Calibration of Surveillance Cameras based on an Auxiliary Color-Depth Camera

Diogo da Silva Góis, José Gaspar
Instituto Superior Técnico / UTL, Lisbon, Portugal
diogo.gois@tecnico.ulisboa.pt, jag@isr.tecnico.ulisboa.pt

ABSTRACT

Camera networks provide large-scale video surveillance due to the summation of the fields-of-view. Usually, the fields-of-view are non-overlapping making difficult the tracking of targets even when their motion speed would allow predicting their future locations. That requires a global calibration methodology, encompassing both the intrinsic and extrinsic parameters of each camera on a common reference frame. This thesis is focused on the specific aspect of using an auxiliary color-depth camera to calibrate one camera at a time. The data acquisition device to perform this global calibration, consists of a pre-calibrated Pan-Tilt-Zoom (PTZ) camera with a laser pointer. This setup allows acquiring point clouds, on the areas corresponding to fields-of-view of the fixed cameras. The calibration methods are based on detecting and corresponding SIFT features and estimating the calibration parameters based on optimization methodologies, namely DLT-Lines and DLT-Points.

I. INTRODUCTION

Indoor and outdoor camera networks are nowadays ubiquitous. Current surveillance networks provide mostly video storage for viewing past events and not so much about identifying objects, people or activities, or predicting events. Calibration of the surveillance cameras can be an enabling technology for further applications.

Despite the existing extensive knowledge on camera calibration, the acquisition of calibration data is still not a commodity for networks of cameras. The usage of cameras like PTZ, that take advantage of the fact that they have pan, tilt and zoom degrees of freedom, provide a wide-area coverage and reveal greater details with the zoom-in feature while not needing the intensive and repetitive interaction of a human.

A network of cameras with non-overlapping fields-of-view must be calibrated in a universal coordinate system to be able to integrate information obtained by each camera. The universal coordinate system can be offered by a mobile robot with global localization and that observes the same scenario of the fixed cameras, [7].

This thesis aims to use an acquisition system consisting of a PTZ camera with a laser pointer rigidly attached at the end of a baseline aiming to calculate depths. These setups, allow acquiring both intrinsic and extrinsic parameters.

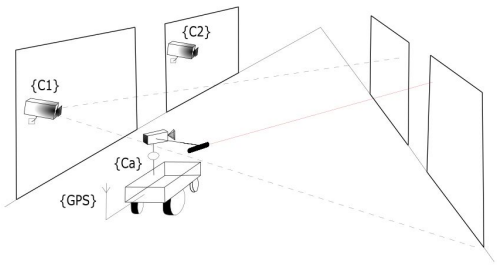


Fig. 1. An auxiliary mobile camera that can take depth measurements allows acquiring calibration information for fixed surveillance cameras.

II. RELATED WORK

The most common calibration methodologies are based on imaging a known pattern [2], [15]. In addition, most networked surveillance cameras do not have overlapping fields-of-view. Therefore, image information cannot be used directly to help the calibration of the network of cameras.

For instance, high-precision measuring devices can be used to obtain an accurate, global coordinate system, calibration. These devices can be theodolites, laser trackers or LRF.

In [14] was used a robot equipped with a laser sensor to track a secondary moving robot with a LED light equipped.

Our work, however, is motivated by seminal works with robots performing Simultaneous Localization And Mapping (SLAM). Laser Range Finders (LRF) mounted on robots, doing SLAM, are effectively high-precision measuring equipments. The reconstructed 3D point clouds may cover the complete area of network, and allow calibrating a set of cameras [10]. It is however a costly solution in terms of the cost of the LRF.

In [7] is proposed calibration methodology based on a Visual-SLAM. More in detail, the calibration of the network of cameras is based on a mobile robot capable of transforming its coordinate system to the set of fixed cameras. Our objective, and focus in this thesis, is therefore finding enough precise correspondences between a fixed surveillance camera and a mobile auxiliary camera.

