

SOFTWARE

MicMac – a free, open-source solution for photogrammetry

Ewelina Rupnik^{1*}[†], Mehdi Daakir^{1,2} and Marc Pierrot Deseilligny^{1,2}

Abstract

The publication familiarizes the reader with **MicMac** – a freeware, open-source photogrammetric software. A brief history of the tool, the organisation of the software and its unique features *vis-à-vis* other software tools are in the highlight.

The essential algorithmic aspects of the *structure from motion* and image dense matching problems are discussed from the implementation and the user's viewpoints. The work finalizes by reviewing the tools which serve to interact with the user, i.e. image measurement, visualization and data conversion tools.

Keywords: freeware; open-source; photogrammetry; structure from motion; bundle adjustment; semi-global image dense matching

Introduction

Photogrammetry is the art, science, and technology of obtaining geometric information on the 3-dimensional shape and orientation of objects from photographs and other imaging sensors. It is a cheap measurement methodology as it can be executed by practically any digital camera of a decent make. It is instantaneous as it captures the observed phenomena at once and in a split second. It is automated to a high degree due to the advent of the digital era and the rapid developments in automated data processing, hence accessible to non-expert users. The milestones that rejuvenated photogrammetry, brought it forward and placed among other competitive remote sensing techniques (e.g. Light Detection and Ranging LiDAR) merits to the discipline of computer vision [1]. Among several milestones leading to this progress are the automated

interest points detection [2], the *Structure from Motion* (*SfM*) algorithms capable of reconstructing scenes from sets of unordered image collections [3, 4], and the dense image matching techniques delivering surface models of resolution equal to the pixel size on the ground [5], [6].

All this contributes to an ever growing visibility of the photogrammetric tools across various fields of science and engineering, the growing market interest, and subsequently a multitude of photogrammetric/-computer vision libraries and software solutions, be it commercial or free/open-source. **MicMac** – together with *Bundler*, *PMVS*, *VisualSfM*, *OpenCV* and others – belongs to the open-source solutions. This publication aims at familiarizing the reader with the philosophy behind **MicMac**, some of its crucial algorithmic aspects, the software architecture and the pool of available tools. The paper is organised as follows

Historical notes

MicMac has been developed at the *National Institute of Geographic and Forestry Information* (IGN) and the *National School of Geographic Sciences* (ENSG), since 2003. Initially, the software tools were developed having in mind exclusively the IGN's cartographic production. The independent tools were interfaced in 2005 via an XML framework, allowing the user to freely parametrize the calculations at all processing stages. In 2007, IGN began to freely distribute **MicMac** under the CECILL-B license that is a version of the L-GPL license adapted to the french law. Until 2008, the dense image matching of already oriented images was possible only with the IGN's internal image file format. In the same year the *Apero* tool was added to the software kernel, offering from now on the possibility to estimate camera exterior and interior orientations, with no restriction on the formatting. In 2010, the XML interface was replaced by a simplified command line. This evolution contributed to an improved accessibility, diffusion and subsequently a better visibility of the software in the scientific communities, and the general public.

From 2010 on, **MicMac** has undergone significant evolution due to its implication in many french and european-based projects. On top of the scientific contributions to the source code, the additional funding

*Correspondence: ewelina.rupnik@ensg.eu

¹Department of Zoology, Cambridge, Waterloo Road, London, UK
Full list of author information is available at the end of the article

[†]Equal contributor

brought about software distributions under Windows, Mac OSX, as well as the GPU processing for certain tasks (e.g. multi-image correlation, optimization with dynamic programming).

MicMac vis-à-vis other tools

From the architecture point of view, the unique feature of **MicMac** with respect to its alternatives is that the user can operate on different software *niveau*. The least experienced user can steer the processing by a simple command line where little or no parameters must be set. A user with a good command of photogrammetry can operate from the same command line by adapting the parameters typically filled with default values. An expert user can have access to every single parameter at any processing step via a set of XML files. On top of that, developers and scientists can use **MicMac** as a library, implement their own algorithms, and share them for the benefit of photogrammetric community. This said, **MicMac** was designed primarily as a tool for professionals and consequently is far from a model design in the sense of HCI^[1].

