

3D Map Computation from Historical Stereo Photographs of Florence

**Marco Fanfani, Fabio Bellavia, Giulio Bassetti,
Fabrizio Argenti, Carlo Colombo**

Department of Information Engineering (DINFO), University of Florence,
Via S. Marta 3, 50139, Florence, Italy

E-mail: {marco.fanfani, fabio.bellavia, fabrizio.argenti, carlo.colombo}@unifi.it,
giulio.bassetti@stud.unifi.it

Abstract.

The analysis of early photographic sources is fundamental for documenting and understanding the evolution of a city so rich in history and art as Florence. Indeed, by the 1860s several photographers used to work in town, and their images (often obtained through stereoscopic set-ups) can help us to reconstruct Florence in 3D as it was by the time of the Italian unification. The first and most delicate part of such reconstruction process is the computation of *disparity maps* from the historical stereo pairs. This is a very challenging task for fully-automatic computer vision algorithms, since XIX century photographs are affected by several problems—ranging from superficial damages to asynchronous acquisition—that are usually absent in modern images. In this paper we describe a semi-automatic method that, through minimal user input, allows the creation of coherent and realistic 3D maps of florentine scenes.

1. Introduction

Thanks to its immediate readability and faithful reproduction accuracy, photographic material from the last two centuries is an invaluable source of information for historians of art and architecture willing to track the changes occurred to the urban space and its monuments over the years. In the early days of photography (mid-19th century), several professional photographers were based in Florence, where they used to produce and sell various kinds of printed images, ranging from commissioned portraits to selections of remarkable city corners and works of art, which were particularly appreciated by tourists from both sides of the Atlantic. Anton Hautmann was one of the most active photographers in Florence between 1858 and 1862. He was a specialist in stereoscopic photography, a fashionable new technology at that time, by which an image pair of the same scene is obtained from two slightly different viewpoints. His “stereographs” were particularly sought-after by amateurs, who were thus able to recreate the depicted scene in 3D by watching the photographs through a simple stereoscope.

The project TRAVIS (Tecniche di Realtà Aumentata per la Visualizzazione di Immagini Storiche), currently carried out by the Computational Vision Group of the University of Florence and funded by Fondazione Cassa di Risparmio di Firenze, is aimed at developing effective computer vision techniques for the 3D reconstruction and rendering of (either single or stereoscopic) historical photographs of Florence. Its main goal is to develop a geo-referenced PC



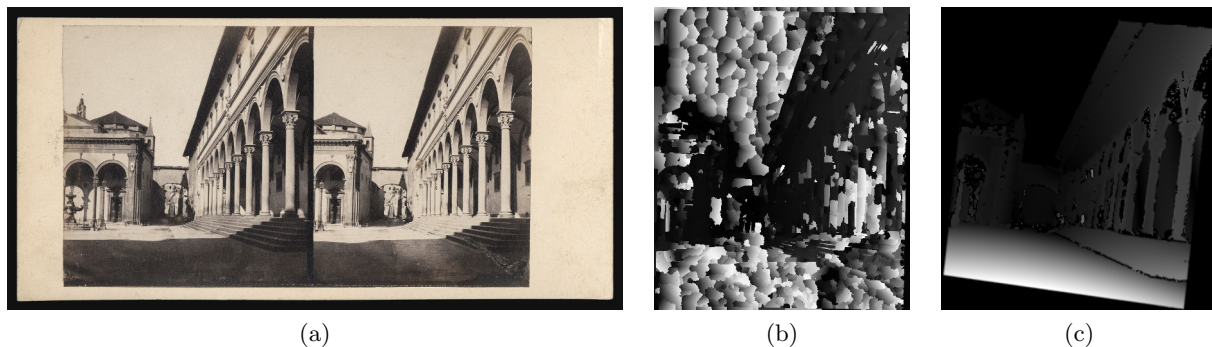


Figure 1: (a) Digital scan of an original stereograph by A. Hautmann depicting a corner of Piazza Santissima Annunziata in Florence; (b) Disparity map obtained with a fully-automatic state-of-the-art software [1]; (c) Result obtained with the proposed method. (Better viewed on a computer screen.)

or mobile app allowing users to navigate through Florence as it was in the XIX century, being able to switch anytime to contemporary Florence, and mark the changes occurred in its urban structure.