RGBD auxiliary calibration cameras have been used as a lower cost alternative in relation to the laser, and as denser reconstructors when compared with Visual-SLAM, as seen in the thesis of Manuel Silva [12] and José Mendes [9]. In both works a color-depth auxiliary camera, namely the Microsoft Kinect, is used to help calibrate a RGB fixed camera.

However, one finds that off the shelf RGBD cameras just allow short distances.

In this thesis, the central objective is to calibrate a set of fixed surveillance cameras with non-overlapping fields of view. A global coordinate system is established by a mobile camera which acquires calibration data to register with imaging of the set of surveillance cameras. This camera can be a pan-and-tilt camera [13].

In [13] is noted the capability of pan-tilt-zoom cameras for 3D modeling of large scenes, reconstructing events and detecting activities within a large area.

In addition, we consider laser pointers instead of LRF, as another low cost control factor.

Given matched features, points or lines, between a fixed surveillance camera and an auxiliary color-depth camera, one wants to calibrate the surveillance camera. Feature points can be extracted e.g. with the SIFT algorithm.

The usage of an auxiliary mobile pan-tilt-zoom camera combined with a laser pointer involves controlling the pan and tilt angles for estimating depth values at desired points.

III. Background

This section introduces all the terminology, notation and methods used for the calibration process. In particular, it introduces aspects such as the back projection model, i.e the relationship between 2D image plane points and 3D points in the world and calibration methods such as DLT, used in other works [12], [9].

A. Camera Projection Model

The used camera model in this thesis is the pin-hole model, consisting of the camera being represented by a point, allowing the light from a point $M = [X\ Y\ Z\ 1]^T$ to pass through that point and finally being projected into the image plane, $m = [u\ v\ 1]^T$.

The intersection of the optical axis and the plane of the image is called the principal point and a point in the corresponding image is described by the intersection of the optical ray r, with the plane of the image. Therefore, the mapping that performs a projective projection of the three dimensional space to a two dimensional space is given by $u = fX/Z$ and $v = fY/Z$ where f is the camera focal length. Since an image is formed by pixels, one wants pixels and not meter coordinates. Therefore, using pixel coordinates we have $u = k_x x + u_0$ and $v = k_y y + v_0$. Being (x, y) the coordinates of the point in meters, (k_x, k_y) the scale factors of the pixels dimensions vertically and horizontally, respectively and (u_0, v_0) the coordinates of the principal point.

Using a world coordinate system instead of camera coordinate system, we know that the euclidean transformation between two images is given by a rotation plus translation. This can be defined by ${}^cM = {}^cR_w\ {}^wM + {}^ct_w$ being c and w indications of the camera and world coordinate systems respectively.

In conclusion, we can write the camera projection model as:

$$m \sim PM = K\left[R \mid t\right] M \tag{1}$$

in which $\sim$ denotes equal up to scale, P is the camera projection matrix (3x4) and R and t represent the rotation matrix (3x3) and the translation vector (3x1) respectively. These two together represent the camera extrinsic parameters.

B. Calibration based on Direct Linear Transform Methods

Aziz and Karara [1] developed the DLT algorithm that solves a linear system of equations stated in the previous section. This algorithm considers both sets of 3D points, $M_i = \begin{bmatrix} X_i & Y_i & Z_i & 1 \end{bmatrix}^T$, and their corresponding image points, $m_i = \begin{bmatrix} u_i & v_i & 1 \end{bmatrix}^T$, and obtains the projection matrix P. Previous works documented calibration procedures using this algorithm [12], [9]. In this section of this thesis is presented the difference between using points and lines for calibration with this method.

1) Calibration using DLT-Points: Given 1 and multiplying in both sides of the equation by the 2D points, m_i, results in $m_i \times m_i \sim m_i \times (PM_i)$, corresponding to the left hand side of the equation being equal to zero, leading to $[m_i]_\times PM_i = 0$, where $[m_i]_\times$ is the cross product operation as a skew-symmetric matrix based on m_i. The properties of Kronecker product documented in [8], $\otimes$, make it possible to rewrite the equation as:

$$(M_i^T \otimes [m_i]_\times) vec(P) = 0 \tag{2}$$

where $vec(P)$ is the vectorization of the matrix P, where the columns are stacked into a single column vector with dimension 12x1. For each pair of 3D and their matched 2D points, (M_i, m_i) we are allowed to write 2 and this provides the following system:

$$\underbrace{\begin{bmatrix} M_1 \otimes [m_1]_\times \\ \vdots \\ M_N \otimes [m_N]_\times \end{bmatrix}}_{A_{2N\times 12}} \underbrace{\begin{bmatrix} P_{11} \\ P_{12} \\ \dots \\ P_{34} \end{bmatrix}}_{P_{12\times 1}} = \underbrace{\begin{bmatrix} 0 \\ 0 \end{bmatrix}}_{b_{2N\times 1}} \tag{3}$$

Each of the N entries provides a set of three equations in the entries of $vec(P)$, but only two of them are linearly independent. In order to obtain the projection matrix P, we solve the resulting system of stacking N matrices, using at least six pairs of 3D and 2D matches points (N=6). The pre-normalization of the input data used on the implementation of this algorithm is crucial as noted in [5]. Hartley defined two main principles, the first in which all the data points used, have their centroids specifically at the origin. Secondly, that 3D points have an average distance of $\sqrt{3}$ and 2D points of $\sqrt{2}$ relative to the origin.

2) Calibration using DLT-Points with Radial Distortion: Many times cameras deviate from the pinhole model leading to radial distortion observed in the images captured. Since it is very hard to represent camera radial distortion effect with high precision it is necessary to find an approximation to the camera's true distortion. Fitzgibbon [4] proposed the

Division Model where from a radially distorted image point, $\hat{m}_d = [u_d \ v_d]^T$, you can compute an undistorted image point, $\hat{m}_u = [u_u \ v_u]^T$. More precisely, both points relate as $\hat{m_u} = \frac{\hat{m_d}}{(1+\lambda\|\hat{m_d}\|^2)}$, where λ represents the distortion parameter. This model written in homogeneous coordinates $[u_u \ v_u \ 1]^T \sim [u_d \ v_d \ (1 + \lambda\|\hat{m_d}\|^2)$ lets us state that an undistorted point is just a simple function of a distorted point, i.e, $m_u = m_d + \lambda\|e_d\|$, where $e_d = [0 \ 0 \ \|\hat{m_d}\|]^T$.

Both coordinates of the undistorted and distorted points are expressed in a 2D coordinate system having the origin coincident with the image principal point c_o. With the model defined, we can now add radial distortion to the DLT-Points calibration by merging 2 and $m_u = m_d + \lambda\|e_d\|$, resulting in $(M_i^T \otimes [m_{id} + \lambda e_{id}]_\times)vec(P) = 0$ which can be written in matrix form form as:

$$\begin{bmatrix} M_1 \otimes [m_{1d} + \lambda e_{1d}]_\times \\ ... \\ M_i \otimes [m_{id} + \lambda e_{id}]_\times \end{bmatrix} \begin{bmatrix} P_{11} \\ P_{12} \\ ... \\ P_{34} \end{bmatrix} = 0 \tag{4}$$

where $A_{i1} = M_i^T \otimes [m_{id}]_\times$ and $A_{i2} = M_i^T \otimes [e_{id}]_\times$. Again with the same methodology, if we consider N pairs of 3D-to-2D (M_i, m_i), one forms two matrices A_1 and A_2 with dimension 3Nx12. A Polynomial Eigenvalue Problem (PEP) is defined when multiplying both matrices by A_1^T:

$$(A_1^T + \lambda A_1^T A_2)vec(P) = 0 \tag{5}$$

which can be solved as a polynomial eigenvalue problem (SVD). Its solution gives simultaneously the projection matrix, $vec(P)$ and the radial distortion parameter λ.

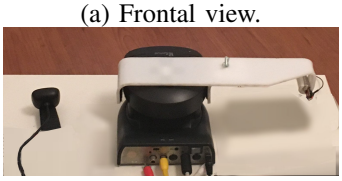
IV. Surveillance Camera Calibration

In this section is presented a methodology for calibrating a fixed surveillance camera with the help of an auxiliary PTZ camera that permits depth measurements with high precision. It is also detailed the calibration of the auxiliary camera, the method to obtain a homography between a real 3D world point and that same point captured in the image. The estimation of the laser line is also detailed.

