Reconstruction of the 3D Scenes from the Matching Between Image Pair Taken by an Uncalibrated Camera

Karima Karim[1(✉)], Nabil El Akkad[1,2(✉)], and Khalid Satori[1(✉)]

[1] LIIAN, Department of Computer Science, Faculty of Science, Dhar El Mahraz, Sidi Mohamed Ben Abdellah University, B.P 1796 Atlas, Fez, Morocco
karima.karim35@gmail.com, nabil.elakkad@usmba.ac.ma, khalid.satori@gmail.com

[2] Department of Mathematics and Computer Science, National School of Applied Sciences (ENSA) of Al-Hoceima, University of Mohamed First, B.P 03 Ajdir, Oujda, Morocco

Abstract. In this paper, we will study a new approach of reconstruction of three-dimensional scenes from an auto calibration method of camera characterized by variable parameters. Indeed, obtaining the 3D scene is based on the Euclidean reconstruction of the interest points detected and matched between pair of images. The relationship between the matches and camera parameters is used to formulate a nonlinear equation system. This system is transformed into a nonlinear cost function, which will be minimized to determine the intrinsic and extrinsic camera parameters and subsequently estimate the projection matrices. Finally, the coordinates of the 3D points of the scene are obtained by solving a linear equation system. The results of the experiments show the strengths of this contribution in terms of precision and convergence.

Keywords: Auto calibration · Reconstruction · Variable parameter
Fundamental matrix

1 Introduction

In this work, we will investigate about the three-dimensional reconstruction being a technique that allows obtaining a 3D representation of an object from a sequence of images of this object taken by different views. In fact, several 3D reconstruction techniques use calibration or Auto-calibration methods.

During this work, we will presented a new approach to reconstructing three-dimensional scenes from a method of autocalibration of cameras characterized by variable parameters. In general, the determination of the 3D scene is based on the euclidean reconstruction of the interest points detected and matched by the ORB descriptor [20]. The intrinsic parameters of the cameras are estimated by the resolution of a nonlinear equation system (using the nonlinear equations of the Levenberg-Marquart algorithm [18]), and they are used with the fundamental matrices (estimated from 8 pairings between the image couples by the RANSAC algorithm [11]) to determine the extrinsic camera parameters, and finally to estimate the projection matrix

Y. Tabii et al. (Eds.): BDCA 2018, CCIS 872, pp. 447–463, 2018.
https://doi.org/10.1007/978-3-319-96292-4_35

(expressed according to the intrinsic and extrinsic parameters of the cameras used). The relationships between camera parameters, projection matrix elements, pairing coordinates, and 3D point coordinates gives a linear equation system and the resolution of this system permits to obtain a cloud of 3D points.

In this introduction, we have therefore provided the general ideas that will be investigated in this paper. The rest of this work is organized as follows:

A diagram of different steps of our method is presented in the second part, the scene and the camera model are presented in the third part, the fourth part treats the auto calibration of the cameras, the fifth part explains the reconstruction of the 3D scene, the experiments will be discussed in the sixth paragraph and the conclusion is presented in the last part.

2 Diagram of Different Steps of Our Method

The Fig. 1. below represents a diagram of different steps of the reconstruction of 3D scene:

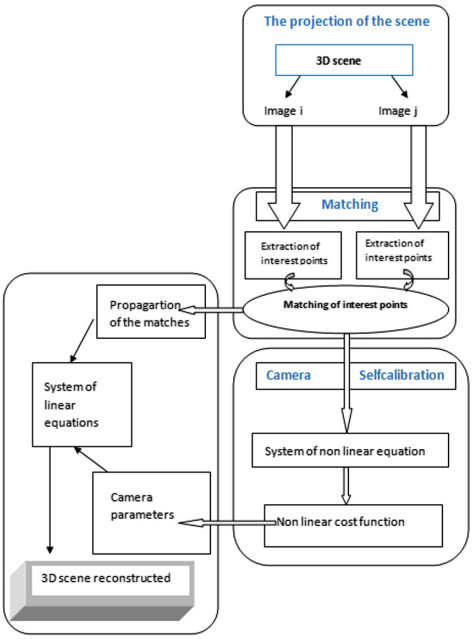


Fig. 1. Diagram of the reconstruction of the 3D scene

3 Scene and Camera Model

3.1 Presentation of the Scene

We consider two points S_1 and S_2 of the 3D scene, there is a single point S_3, such as $S_1S_2S_3$: is an equilateral triangle. $R_e(OX_eY_eZ_e)$ is the euclidean reference associated to the triangle wich O is its center and b its side.