MicMac as a photogrammetric tool stresses the metrological aspect of the reconstruction problem, and provides tools typically unavailable in existing software alternatives, for instance:

- access to intermediary results in open data formats allowing to interact with the processing chain at any desired stage,
- qualitative evaluation of the results via quality indicators (e.g. bundle adjustment residuals, correlation images),
- a wide range of camera calibration models,
- two-dimensional dense image matching for deformation studies,
- processing of frame cameras and pushbroom sensors,
- processing of scanned analogue images,
- architecture adapted to big datasets (up to tens of thousands images)

Software organisation

MicMac is organised in a number of modules, accessible through a common command `mm3d`. Execution with a unique command has a twofold advantage of (a) allowing for factorization of certain developments while keeping the binary files compact, and from the point of view of a user, (b) it is easy to remember giving instantaneous access to all modules. For instance, typing the `mm3d` will print the list of the modules at the disposal, c.f. Fig. 3. Invoking a selected module is followed by a number of parameters. The obligatory parameters must be specified at first and since they

do not carry names, their order of occurrence is important. In the example shown in Fig. 4, the obligatory parameter would be the format specification, and the change coordinate file `ChSys` serves as an option.

The algorithmic aspects

The photogrammetric workflow encompasses the passage from images, through the estimation of the orientation parameters, all the way to a 3D surface model. In other words, it is a passage from a 2D representation of the world captured by a camera, through inference of the position and rotation of that camera at the moment of the image taking, towards a 3D restitution of the lost dimension. As long as the quality of the final output depends on the skillful engineering of many small processing steps, estimation of the camera orientation parameters and the matching algorithms constitute the heart of the pipeline. Respectively, the coming sections concentrate on these two aspects, report on the adopted methods and give a global look on what is possible in **MicMac**.

Structure from motion

Recovery of the structure (i.e. 3D coordinates of the observed scene) and the motion (i.e. the camera interior and exterior orientation parameters) is essentially an estimation problem, where the imaging equation constitutes the model. It is well known that this model (i.e. the transformation from 3D to 2D space and vice-versa also referred to as collinearity equation) is not linear, hence requires linearization. Moreover, there exists no direct algorithm able to compute orientation parameters globally consistent with a number of images (generally $n > 3$). To overcome this gap, bootstrap solutions were proposed. Using direct algorithms for a single image, a pair or triplet of images, the global orientation is deduced sequentially [3] or hierarchically [7] starting from a seed image pair. The so-obtained parameters serve as input to a system of equations composed of the linearized collinearity equations, where their optimal values – in the stochastic sense – are found iteratively. The observations (e.g. tie points) are redundant therefore the solution is deduced with the least squares method by minimizing an objective function. The typical function is the squared differences between the nominal observations and those predicted from the estimated model, possibly subject to constraints. The latter stage is also known as the bundle adjustment (BA).

MicMac implementation. **MicMac** solves the BA with the Levenberg-Marquardt (L-M) method. The L-M is in essence the Gauss-Newton method enriched with a damping factor to handle rank-deficient (or nearly so)

^[1]Human-computer interaction

Jacobian matrices [8]. A range of different observations equations, constraints and unknowns (i.e. parameters) are allowed in the adjustment routine, among them:

- tie points (adjustment with points),
- lines (adjustment with lines; added in the context of georeferencing aerial photographs with a 3D database of roads),
- perspective centers derived from direct georeferencing (i.e. GNSS),
- ground control points (GCP),
- the lever-arm vector separating the GNSS antenna and the perspective center,
- rigid relative orientation between cameras (i.e. the rigid block; useful for multi-camera systems)

On top of that, a broad spectrum of camera calibration models are accessible. Both argumentation are supported, the physical and the phenomenological. The former models any imaging error attributed to a phenomenon of a physical nature, while the latter models solely the effects of the lens imperfections, without interrogating its causes. The total distortion is defined as a composition of predefined elementary distortions, such as radial, decentric, polynomial, fish-eye etc., however, typically the major distortion are modeled by the physical model, while the polynomial models remove the less significant remaining systematism.