A. Surveillance and Auxiliary Cameras Setup

It is proposed to construct a compact and low cost system capable of calibrating a network of surveillance cameras. This system is established by a mobile PTZ camera with a laser and a raspberry, demonstrated in the Fig. 2. The PTZ camera and laser pointer will be responsible for obtaining information for the calibration of the fixed camera. The raspberry serves as an intermediate communicator between setup and the computer in order to create a standalone system that can later be implemented in a moving robot. Cables such as USB Serial and Serial-MiniDin 8 Pin are used to connect the PTZ camera with the raspberry pi. USB grabber is used to be able to connect analogue video source, like the PTZ camera, to a computer like the raspberry via USB. The jump wires are used to connect the laser pointer to a circuit in a breadboard.

(a) Frontal view.



(b) Back view.

Fig. 2. Surveillance and auxiliary cameras setup.

B. Setup and Automated Calibration Process

A network of surveillance cameras is mostly formed by fixed cameras. Mobile surveillance cameras are typically a minority. In this text when we refer *surveillance cameras* we are considering just the fixed ones. Figure 1 show a set of fixed cameras, C_i and a mobile camera, C_a, which we term the auxiliary (color-depth) camera.

Our auxiliary camera setup is composed by two elements, a PTZ mobile camera and a laser pointer attached rigidly to the camera, Fig. 4. Is important to note that since the laser is attached to the camera, changing the pan and tilt angles will correspond to the laser changing its pose also. The laser allows to get depth measurements for a point of interest, and combined with the PTZ camera, the setup concedes a complete 3D data from the world.

Figure 1 represents the PTZ camera, C_a, with the laser to measure depths, and several C_i cameras that belong to the camera network. We want to allow C_a to be able to calibrate C_i. Assuming two cameras, a RGBD (color-depth) and a RGB, if they have overlapped fields-of-view we can use the RGBD camera and a set of 3D-to-2D match points to calibrate the RBG camera. This method combines the best of both worlds, allowing to obtain accurate calibration results when performing first a DLT based on points and then a DLT based on corresponding lines.

The first step consists of obtaining two images from both cameras. Then, by applying the SIFT algorithm to both images, are detected corresponding matching points (m_i, M_i) between them. Using the equation 2 from the DLT-Points calibration method and the previously mentioned matching points (at least 6) we can get a first estimation of the projection matrix P. The second step comprises on applying algorithms like Canny Edge Detector and Hough Transforms on both images again, but this time to detect lines. As stated before, the utilization of lines has the advantage of applying fitting and filtering processes. Therefore, we use the RANSAC algorithm to do 3D line fitting. Since we have already a first estimated projection matrix P it is possible to obtain corresponding matching between lines in the RGBD image that lie also in the RGB image. Finally for last step, combining both equations from DLT-Lines and DLT-Points respectively, one can perform a new estimation of projection matrix P that can also prove if the equation 2 is correct.

C. Acquiring 3D data with the Auxiliary Camera

In this section is detailed the acquisition of 3D data with the auxiliary color-depth camera. Recall that the auxiliary camera is formed by a PTZ camera having a laser pointer rigidly attached, at a baseline distance, to its moving head.

Considering that all the laser points lying in the laser ray, in homogenous coordinates are given as ${}^{L}\tilde{M} = [0 \;\; 0 \;\; {}^{L}Z \;\; 1]^T$. Those same points in relation to the camera frame have the same $z - axis$ and can be formulated as follows, ${}^{C}\tilde{M} = [b \;\; 0 \;\; {}^{L}Z \;\; 1]^T$, where b denotes the baseline, i.e, the distance between the camera's optical axis and the laser. Since the laser is attached to the camera, if the camera suffers changes in the pan and tilt angles, the laser will follow those actions. So, is possible to say that the laser points that lie in the laser ray will suffer also rotations of pan and tilt ${}^{w}M = {}^{w}R_c \begin{bmatrix} b & 0 & {}^{L}z \end{bmatrix}^T$ i.e. ${}^{w}\tilde{M} = [{}^{w}M^T \;\; 1]^T$ where ${}^{w}R_c$ represents the camera and laser rotation w.r.t the world coordinate system, calculated as ${}^{w}R_c = R_{pan}R_{tilt}$. Note that the pan rotation consists of a rotation in the $y - axis$, R_y, and a tilt rotation consists of a $x - axis$ rotation, R_x.