3.2 Model of the Camera

We are using the pinhole model of the camera Fig. 2. so that we project the points of the 3D scene in the planes of images, this model is characterized by a matrix $K_i(R_it_i)$ of size (3×4), with:

R_i : the rotation matrix
t_i : the translation vector
K_i : The matrix of intrinsic parameters defined by:

$$K_i = \begin{pmatrix} f_i & s_i & u_{\circ i} \\ 0 & \varepsilon_i f_i & v_{\circ i} \\ 0 & 0 & 1 \end{pmatrix} \tag{1}$$

with f_i : focal length
ε_i : the scaling factor
s_i : the skew factor
(u_{0i}, v_{0i}) : the coordinates of the principal point.

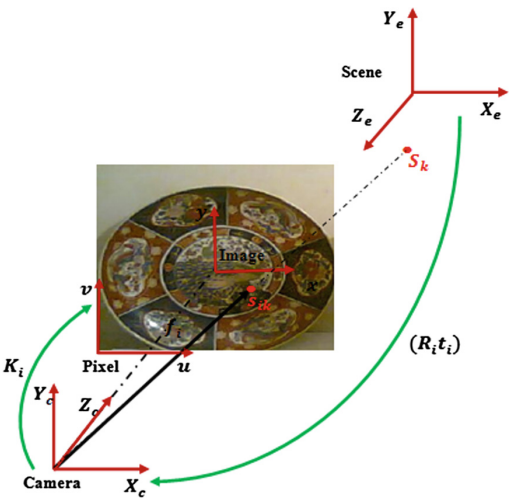


Fig. 2. Representation of the scene

4 Camera Autocalibration

The auto Calibration [1–10] is a technique that allows us to estimate the parameters of the cameras without any prior knowledge on the scene.

4.1 ORB Descriptor: Oriented FAST and Rotated BRIEF

The detection [12–14] and the matching [15–17] of the interest points are important steps in the autocalibration and the reconstruction of 3D scenes, in this paper we based on the ORB descriptor: Oriented FAST and rotated BRIEF [21] (ORB: Binary Robust Independent Elementary Features) which is a fast robust local feature detector, first presented by Rublee et al. in 2011 [20], that can be used in computer vision tasks like object recognition or 3D reconstruction. It is a fusion of the FAST key point detector and BRIEF descriptor with some modifications [9]. Initially to determine the key points, it uses FAST. Then a Harris corner measure is applied to find top N points. FAST does not compute the orientation and is rotation variant. It computes the intensity weighted centroid of the patch with located corner at center. The direction of the vector from this corner point to centroid gives the orientation. Moments are computed to improve the rotation invariance. The descriptor BRIEF poorly performs if there is an in-plane rotation. In ORB, a rotation matrix is computed using the orientation of patch and then the BRIEF descriptors are steered according to the orientation.

The ORB descriptor is a bit similar to BRIEF. It doesn't have an elaborate sampling pattern as BRISK [26] or FREAK [27]. However, there are two main differences between ORB and BRIEF:

1. ORB uses an orientation compensation mechanism, making it rotation invariant.
2. ORB learns the optimal sampling pairs, whereas BRIEF uses randomly chosen sampling pairs.

ORB uses a simple measure of corner orientation – the intensity centroid [28]. First, the moments of a patch are defined as:

$$\forall p,\, q \in \{0,1\} : m_{pq} = \sum_{x,y} x^p y^q\, I(x,y) \tag{2}$$

With:
$p, q \in \{0,1\}$ Binary selector for x and y direction

x,y Circular window
$x^p y^q$ weighted by coordinate
$I(x,y)$ image function

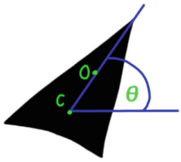
Image moments help us to calculate some features like center of mass of the object, area of the object etc.