The critical element of the BA's mathematical model is the observation weighting, known as the stochastic model. **MicMac** specifies three different weighting strategies. The first strategy corresponds to the weighting in the classical model known in photogrammetry – the Gauss-Markov – where the observations are weighted by their true standard deviations known *a priori*. The second strategy controls the influence of a particular category of observations so as to avoid solutions that are driven by a single category only for the sheer reason of its abundant observations. The last strategy addresses the robustness. The weight is a function of the observation's residual in the BA, therefore giving more credibility to observations that are close to the model being estimated, and contrary, limiting the influence of observations with high residuals.

User's viewpoint. There are two principal modules that handle the camera orientation recovery with the simplified interface – **Tapas**, **Campari** – and either of the modules calls the *parent* orientation tool **Apero** (cf. Fig. 1). **Tapas** calculates the purely relative orientation of images, using observed tie points as the only input. Since at this point there is no *a priori* on positions and rotations of the cameras, **Tapas** also entails the initialization step where it progressively reconstructs the cameras with the help of the direct orientation algorithms, and intertwines it with the BA routine.

Unlike in **Tapas**, **Campari** is a BA module adapted to handling heterogeneous observations. Besides the tie points, it works with GCPs, the GNNS data and can impose various constraints on the solution. This module is typically executed once a good relative orientation has been established, and, all input data shall be defined in a consistent coordinate system (CS). To bring the cameras' relative orientation to the same CS of e.g. the GCPs, a module performing spatial similarity transformation (e.g. **GCPBascule**; c.f. Fig. 1) must be evoked beforehand. Both, **Campari** and all variations of the **Bascule** can be regarded as georeferencing tools.

Multi-view stereo image matching (MVSM)

Given several overlapping images with known orientation parameters, MVSM is the process of reconstructing a complete 3D model of the scene by finding corresponding pixels (i.e. their disparities or depths, respectively) between image pairs, and triangulating them in space. The generic algorithm is defined as an energy minimization problem that searches for a disparity map that minimizes the energy. The minimization can be solved with local, semi-global and global algorithms. Today the golden standard method for producing dense 3D models from images is the semi-global matching [6].

Image matching breaks down into three processing stages: computation of the pixel matching cost, cost aggregation and disparity calculation. The matching cost is a measure of similarity, i.e. describes the likelihood that two pixels do belong to a unique point in 3D. As the matching can be ambiguous, and the minimum cost insignificant, an additional *a priori* is imposed on the energy function that penalizes changes in disparities of neighboring pixels (the case of local and semi-global algorithms). This aggregation takes place within a window, a 1D path (locally) or multiple paths (semi-globally). In the latter case, the cost for a given disparity is an accumulated cost along many paths that end in the pixel in question [9].

MicMac implementation Image dense matching in **MicMac** is possible with a semi-global and global algorithms. The objective of matching is to find a mapping \mathcal{F}_{px} from the space τ where the depth is computed, to the disparity space ε_{px} . Depending on the restitution geometry, the τ can be the euclidean space or the image space of the master image, while the ε_{px} can be either the disparity or the depth of the euclidean \mathcal{Z} .

The motivation to distinguish between these two restitution geometries – the ground and the image – is twofold. Firstly, a Digital Surface Models (DSM) produced from aerial images is normally defined in

some reference CS, therefore it is more intuitive to perform the reconstruction directly in the target CS (i.e. ground geometry; alternatively one can use an arbitrary CS and transform the result *a posteriori*). Here, the disparity space ε_{px} is the euclidean \mathcal{Z} (cf. Fig. 2(A)). On the other hand, it is known that matching in image space with a master and a set of slaves images is more reliable, especially in close-range photogrammetry applications and for small datasets. The disparity space ε_{px} in the image geometry is then either the depth along the ray or the respective disparity in image space (i.e. the image geometry; cf. Fig. 2(B) and 2(C)). It is up to the user which geometry to employ.