While 3D reconstruction from images is a widely explored topic in photogrammetry and computer vision, its application to historical images is still challenging. Some published works describe solutions based on manual 3D modeling using commercial softwares [2, 3], others exploit projective constraints on manually selected features [4]. In [5] an automatic Structure from Motion (SfM) method is employed, leveraging on a huge data collection—800 photos from which the authors were able to select a small sub-set (44 images) on which computer vision algorithms work reliably. A similar problem is described in [6], where automatic methods can deal with images spanning only 1956 to 1975, while using manual input the authors are able to register images from 1864 to 2008.

Differently from other approaches, in this paper focuses on the dense stereo analysis of Hautmann’s stereographs—see Fig. 1a. Being the first and most important step toward dense 3D reconstruction, dense stereo analysis is a well-known topic in computer vision, and many algorithms have been presented so far to solve the problem [7]. However, as SfM fails, so do even the best automatic dense stereo methods that are not able to deal satisfactorily with early stereographs (see Fig. 1b). These images are typically—and often simultaneously—affected by several problems not occurring in modern images: Low resolution, blurred areas, physical damages as scratches or stains, low contrast, asynchronous acquisition. For this reason, we developed a semi-automatic method that, by leveraging minimal user input, can obtain coherent and realistic *disparity maps* from historical photo pairs (see Fig. 1c). This is achieved by computing an initial disparity map, which is then improved through manual input and the use of planar projective transformations, according to the observation that urban man-made environments are mainly composed by planes.

The next Section briefly presents Hautmann’s stereographs collection. Sect. 3 introduces basic dense stereo terminology and concepts. Section 4 presents the developed StereoDisparity App. Finally, in Sect. 5 conclusions are given and future work is outlined.

2. Hautmann’s Stereographs Collection

Anton Hautmann (Munich 1821 – Florence 1862) was one of the most active photographers in Florence between 1858 and 1862. He left us with one of the largest photographic collections of Florence, characterized by both a high technical quality and a noticeable artistic skill. His



Figure 2: Examples of stereo photographs from H. Hautmann: note how all frames are affected by noise, presenting saturated areas, low contrast, different exposition between the left and right frames. Stains and scratches are clearly visible, as well as vignetting effects. In (a), a view of Piazza Santissima Annunziata; (b) Palazzo Medici Riccardi in Via Larga (today Via Cavour); (c) and (d) two views of Piazza Santa Maria Novella. Notice also that (a) is an example of asynchronous acquisition: in the right frame there is a carriage near the statue, absent in the left frame.

florentine views encompass a wide range of subjects (from palaces and squares to street life scenes), often taken from unusual yet effective viewpoints. Part of Hautmann's collection (123 stereoscopic photographs) is described in [8]. Thanks to Hautmann's living heirs, we were able to collate 174 stereographs of Florence and other places in Tuscany. High resolution digital scans of the original stereo pairs were then acquired in cooperation with the Kunsthistorisches Institut in Florence. Figure 2 shows some examples of Hautmann's stereographs collection. Observing the photographs, it's easy to notice that the image quality is very poor. This is partly due to the limited photographic instrumentation available in the XIX century. Most photos are quite noisy and have poor definition, low contrast, saturated or blurred areas. Moreover, during the years, some photographs were partially damaged with scratches and stains. It is also worth noting that some stereo pairs were acquired with a single camera moved in two positions along a rail (asynchronous acquisition), instead than with a real stereo camera with two lenses: this can be easily detected by observing that some objects in the scene change their position in the two frames—this is a serious challenge for any automatic stereo matching algorithm.

3. Dense Stereo

Given two images of the same scene, it is possible to define a set of constraints, referred to as *epipolar geometry*, that can be exploited in order to compute the 3D structure of the scene [9, 10]. Let \mathbf{X}_1 be a 3D point in the scene, and $\mathbf{x}_1^l, \mathbf{x}_1^r$ be its projection onto the left and right stereo frames, respectively. Let \mathbf{X}_2 be a second 3D point, collinear with \mathbf{X}_1 and the left camera center \mathbf{C}_l . Under this setup, the projection of \mathbf{X}_2 will perfectly overlap with \mathbf{x}_1^l on the left frame, while on the right frame \mathbf{x}_2^r it will fall away from \mathbf{x}_1^r (see Fig. 3a). This displacement between \mathbf{x}_1^r and