With these moments we can find the centroid, the "center of mass" of the patch as:

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}}\right) \tag{3}$$

and by constructing a vector from the patch center O to the centroid C, we can define the relative orientation of the patch as:

$$\underset{OC}{\rightarrow} \theta = \text{atan2}(m_{01}, m_{10}) \tag{4}$$



ORB discretize the angle to increments of $\frac{2\pi}{30}$ (12°), and construct a lookup table of precomputed BRIEF patterns. As long as the keypoint orientation θ is consistent across views, the correct set of points will be used to compute its descriptor.

To conclude, ORB is binary descriptor that is similar to BRIEF, with the added advantages of rotation invariance and learned sampling pairs. You're probably asking yourself, how does ORB perform in comparison to BRIEF. Well, in non-geometric transformation (those that are image capture dependent and do not rely on the view-point, such as blur, JPEG compression, exposure and illumination) BRIEF actually outperforms ORB. In affine transformation, BRIEF perform poorly under large rotation or scale change as it's not designed to handle such changes. In perspective transformations, which are the result of view-point change, BRIEF surprisingly slightly outperforms ORB.

4.2 The Projection Matrix

We consider S_1 and S_2 two points of the 3D scene and π the plan which contains these two points.

$R_e(O\ X_eY_eZ_e)$ is the Euclidian reference which is associated to the triangle of the center O and side b

The coordinates of points S_1, S_2 and S_3 Fig. 3 are given as below:

$$S_1 = \left(\frac{b}{2}, \frac{\sqrt{3}}{2}b, 1\right)^T$$

$$S_2 = (b, 0, 1)^T$$

$$S_3 = (0, 1, 1)^T$$

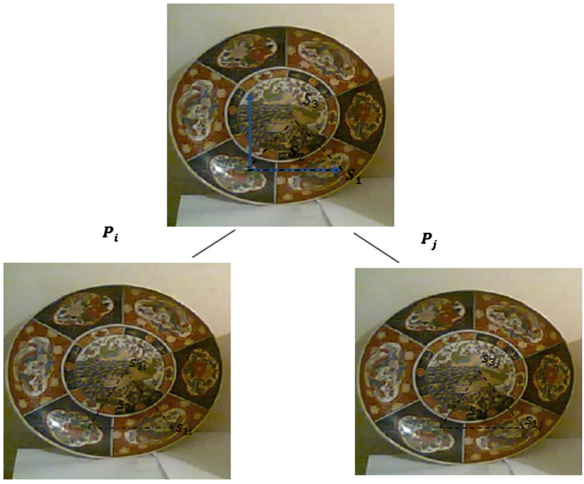


Fig. 3. Representation of points S_1, S_2 and S_3 in the two images i and j.

We consider the two homography H_i and H_j that can be used to project the plan in the images i and j, so the projection of the two points can be represented by the following expressions:

$$s_{im} \sim H_i S_m \tag{5}$$

$$s_{jm} \sim H_j S_m \tag{6}$$

With $m = 1, 2$. s_{im} and s_{jm} represent respectively the points in the images i and j which are the projections of the two summits S_1 and S_2 of the 3D scene, and H_n represents the homography matrix defined by:

$$H_n = K_n R_n \begin{pmatrix} 1 & 0 & \\ 0 & 1 & R_n^T t_n \\ 0 & 0 & \end{pmatrix}; n = i, j \tag{7}$$

With:
R_n : the rotation matrix
t_n : the translation vector
K_n : The matrix of intrinsic parameters.

The expressions (5) and (6) can be written as :

$$s_{im} \sim H_i B S'_m \tag{8}$$

$$s_{jm} \sim H_j B S'_m \tag{9}$$

$$\text{With} : B = \begin{pmatrix} b & \frac{b}{2} & 0 \\ 0 & \frac{\sqrt{3}}{2} b & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$S'_m = \begin{pmatrix} a \\ b \\ 1 \end{pmatrix}$$

For:

$$\begin{cases} m = 1 <= > a = 0 \; and \; b = 1 \\ m = 2 <= > a = 1 \; and \; b = 0 \end{cases}$$

We put:

$$P_n \sim H_n B \; ; n = i, j \tag{10}$$

With P_i and P_j are the projections matrix of the two points $S_1^{'}$ and $S_2^{'}$ in the images i and j Figs. 3 and 6.