MicMac offers a very flexible formulation of the matching cost function as far as its terms, the optimization algorithm and the dimension of the matching are regarded (cf. Eq. 1).

$$\mathcal{E}(\mathcal{F}_{px}) = \iint_{\mathcal{T}} \mathcal{A}(x, y, \mathcal{F}_{px}(x, y)) + \|\nabla(\mathcal{F}_{px})\|^{reg} \quad (1)$$

$$\|\nabla(\mathcal{F}_{px})\|^{reg} = \alpha_1 * |\nabla(\mathcal{F}_{px}^1)| + \alpha_2 * |\nabla(\mathcal{F}_{px}^2)| \quad (2)$$

where

$\mathcal{A}(x, y, \mathcal{F}_{px}(x, y))$ is the measure of similarity between pixels with $\mathcal{A} = 0$ when they are identical; $\|\nabla(\mathcal{F}_{px})\|^{reg}$ a norm on the gradient, which is used as a regularity criteria (it penalizes the variation of \mathcal{F}_{px}); α_1 is the regularization on first component of disparity and α_2 the regularization on the second component (equivalent of matching in the direction transverse to the epipolar line).

The $\|\nabla(\mathcal{F}_{px})\|^{reg}$ controls the *a priori* on the disparity space in a way that:

- if the desired model is smooth, a convex F is adequate (it's better to climb a given jump by regular step),
- if the desired model has many discontinuities, a F concave is adequate (it's better to climb a given jump in one single step),
- when there is no strong prior, the default choice is to have F linear,
- if *a priori* knowledge on scene slope exists, it can impose an allowable maximum scene slope,
- for 2D matching, non-isotropic smoothing factors can be set.

The actual similarity measure $\mathcal{A}(x, y, \mathcal{F}_{px}(x, y))$ is the normalized cross correlation coefficient, defined as a function of multiple images, rather than a single pair. The coefficient can privilege an image out of a set of images (e.g. a master image), or consider the set as

"symmetrical". Varying cross correlation window sizes and weighted windows are also possible.

To find the disparity map that minimizes the energy, **MicMac** by default employs a multi-directional variant of the dynamic programming algorithm. The results along independent directions are aggregated by taking a mean of all results or simply the best result. Optionally, a global minimization by the Min-Cut/Max-Flow algorithm [10] can be enforced.

In order to limit the disparity search-space, speed up the calculation and to reduce the noise, **MicMac** adopts a multi-resolution matching approach where coarse resolution disparity maps serve as predictors of the fine resolution output.

User's viewpoint DSM creation in **MicMac** takes places via several tools that are semi- or fully-automated. In both cases the *parent* tool to handle the low-level dense matching is the **MicMac**, which an expert user can access via an XML file.

Malt, the semi-automated tool operates in three pre-defined modes, i.e. **Ortho**, **UrbanMNE** and **GeomImage** that are adapted to producing orthophotography, urban DSMs in ground geometries and DSMs in image geometry. Accordingly, **Ortho** and **UrbanMNE** would be used for flat terrains or planar surfaces (i.e. 2.5D; c.f. Fig. 6, 7, 9) whereas **GeomImage** is more suitable for true 3D surfaces (c.f. Fig. 5).

The inconvenience of the **Malt GeomImage** is that for a complete 3D restitution of a surface, one must indicate several master and their relevant slave images. As long as for simple object geometries it is feasible, it becomes inefficient for more complex and extensive objects. The fully automated tools overcome this impediment by making this choice for the user, while prioritising completeness and precision of the output. The two automated tools are **C3DC** and **PIMs** which furnish a point cloud and a set of depth maps, respectively. To proceed with the orthophotomap generation, the **PIMs2MNT** and subsequently the **Tawny** shall be executed (an exemplar orthophoto in Fig. 8). The latter collects the per-image orthophotos and merges it to a single mosaic. The individual orthophotos can be created either by **PIMs2MNT** tool or **Malt**; c.f. Fig. 1.