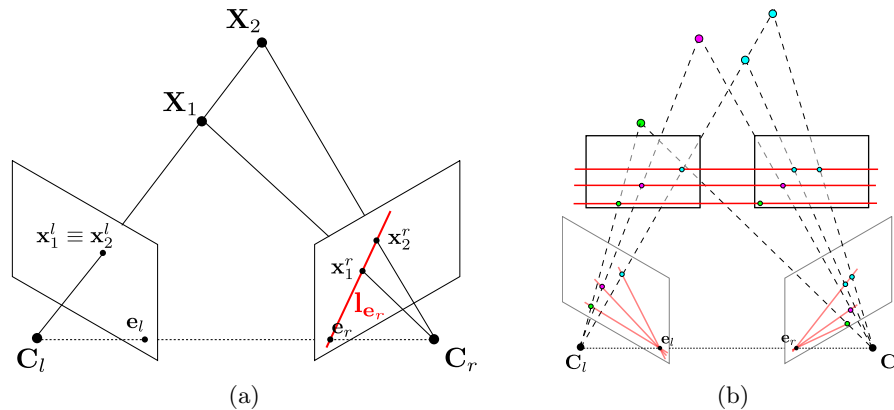


Figure 3: Epipolar geometry: (a) *Motion parallax*. \mathbf{X}_1 and \mathbf{X}_2 are collinear w.r.t. the left camera, so that \mathbf{x}_1^l and \mathbf{x}_2^l are coincident, while \mathbf{x}_1^r and \mathbf{x}_2^r belong to the same epipolar line \mathbf{l}_{e_r} . \mathbf{C}_l and \mathbf{C}_r are the camera centres, while \mathbf{e}_l and \mathbf{e}_r are the epipoles (i.e. the projections of one camera centres onto the other image). (b) A graphical representation of the rectification transform that maps the epipolar lines in a parallel configuration, sending the epipoles to infinity.

\mathbf{x}_2^r is called *motion parallax*: The closer the 3D points are to the camera, greater will be the motion of its projections on the left and right frame. The line connecting \mathbf{x}_1^r and \mathbf{x}_2^r (that is the projection of the line between \mathbf{X}_1 and \mathbf{X}_2) is called *epipolar line* and each 3D point projecting in \mathbf{x}_1^l must fall on the right frame on the same epipolar line. So, for each point in the left image \mathbf{x}_l , there exists a line in the right image that passes through its corresponding point \mathbf{x}_r . This constraint is encoded in the *fundamental matrix* F , such that $\mathbf{x}_r^T F \mathbf{x}_l = 0$.

Given a stereo pair, a matching problem has to be solved in order to obtain an estimate of the 3D structure of the scene. In particular, a given point in the left image has to match one-to-one with another point in the right image. To find these correspondences, it is possible to exploit the epipolar constraint to restrict the search space to a line. Better still, in order to ease the search problem, stereo frames are typically reported to a more convenient standard form using a *stereo rectification* technique (Fig. 3b). The aim of rectification is to transform the original frames in a way that all epipolar lines are overlapping, so that all matching points have the same row in both the left and right images. A rectified stereo pair can then be used to compute a *disparity map*, where each pixel value represents the displacement between the left and right views. If, in addition to epipolar geometry, the internal *calibration parameters* of the cameras are known, the disparity map can eventually be converted easily into a *depth map*, i.e., into a 3D representation of the scene.

4. The StereoDisparity App

Our StereoDisparity App, developed in Matlab, includes: (1) a point selection tool to identify correspondences between the left and right photos; (2) an automatic rectification algorithm to correct the original misalignment of epipolar lines; (3) enhancement algorithms to correct illumination and reduce noise; (4) a dense stereo correspondence software, based on block matching, with manual tuning of parameters; and (5) a densification procedure based on scene plane homographies. Details are provided in the following subsections.

As shown in the block diagram of Fig. 4, from a functional viewpoint StereoDisparity is composed by three main blocks: (i) a *MainGUI* that is used to call the other software modules as well as to compute the disparity map using automatic algorithms; (ii) the *RectificationTool*, used to rectify the stereo pair; and the *PlaneSelectionTool* for selecting corresponding planar

regions in the left and right images and to compute missing disparities according to planar homographies. At startup, the stereo pair is loaded in MainGUI. Then, RectificationTool is called to correct image misalignments.