From the Eq. (10) we have:

$$P_j \sim H_{ij} P_i \tag{11}$$

With:

$$H_{ij} \sim H_j H_i^{-1} \tag{12}$$

H_{ij} is the homography between the images i and j.

The Eqs. (8), (9) and (10) give:

$$s_{im} \sim P_i S'_m \tag{13}$$

$$s_{jm} \sim P_j S'_m \tag{14}$$

And from the Eqs. (11) and (14) we have :

$$s_{jm} \sim H_{ij} P_i S'_m \tag{15}$$

The Eq. (15) gives:

$$e'_j s_{jm} \sim e'_j H_{ij} P_i S'_m \tag{16}$$

This later gives:

$$e'_j s_{jm} \sim F_{ij} P_i S'_m \tag{17}$$

With F_{ij} is the fundamental matrix between the images i and j.

$$e'_j = \begin{pmatrix} 0 & -e_{j3} & e_{j2} \\ e_{j3} & 0 & -e_{j1} \\ -e_{j2} & e_{j1} & 0 \end{pmatrix}$$

$(e_{j1} e_{j2} e_{j3})^T$ are the coordinates of the epipole of the right image, this epipole can be estimated by the fundamental matrix.

The expression (18) gives:

$$s_{i1} \sim P_i S'_1 \tag{18}$$

$$s_{i2} \sim P_i S'_2 \tag{19}$$

So from the two last relationships, we gets four equations with eight unknowns that are the elements of P_i

The expression (17) gives:

$$e'_j s_{j1} \sim F_{ij} P_i S'_1 \tag{20}$$

$$e'_j s_{j2} \sim F_{ij} P_i S'_2 \tag{21}$$

From the two last relationships, we get four other equations with eight unknowns which are the parameters of P_i. So we can estimate the parameters of P_i, because we have a total of eight unknown equations that are the elements of P_i.

The Eq. (11) gives:

$$e_j^{'} P_j \sim e_j^{'} H_{ij} P_i \tag{22}$$

That gives:

$$e_j^{'} P_j \sim F_{ij} P_i \tag{23}$$

The previous expression gives eight unknown equations that are the elements of P_j.

So we can estimate the parameters of P_j from these eight equations with eight unknown.

4.3 Autocalibration Equations

In this part, we will determine the relationship between the images of the absolute conic (ω_i and ω_j), and a relationship between the two points (S_1, S_2) of the 3D scene and their projections (s_{i1}, s_{i2}) and (s_{j1}, s_{j2}) in the planes of the left and right images respectively, the different relationships are established from some techniques of projective geometry. A nonlinear cost function will be defined from the determination of these relationships. The formulated cost function will be minimized by the Levenberg-Marquardt algorithm [18] to estimate ω_i and ω_j and finally the intrinsic parameters of the cameras used [24].

The Eq. (11) gives:

$$\lambda_{im} s_{im} = P_i S_m^{'} \tag{24}$$

With: $P_i = \begin{pmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{pmatrix}$

$$s_{im} = \begin{pmatrix} x_{im} \\ y_{im} \\ 1 \end{pmatrix}$$

$$P_i^T \omega_i P_i \sim \begin{pmatrix} B'^T B' & B'^T R_i^T t_i \\ t_i^T R_i B' & t_i^T t_i \end{pmatrix} \tag{25}$$

With:

$$B' = \begin{pmatrix} b & \frac{b}{2} \\ 0 & \frac{\sqrt{3}}{2} b \\ 0 & 0 \end{pmatrix} \tag{26}$$

K_i is an upper-triangular matrix normalized as $\det K_i = 1$

$\omega_i = \left(K_i K_i^T\right)^{-1}$ is the image of the absolute conic.