Besides the classical 1D matching (along an image ray, an image row or the Z-coordinate) for reconstruction of object geometry, **MicMac** also implements a 2D matching strategy. This strategy is useful in 2D deformation studies between images (c.f. Fig. 7; also **MM2DPosSism**), as an orientation quality estimation tool to asses the remaining y-parallax (**MMTestOrient**; c.f. Fig. 9), or in cases when orientation parameters are unknown or known with poor precision (executed with XML file). Because the expected disparities pertinent to geometry are of higher frequencies than those

along the y-parallax, the regularization of the energy function in the two direction is managed separately; c.f. Eq. (2).

Lastly, in the event when all camera optical centers are located on a 3D line, parallel to the surveyed object, and the reconstruction is defined in object space, the matching becomes ambiguous along the epipolar lines. To avoid the ambiguities and the resultant noise, MicMac calculates the matching in ortho-cylindrical coordinates (see RepLocBascule).

Interactive tools

Image measurement tools

Visualization tools

Data exchange with other software

Competing interests

The authors declare that they have no competing interests.

Author's contributions

Text for this section ...

Acknowledgements

Text for this section ...

Author details

¹Department of Zoology, Cambridge, Waterloo Road, London, UK.

²Marine Ecology Department, Institute of Marine Sciences Kiel, Düsternbrooker Weg 20, 24105 Kiel, Germany.

References

1. Granshaw, S.I., Fraser, C.S.: Editorial: Computer vision and photogrammetry: Interaction or introspection? *The Photogrammetric Record* **30**(149), 3–7 (2015)
2. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* **60**(2), 91–110 (2004). doi:10.1023/B:VISI.0000029664.99615.94
3. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: exploring photo collections in 3d. In: *ACM Transactions on Graphics (TOG)*, vol. 25, pp. 835–846 (2006). ACM
4. Pierrot-Deseilligny, M., Clery, I.: Apero, an open source bundle adjustment software for automatic calibration and orientation of set of images **38**(5), 269–276 (2011)
5. Pierrot-Deseilligny, M., Paparoditis, N.: A multiresolution and optimization-based image matching approach: An application to surface reconstruction from spot5-hrs stereo imagery. *Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* **36**(1/W41) (2006)
6. Hirschmuller, H.: Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence* **30**(2), 328–341 (2008)
7. Toldo, R., Gherardi, R., Farenzena, M., Fusiello, A.: Hierarchical structure-and-motion recovery from uncalibrated images. *Computer Vision and Image Understanding* **140**, 127–143 (2015)
8. Nocedal, J., Wright, S.: *Numerical Optimization*. Springer, ??? (2006)
9. Szeliski, R.: *Computer Vision: Algorithms and Applications*. Springer, ??? (2010)
10. Roy, S., Cox, I.J.: A maximum-flow formulation of the n-camera stereo correspondence problem. In: *Computer Vision, 1998. Sixth International Conference On*, pp. 492–499 (1998). IEEE

Figures

```

1 :~$ mm3d
2 mm3d : Allowed commands
3 AllDev Force devlopment of all tif/xif
4 Ann Matches points of interest
5 AperiCloud Visualization of camera in ply
6 Apero Compute external/internal
   orientations
7 BatchFDC Tool for batching a set of commands
8 ...

```

Figure 3 mm3d is the common command that invokes all MicMac modules and their brief description.

```

1 :~$ mm3d OriConvert -help
2 ****
3 * Help for Elise Arg main *
4 ****
5 Mandatory unnamed args :
6   * string :: {Format specification}
7   * string :: {Orientation file}
8   * string :: {Targeted orientation}
9 Named args :
10  * [Name=ChSys] string :: {Change coordinate
     file}
11  * [Name=Calib] string :: {External XML
     calibration file}
12  * [Name=AddCalib] bool :: {Try to add
     calibration, def=true}
13 ...

```

Figure 4 Each module is executed with some obligatory and optional parameters. Adding -help at the end will print out the expected input arguments.