4.1. Rectification

Even if the input stereographs seem to be already rectified (see again Fig. 2), they actually have some residual misalignments. To refine the input data, RectificationTool lets the user select point matches on the stereo pair with a few mouse clicks on a simple graphical interface. Once enough correspondences have been selected, the rectification algorithm described in [11] is run. To ease the matching selection, we also included two enhancement algorithms for improving the image quality: A color correction method that tries to mitigate the illumination differences between the left and right frame [12], and a de-noising method [13].

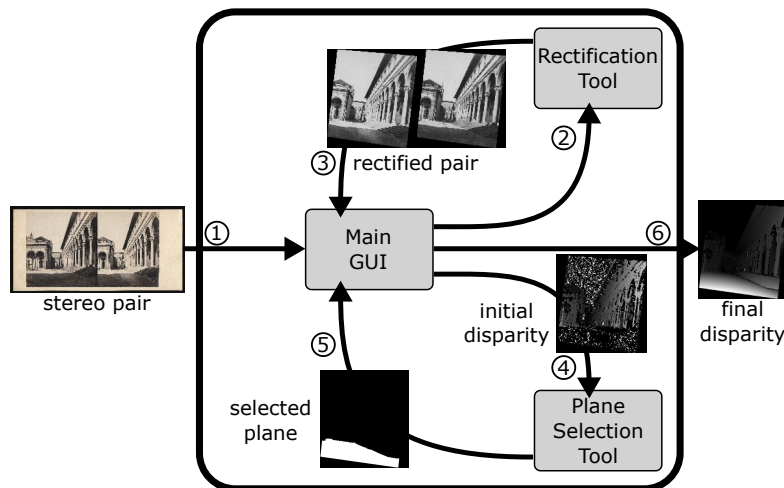


Figure 4: Block diagram of the StereoDisparity App. From the MainGUI, the Rectification Tool and the PlaneSelection Tool are called. Intermediate outputs are also shown.

4.2. Disparity map computation

Given the rectified pair, initial estimates of the disparities can be obtained with automatic stereo matching methods. We use three simple block matching methods: Two from the OpenCV library (the Block Matching and Semi Global Block Matching), and the Matlab implementation. For each of these solutions, StereoDisparity lets the user select appropriate parameters and observe the updated results (see Fig. 5a). However, the obtained disparity map, even after manual tuning, usually contains many missing areas and errors (see Fig. 5b) that can be fixed through PlaneSelectionTool, thus improving the result.

4.3. Disparity map improvement

PlaneSelectionTool is an interface—similar to Rectification Tool—for selecting matching points. In this case, the user has to pay attention on selecting only points belonging to corresponding planar surfaces. Once selected enough points, a *planar homography* transformation is estimated by the software. A planar homography is a perspective transformation. In particular, we are interested in homographies induced by a plane in the scene. Let \mathbf{X} be a 3D point belonging to the plane π , and $\mathbf{x}_1, \mathbf{x}_2$ the projections of \mathbf{X} on two different images. Then the homography H_π , induced by π , is a map such that $H_\pi \mathbf{x}_1 = \mathbf{x}_2$ (or $H_\pi^{-1} \mathbf{x}_2 = \mathbf{x}_1$). H_π has 8 degrees of freedom,