The same for P_j:

$$P_j^T \omega_j P_j \sim \begin{pmatrix} B'^T B' & B'^T R_j^T t_j \\ t_j^T R_j B' & t_j^T t_j \end{pmatrix} \tag{27}$$

We can deduce that the first rows and columns of the matrix $P_i^T \omega_i P_i$ and $P_j^T \omega_j P_j$ are the same.

We put X_i and X_j the two matrix corresponding respectively to the first two rows and columns of the two previous matrices.

$X_m = \begin{pmatrix} x_{1m} & x_{3m} \\ x_{3m} & x_{2m} \end{pmatrix}$, with $m = i, j$.

So we conclude the 3 following equations:

$$\begin{cases} x_{1i} = x_{2i} \\ x_{1j} = x_{2j} \\ x_{1i} x_{3j} = x_{1j} x_{3i} \end{cases} \tag{28}$$

Each image pair gives a system of 3 equations with 8 unknown (4 unknown for ω_i and 4 unknown for ω_j), so to solve the equation system (28), we need at least 4 images.

The equation system (28) is nonlinear, so to solve this system of equations we minimize the following nonlinear cost function:

$$\min_{\omega_k} \sum_{j=i+1}^{n} \sum_{i=1}^{n-1} \left(\alpha_{ij}^2 + \beta_{ij}^2 + \gamma_{ij}^2\right) \tag{29}$$

With: $\propto_{ij} = q_{1i} - q_{2i}; \beta_{ij} = q_{1j} - q_{2j}; \gamma_{ij} = q_{1i} q_{3j} - q_{1j} q_{3i}$, and : n is the number of images.

The Eq. (29) will be minimized by the Levenberg–Marquardt algorithm [18], this algorithm requires an initialization step. So the camera parameters are initialized as follows:

Pixels are squares, so: $\varepsilon_i = \varepsilon_j = 1$, $s_i = s_j = 0$,

The principal point is in the centre of the image so: $x_{0i} = y_{0i} = x_{0j} = y_{0j} = 256$ (because the images used are of sizes 512×512), and the focal distances f_i and f_j are obtained by the resolution of the equation system (29).

4.4 General Algorithm

1. Detecting and matching of interest points respectively by ORB algorithm.
2. Determination of the Fundamental matrix by Ransac algorithm using eight matches.
3. Calculation of the projection matrices used by the projection of two points.
4. Formulation of the non-linear cost function
5. Minimization of non-linear cost function by the Levenberg-Marquardt algorithm.
   - 5.1. Initialization: we suppose that the principal point is in the center of the image, the pixels are squared, and we calculate the focal length.
   - 5.2. Optimization of the non-linear cost function.

5 Reconstruction of the 3D Scene

This part is dedicated to the 3D reconstruction to determine a cloud of 3D points from the matching between the pairs of images [19, 22, 23, 25]. In theory, getting the position of 3D points from their projections in the images is trivial. The matching 2D point pair must be the projections of the 3D points in the images.

This reconstruction is possible when the geometric relationship between the cameras is known and when the projection of the same point is measured in the images.

The reconstruction of a few points of the 3D scene requires the estimation of the projection matrix of this scene in different images.

We have: P_0 and P_1 two projection matrices of the 3D scene, respectively in the plane of the images, such as:

$$s_{0m} \sim P_0 S_m \tag{30}$$

$$s_{im} \sim P_i S_m$$

We have $P \sim K(R\ t)$
So,

$$P_0 \sim K_0(I_3 O) \tag{31}$$

$$P_1 \sim K_1(R_1 t_1)$$

The essential matrix [29] is the specialization of the fundamental matrix to the case of normalized image coordinates. Historically, the essential matrix was introduced (by

Longuet-Higgins) before the fundamental matrix, and the fundamental matrix may be thought of as the generalization of the essential matrix in which the (inessential) assumption of calibrated cameras is removed. The essential matrix has fewer degrees of freedom, and additional properties, compared to the fundamental matrix.

The defining equation for the essential matrix is:

$$\widehat{X}_1^T E \widehat{X}_0 = 0$$

With $\widehat{X} = K^{-1}X$.

