscape repeat photography as a tool for land use change research. *Norwegian Journal of Geography* (2005), 253-268.