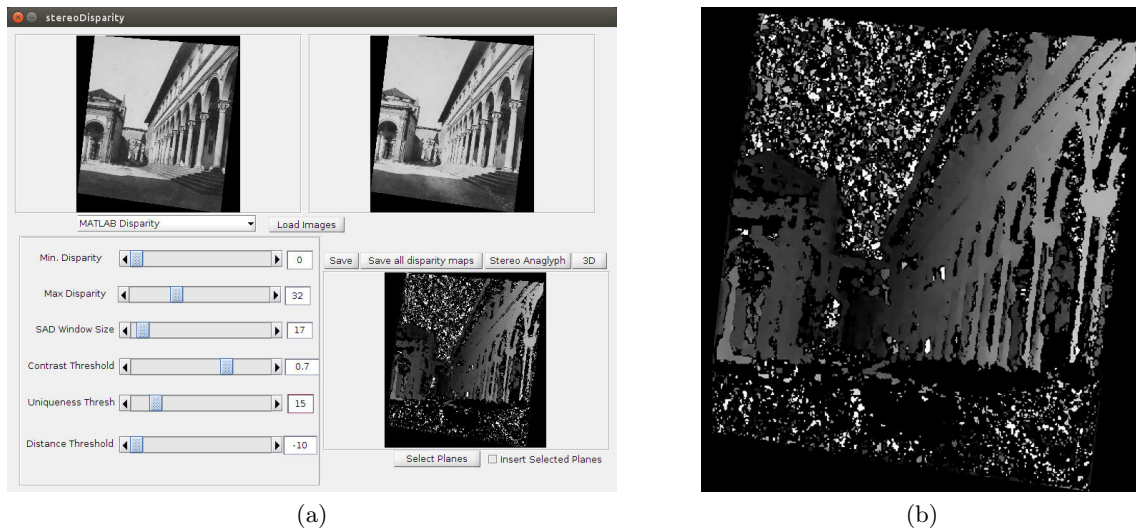


Figure 5: (a) The MainGUI of the StereoDisparity App. The user can choose the stereo algorithm to use and tune its parameters. (b) Disparity map obtained with the automatic block matching algorithm of Matlab.

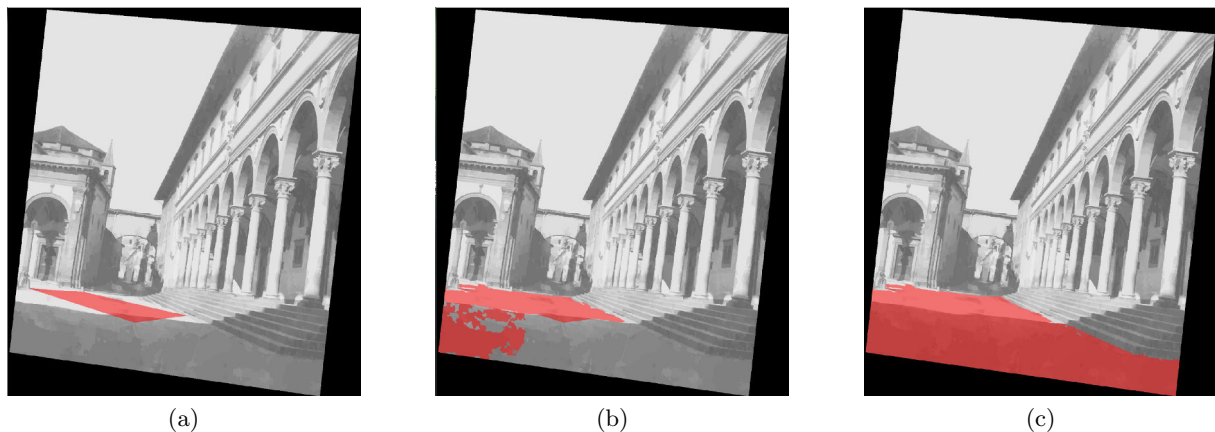


Figure 6: Interactive expansion scheme for planar surfaces: in (a) the initial planar surface detected. In (b) and (c) the region is progressively expanded by including super-pixels of areas belonging to the same plane (in this case the ground plane).

and each point correspondence produces two independent constraints, so four matching points are sufficient to estimate H_π . More details can be found in [9].

Manually selecting a whole plane region can be tedious and can take a lot of time. We simplify this step by implementing a planar expansion scheme exploiting super-pixels. After obtaining the initial plane region and the respective homography mapping (by selecting four or more correspondences), we segment the left frame into super-pixels, using EDISON that implements mean shift segmentation [14, 15, 16]. The initial planar region is superimposed on the image (Fig 6a), and the user can manually select single super-pixels to be included into the plane (Fig. 6b and 6c). In this way, the estimated homography can be applied to all the points of the selected regions, thus obtaining a dense disparity estimation. By repeating these operations, multiple planar regions can be selected to augment and improve the disparity map

up to the desired level of detail.

Finally, all disparity values obtained by PlaneSelectionTool are fused with those of the initial disparity map, thus obtaining a richer description of the scene geometry (see Fig. 7).