In terms of the normalized image coordinates for corresponding points $X_0 \leftrightarrow X_1$

Substituting for $\widehat{X}_0$ and $\widehat{X}_1$ gives $X_1^T K_1^T E K^{-1} X_0 = 0$. Comparing this with the relation $X_1^T F_{12} X_0 = 0$ for the fundamental matrix, it follows that the relationship between the fundamental and essential matrices is

$$\mathrm{E}_{12} = \mathrm{K}_1^{\mathrm{T}} \mathrm{F}_{12} \mathrm{K}_0 \tag{32}$$

With: F_{12} represent the fundamental matrix between the first and second images, It is estimated from 8 matches between this couple of images.

E_{12} is decompose into singular value in the following equation:

$$\mathrm{E}_{12} = \lambda \mathrm{L}_1 \mathrm{U}(1 \quad 1 \quad 0)\mathrm{L}_2^{\mathrm{T}} \tag{33}$$

With

λ is a non-zero scalar,

And $\mathrm{U}(1 \quad 1 \quad 0)$ is written in the following form:

$$\mathrm{U}(1 \quad 1 \quad 0) = \mathrm{N}_1 \mathrm{N}_2^{\mathrm{T}} = -\mathrm{N}_1 \mathrm{N}_2^{\mathrm{T}} \tag{34}$$

$$\mathrm{N}_1 = \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \mathrm{N}_2 = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{35}$$

From (33) and (34), we have:

$$\mathrm{E}_{12} = \lambda \mathrm{L}_1 \mathrm{N}_1 \mathrm{N}_2^{\mathrm{T}} \mathrm{L}_2^{\mathrm{T}} = -\lambda \mathrm{L}_1 \mathrm{N}_1 \mathrm{N}_2^{\mathrm{T}} \mathrm{L}_2^{\mathrm{T}} \tag{36}$$

L_1 is orthonormal, so the matrix E_{12} can be written as the following form:

$$\mathrm{E}_{12} \sim \mathrm{L}_1 \mathrm{N}_1 \mathrm{L}_1^{\mathrm{T}} \left(\mp \mathrm{L}_1 \mathrm{N}_2 \mathrm{L}_2^{\mathrm{T}}\right) \sim -\mathrm{L}_1 \mathrm{N}_1 \mathrm{L}_1^{\mathrm{T}} \left(\mp \mathrm{L}_1 \mathrm{N}_2^{\mathrm{T}} \mathrm{L}_2\right) \tag{37}$$

On the other hand, E_{12} is expressed as follows:

$$\mathrm{E}_{12} \sim [\mathrm{t}_1]_{\wedge} \mathrm{R}_1 \tag{38}$$

$$[t_1]_\wedge = \begin{bmatrix} 0 & -t_{13} & t_{12} \\ t_{13} & 0 & -t_{11} \\ -t_{12} & t_{11} & 0 \end{bmatrix} \tag{39}$$

And $(t_{11}t_{12}t_{13})^T$ are the coordinates of the translation vector t_1.

From the two latest expressions, we can conclude the vector t_1 that admits an unique solution:

$$[t_1]_\wedge \sim L_1 N_1 L_1^T \tag{40}$$

And the rotation matrix R_1 admits 4 solutions

$$R_1 \sim \mp L_1 N_2 L_2^T \text{ or } R_1 \sim \mp L_1 N_2^T L_2^T \tag{41}$$

But the determinant of the rotation matrix must be equal to 1, which allows fixing a sign for the two matrices:

$$\mp L_1 N_2 L_2^T \text{ and } \mp L_1 N_2^T L_2^T$$

So the number of solutions for R_1 becomes 2.

We use the two solutions to reconstruct the 3D scene, and finally we choose the solution that gives the best Euclidean reconstruction.

From the Eq. (30), we obtain the following linear system of equations:

$$M(X\ Y\ Z)^T = N \tag{42}$$

M : Matrix of size 4 x 3
N : Vector of size 4

These two matrices are expressed in function of the elements of the projection matrices and the coordinates of the matches.

$(X\ Y\ Z)^T$: The vector of the coordinates of the searched 3D point.