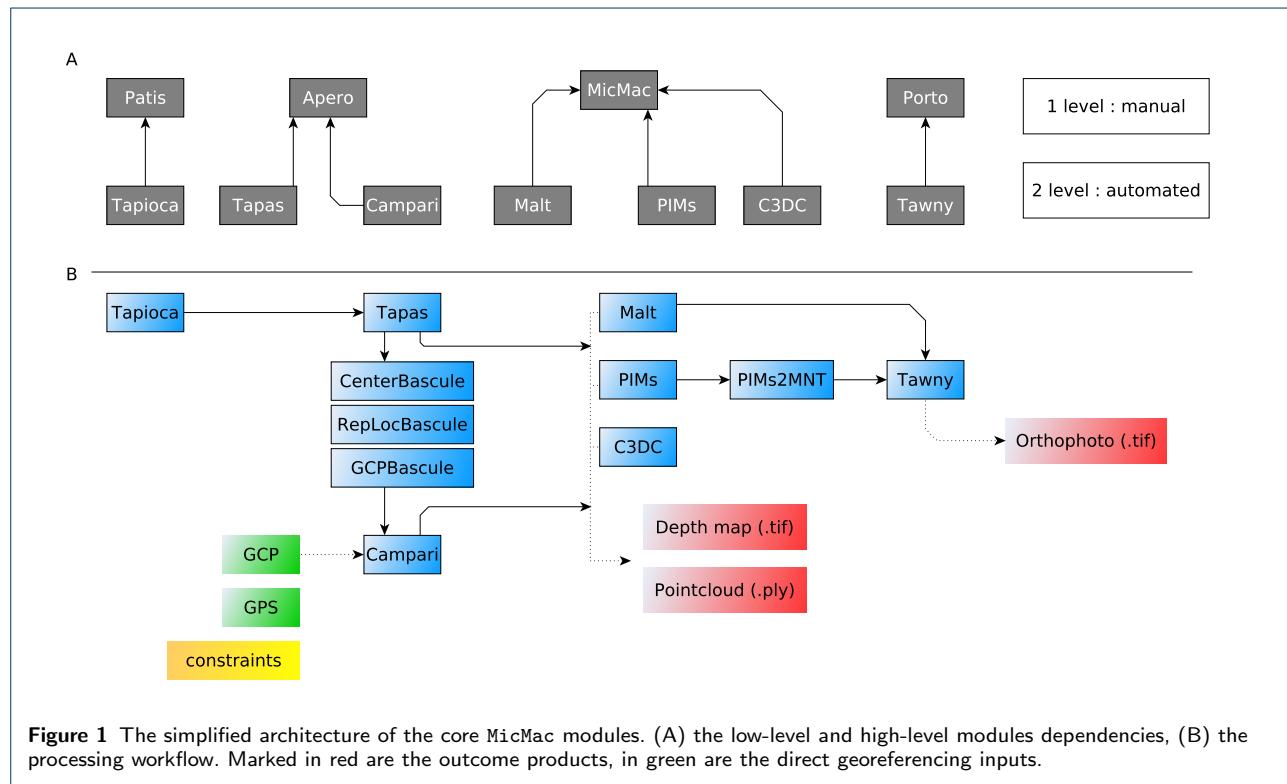


Figure 1 The simplified architecture of the core MicMac modules. (A) the low-level and high-level modules dependencies, (B) the processing workflow. Marked in red are the outcome products, in green are the direct georeferencing inputs.