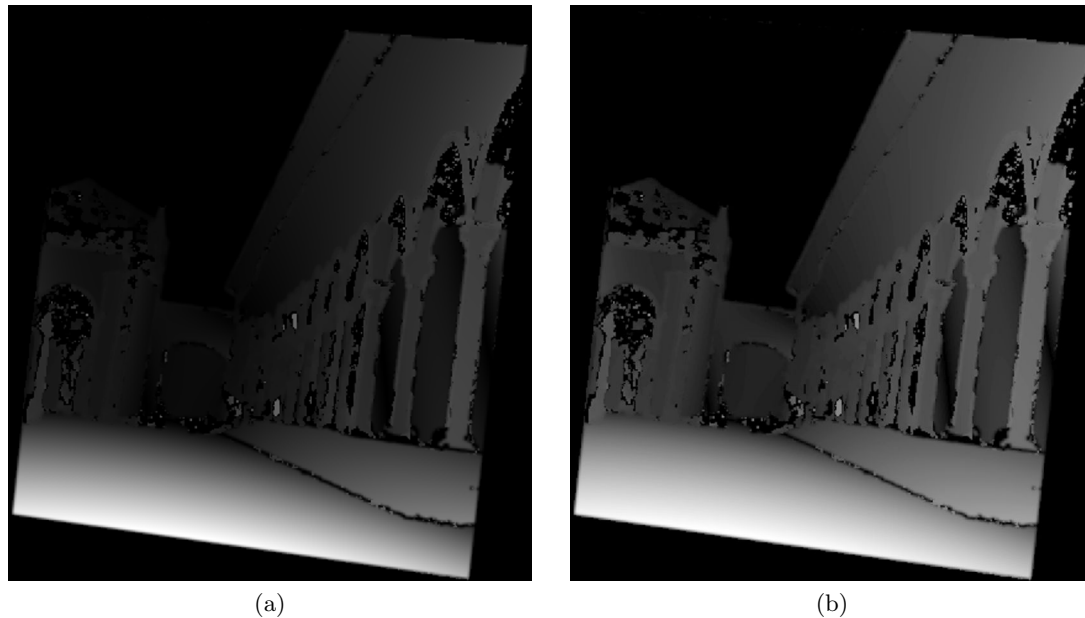


Figure 7: (a): Disparity map obtained with the proposed method for the stereo pair of Fig. 1a. (b): The same after contrast enhancement, that emphasizes depth details. (Better viewed on a computer screen.)

This final dense disparity map can be used to achieve a 3D reconstruction of the scene. At the current state of the application, we limited our 3D output to a projective reconstruction, since calibration data (e.g. focal lengths) of the historical stereo pairs have not been estimated. Nevertheless, this 3D reconstruction it is already sufficient to appreciate the different depths of the scene, even if perspectively distorted (see Fig. 8).

5. Conclusions and Future Work

In this paper we presented StereoDisparity App, a computer vision-based graphical user interface to compute disparity maps and projective 3D reconstructions from stereo photographs, through limited manual user interaction. StereoDisparity can be used to work on historical images, that are practically intractable with automatic state-of-the-art stereo methods.

Future work will focus at first on obtaining a full metric 3D reconstruction. Camera intrinsics are missing for the historical photos: We will include in our StereoDisparity App a calibration tool exploiting self-calibration techniques, so that a metric 3D model can be finally obtained. Using such a reconstruction we will be able to register the old photos with the modern scene. This can be used, for instance, to implement augmented reality applications that superimpose the historical appearance to the contemporary scene.

Acknowledgements

This work has been carried out during the TRAVIS (Tecniche di Realtà Aumentata per la Visualizzazione di Immagini Storiche) project supported by Fondazione Ente Cassa di Risparmio di Firenze, within the call "Giovani Ricercatori Protagonisti" (University of Florence,