The coordinates of the 3D points (the solution of the Eq. (42)) are obtained by the following expression:

$$\det M^T M \neq 0 \text{ so } M^T M \text{ is no singular}$$

$$(X\ Y\ Z)^T = \left(M^T M\right)^{-1} M^T N \tag{43}$$

6 Experimentations

In this part, we have taken two images of an unknown three-dimensional scene by a CCD camera characterized by variable intrinsic parameters Fig. 4. In the first step, we applied the ORB descriptor to determine the interest points Fig. 5. And the matching between the two selected images Fig. 6. Subsequently and after implementation the algorithms of Ransac and Levenberg-Marquardt while relying on the Python programming language, we got the result of the 3D reconstruction below Fig. 7:

Fig. 4. Two images of unknown 3D scene

Fig. 5. The interest points in the two images (blue color) (Color figure online)

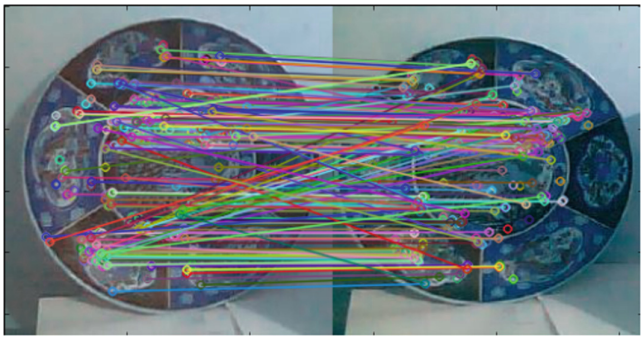


Fig. 6. The matches between the two images

Fig. 7. The reconstructed 3D scene

The detection of interest points, Fig. 5. And the mapping Fig. 6 are carried out by the descriptor ORB [20]. The determination of the relationship between the matches and the camera parameters permit to formulate a system of non-linear equations. This system is introduced in a non-linear cost function. The minimization of this function by Levenberg-Marquardt algorithm [18] allows finding an optimal solution of the camera parameters. These parameters are used with the matches to obtain an initial point cloud Fig. 7.

We have a lot of values to estimate, every parameters have a minimum value.

The intrinsic camera parameters (focal lengths, coordinates of the principal points, scale factors, skew factors) and the rotation matrices.

This population is chosen in a way that each parameter belongs to a specific interval Table 1.

Table 1. Intervals of camera parameters

Parameters	Intervals
f_τ	[800 2000]
ϵ_τ	[0 1]
s_τ	[0 1]

The usefulness of our contribution is to obtain a 3D scene reconstructed just from 2 images taken from an uncalibrated camera and with variable intrinsic parameters. The next steps will be the 3D modeling in order to finalize our work and find a robust results and a very well a 3D scene reconstructed based on a triangulation construction and a texture mapping.

7 Conclusion

In this work we have treated a new approach of the reconstruction of three-dimensional scenes from a method of autocalibration of cameras characterized by variable intrinsic parameters. The interest points are detected and matched by the ORB descriptor, and it's used later with the projection matrix (expressed according to camera settings) of the scene in the planar images to determine coordinate of the point cloud, so that we can reconstruct the scene.

References

1. Lourakis, M.I.A., Deriche, R.: Camera self-calibration using the kruppa equations and the SVD of the fundamental matrix: the case of varying intrinsic parameters. Technical report 3911, INRIA (2000)
2. Sturm, P.: Critical motion sequences for the self-calibration of cameras and stereo systems with variable focal length. Image Vis. Comput. **20**(5–6), 415–426 (2002)
3. Malis, E., Capolla, R.: Camera self-calibration from unknown planar structures enforcing the multi-view constraints between collineations. IEEE Trans. Pattern Anal. Mach. Intell. **4**(9) (2002)
4. Gurdjos, P., Sturm, P.: Methods and geometry for plane-based self-calibration. In: CVPR, pp. 491–496 (2003)
5. Liu, P., Shi, J., Zhou, J., Jiang, L.: Camera self-calibration using the geometric structure in real scenes. In: Proceedings of the Computer Graphics International (2003)
6. Hemayed, E.E.: A survey of camera self-calibration. In: Proceedings of the IEEE Conference on AVSS (2003)
7. Zhang, W.: A simple method for 3D reconstruction from two views. In: GVIP 05 Conference, CICC, Cairo, Egypt, December 2005
8. Boudine, B., Kramm, S., El Akkad, N., Bensrhair, A., Saaidi, A., Satori, K.: A flexible technique based on fundamental matrix for camera self-calibration with variable intrinsic parameters from two views. J. Vis. Commun. Image R. **39**, 40–50 (2016)
9. El Akkad, N., Merras, M., Saaidi, A., Satori, K.: Camera self-calibration with varying intrinsic parameters by an unknown three-dimensional scene. Vis. Comput. **30**(5), 519–530 (2014)