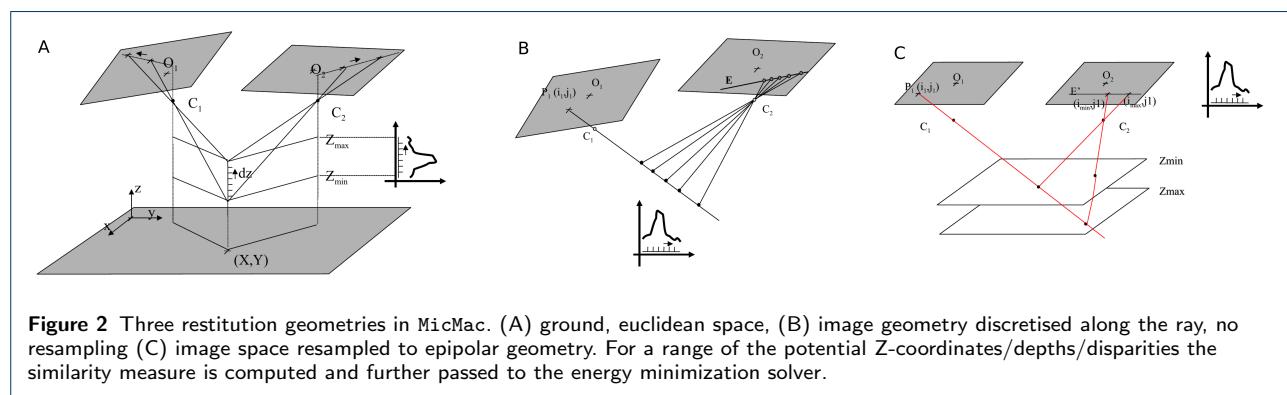


Figure 2 Three restitution geometries in MicMac. (A) ground, euclidean space, (B) image geometry discretised along the ray, no resampling (C) image space resampled to epipolar geometry. For a range of the potential Z-coordinates/depths/disparities the similarity measure is computed and further passed to the energy minimization solver.

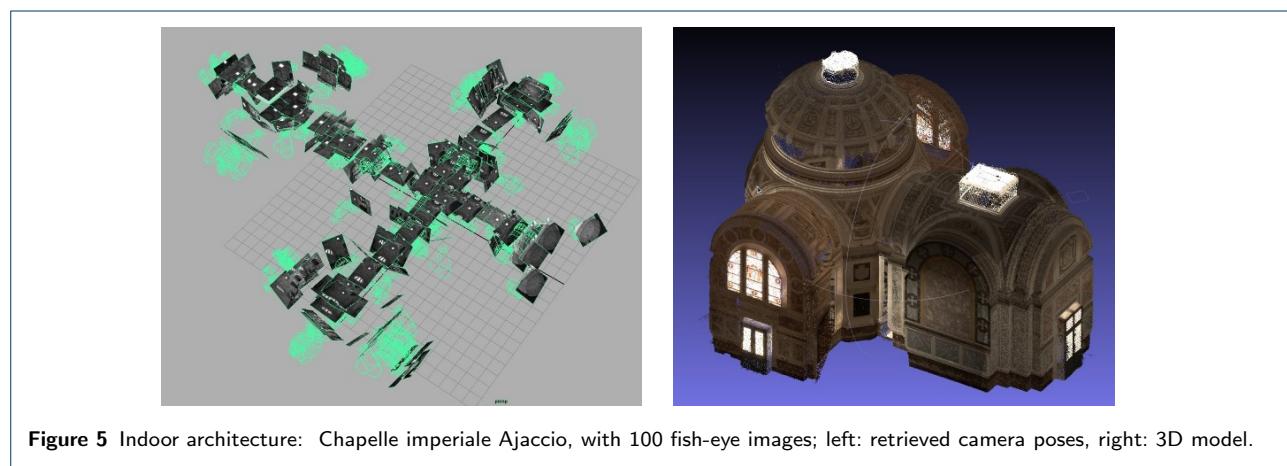


Figure 5 Indoor architecture: Chapelle imperiale Ajaccio, with 100 fish-eye images; left: retrieved camera poses, right: 3D model.