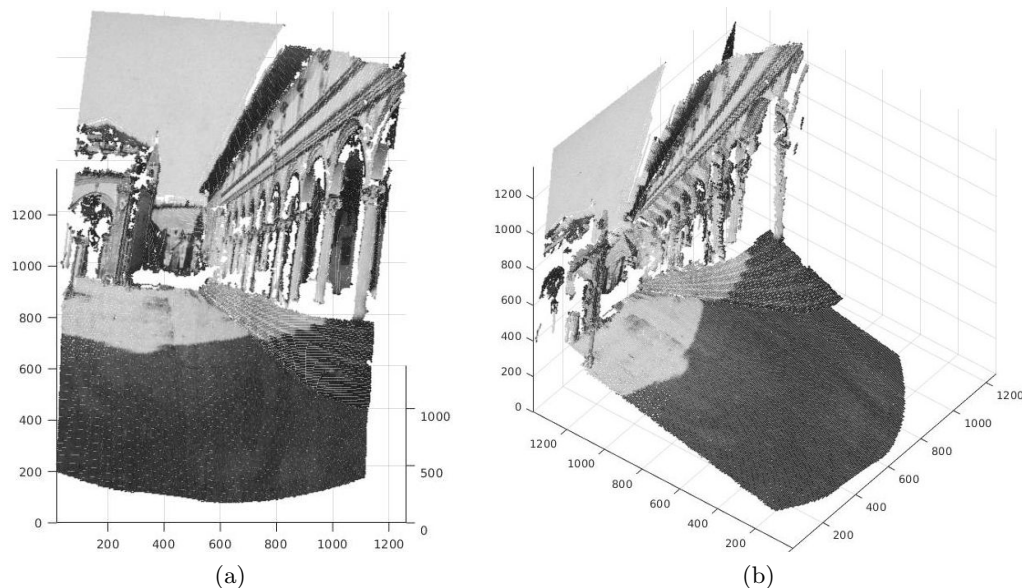


Figure 8: A 3D perspective reconstruction obtained from the enhanced disparity map.

D.R. N. 127804 (1206), 01/10/2015). We would like to thank Mrs Anna Pelleri Hautmann for allowing the reproduction of the photos in this paper. We also thank Drs. Costanza Caraffa and Ute Dercks at Photothek des Kunsthistorischen Instituts in Florenz—Max-Planck-Institut, for their kind support. Digital scans: ©Stefano Fancelli/KHI.

References

- [1] Tanai T, Matsushita Y, Sato Y and Naemura T 2017 *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (accepted) URL <https://github.com/t-tanai/LocalExpStereo>
- [2] Dzwierzynska J 2017 *IOP Conference Series: Materials Science and Engineering* **245** 062015
- [3] Bitelli G, Dellapasqua M, Girelli V A, Sbaraglia S and Tinia M A 2017 *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLII-5/W1* 113–119 URL <https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-5-W1/113/2017/>
- [4] Bräuer-Burchardt C and Voss K 2001 *Virtual and Augmented Architecture (VAA'01)* (Springer London) pp 159–170
- [5] Maiwald F, Vietze T, Schneider D, Henze F, Münster S and Niebling F 2017 *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 447–452
- [6] Schindler G and Dellaert F 2012 **7**
- [7] Scharstein D and Szeliski R 2002 *International Journal of Computer Vision* **47** 7–42 URL <http://vision.middlebury.edu/stereo/>
- [8] Fanelli G 1999 *Firenze in Stereoscopia* (Octavo Editore)
- [9] Hartley R I and Zisserman A 2004 *Multiple View Geometry in Computer Vision* 2nd ed (Cambridge University Press, ISBN: 0521540518)
- [10] Szeliski R 2010 *Computer Vision: Algorithms and Applications* 1st ed (New York, NY, USA: Springer-Verlag New York, Inc.) ISBN 1848829345, 9781848829343
- [11] Fusiello A and Irsara L 2008 *19th International Conference on Pattern Recognition* pp 1–4 ISSN 1051-4651
- [12] Reinhard E, Adhikhmin M, Gooch B and Shirley P 2001 *IEEE Computer Graphics and Applications* **21** 34–41
- [13] Katkovnik V, Foi A, Egiazarian K and Astola J 2010 *Int. J. Comput. Vision* **86** 1–32
- [14] Meer P and Georgescu B 2001 *IEEE Transactions on Pattern Analysis and Machine Intelligence* **23** 1351–1365
- [15] Christoudias C M, Georgescu B and Meer P 2002 *Object recognition supported by user interaction for service robots* vol 4 pp 150–155
- [16] Comaniciu D and Meer P 2002 *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24** 603–619