10. El Akkad, N., Merras, M., Saaidi, A., Satori, K.: Camera self-calibration with varying parameters from two views. WSEAS Trans. Inf. Sci. Appl. **10**(11), 356–367 (2013)
11. Torr, P.H.S., Murray, D.W.: The development and comparison of robust methods for estimating the fundamental matrix. IJCV **24**, 271–300 (1997)
12. Trajkovic, M., Hedley, M.: Fast corner detection. Image Vis. Comput. **16**, 75–87 (1998)
13. Harris, C., Stephens, M.: A combined corner et edge detector. In: 4th Alvey vision Conference, pp. 147–151 (1988)
14. Smith, S.M., Brady, J.M.: A new approach to low level image processing. Int. J. Comput. Vis. **23**(1), 45–78 (1997)
15. Saaidi, A., Tairi, H., Satori, K.: Fast stereo matching using rectification and correlation techniques. In: ISCCSP, Second International Symposium on Communications, Control And Signal Processing, Marrakech, Morrocco, March 2006
16. Chambon, S., Crouzil, A.: Similarity measures for image matching despite occlusions in stereo vision. Pattern Recognit. **44**(9), 2063–2075 (2011)
17. Mattoccia, S., Tombari, F., Di Stefano, L.: Fast full-search equivalent template matching by enhanced bounded correlation. IEEE Trans. Image Process. **17**(4), 528–538 (2008)
18. Moré, J.J.: The Levenberg-Marquardt algorithm: implementation and theory. In: Watson, G. A. (ed.) Numerical Analysis. LNM, vol. 630, pp. 105–116. Springer, Heidelberg (1978). https://doi.org/10.1007/BFb0067700
19. El Akkad, N., El Hazzat, S., Saaidi, A., Satori, K.: Reconstruction of 3D scenes by camera self-calibration and using genetic algorithms. 3D Res. **7**, 6 (2016)
20. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: ORB: an efficient alternative to SIFT or SURF. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 2564–2571. IEEE (2011)
21. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: BRIEF: binary robust independent elementary features. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6314, pp. 778–792. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15561-1_56
22. Merras, M., Saaidi, A., El Akkad, N., Satori, K.: Multi-view 3D reconstruction and modeling of the unknown 3D scenes using genetic algorithms. Soft Comput. (2017). https://doi.org/10.1007/s00500-017-2966-z
23. El Hazzat, S., Merras, M., El Akkad, N., Saaidi, A., Satori, K.: 3D reconstruction system based on incremental structure from motion using a camera with varying parameters. Vis. Comput. (2017). https://doi.org/10.1007/s00371-017-1451-0
24. El Akkad, N., Merras, M., Baataoui, A., Saaidi, A., Satori, K.: Camera self-calibration having the varying parameters and based on homography of the plane at infinity. Multimed. Tools Appl. (2017). https://doi.org/10.1007/s11042-017-5012-3
25. El Akkad, N., El Hazzat, S., Saaidi, A., Satori, K.: Reconstruction of 3D scenes by camera self-calibration and using genetic algorithms. 3D Res. **7**(6), 1–17 (2016)
26. Leutenegger, S., Chli, M., Siegwart, R.Y.: BRISK: binary robust invariant scalable keypoints. In: 2011 IEEE International Conference on Computer Vision (ICCV). IEEE (2011)
27. Alahi, A., Ortiz, R., Vandergheynst, P.: Freak: fast retina keypoint. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (2012)
28. Rosin, P.L.: Measuring corner properties. Comput. Vis. Image Underst. **73**(2), 291–307 (1999)
29. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge (2004)