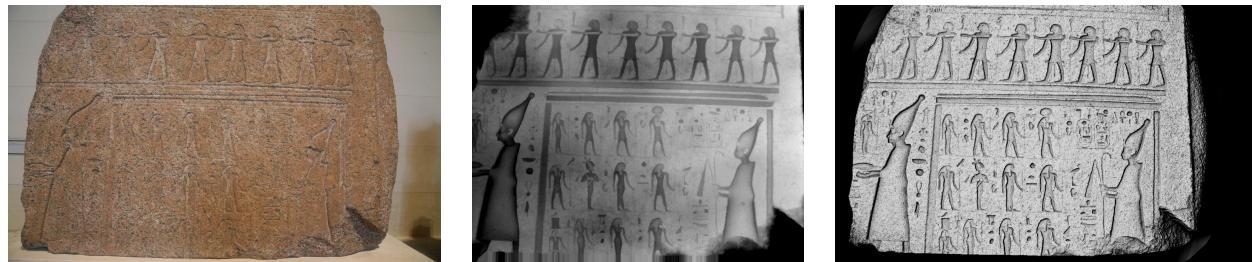


Figure 6 Bas Relief stone in Louvre: the image and the 3D model rendered in shading and depth map.

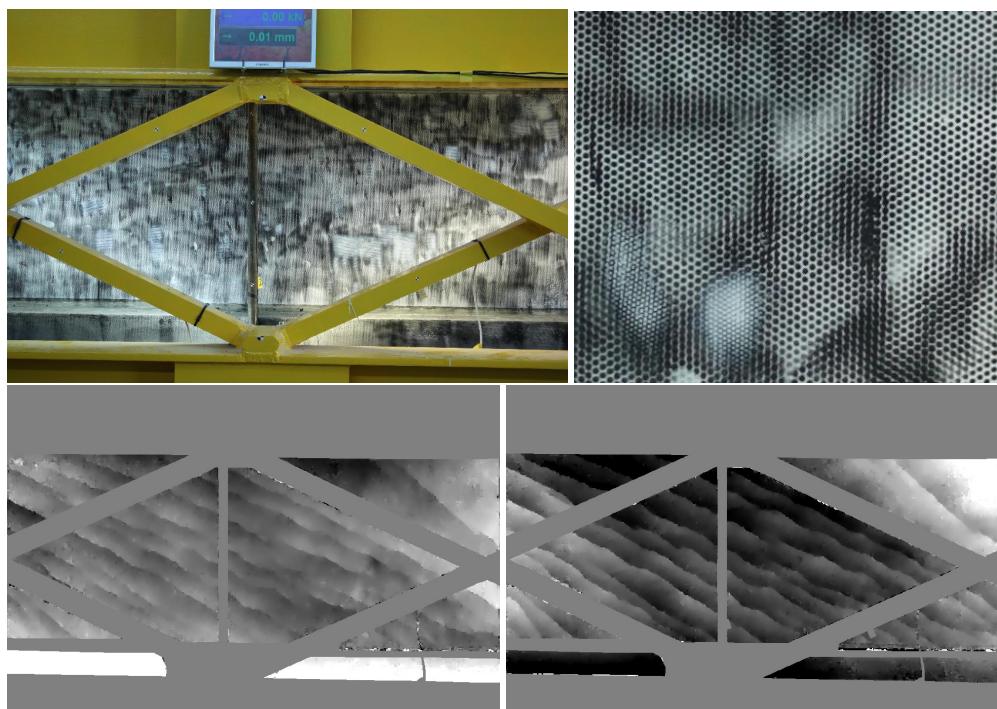


Figure 7 Under induced force the concrete beam deformed, while a still camera measured the displacements in during the breaking phase. Upper left: the full view image, upper right: a close-up of the concrete structure, bottom left: displacement in the x-coordinate (max amplitude $\frac{1}{4}$ pixel), bottom right: displacement in the y-coordinate (max amplitude $\frac{1}{4}$ pixel)



Figure 8 Forteresse de Salses, photo acquired from a drone, in collaboration with Map-CNRS/ Left: hypsometry and shading maps overlapped, right: an orthophotography.



Figure 9 Munich dataset, acquired with a DMC camera at GSD of 10 cm. Left: the depth map, right: the y-parallax generated with MMTestOrient. The amplitude of transverse parallax is ± 0.1 pixel