Given a 3D target point in the world, ${}^{w}\tilde{M}_0$, and two possible laser points, ${}^{w}\tilde{M}_1$ and ${}^{w}\tilde{M}_2$, with coordinates $z = 0$ and $z = 1$ respectively. Pointing the laser to the target point is only possible when trying different pan and tilt angles until the laser hits the desired point. This allows to conclude whether or not it corresponds with the point observed in the image.

A back-projection is done in order to find the ${}^{w}M_0$ coordinates, (X, Y, Z), where λ is a scalar, varied to find the target point ${}^{w}M_0 = C + \lambda D$ where C is the camera's projection center and D the direction of the projection ray to the target point. After the coordinates are calculated, angles of pan and tilt are tested to point the laser to the target point. Then, norms between ${}^{w}M_0$ and ${}^{w}M_2$, as well as between ${}^{w}M_2$ and ${}^{w}M_1$ are calculated as $V_{10} = {}^{w}M_0 - {}^{w}M_2$ and $V_{12} = {}^{w}M_2 - {}^{w}M_1$.

The distance V_0 is calculated as $V_0 = V_{10} - proj_{V_{12}}V_{10}$ where $proj_{V_{12}}V_{10}$ denotes the projection of V_{10} in V_{12}.

Finally, an optimization algorithm, Matlab fminsearch, is used to find the pan and tilt angles that lead to the minimum error between both vector V_{10} and V_{12}, calculated as $error = \|V_0\|$. If both vectors have the same direction then the error is zero and it is verified that the laser points to the target point, and is possible to confirm the 3D coordinates of the point, Fig. 3.

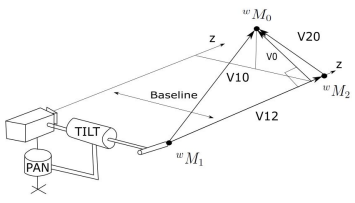


Fig. 3. Finding pan and tilt angles to the specific target point

D. Calibration of the Auxiliary Color-Depth Camera

In this section is detailed the protocol, namely the sequence of steps that should be taken for the calibration of the auxiliary camera used to calibrate a network of cameras. In order to obtain accurate 3D coordinates of the world through the laser, the method to evaluate if the laser goes through the desired point is also presented.

1) PTZ Camera Calibration using a Standard Toolbox: The first point that we have to consider is that we have to calibrate the auxiliary PTZ camera if we want good metric 3D measurements using the laser. In [11] is presented a camera calibration method based on the use of patterns, more precisely a chessboard. In this thesis, we follow that methodology, i.e, the observation of a chessboard in various poses as indicated by the Bouguet's calibration toolbox [2]. This camera calibration let us find the characteristics internal to the camera, C_a, and the camera's location in the 3D space with respect to the chessboard pattern.

2) Calibration Method: The calibration performed consists of the use of the dataset created in [3]. This dataset has several photographs of the 15x15 chess pattern in various poses and distances, taken with the PTZ camera.

For starters, the pattern has been fixed on a platform that will be moved to various distances in front of the auxiliary camera and that will apply angles to the pattern that will vary its pose. The distances used were 1m to 5m in relation to the location of the PTZ camera.

3) Laser-line pose w.r.t. the PTZ Camera: The PTZ camera combined with the laser point allows estimating depths. In order to do depth estimation, PTZ calibration needs to be complemented with an estimation of the laser-line pose w.r.t. the PTZ. Laser-line pose estimation is detailed in this section in two steps (i) estimate line pose assuming known 3D laser-line points, see Fig. 4, and (ii) estimating 3D laser-line points reflected on a chessboard pattern.

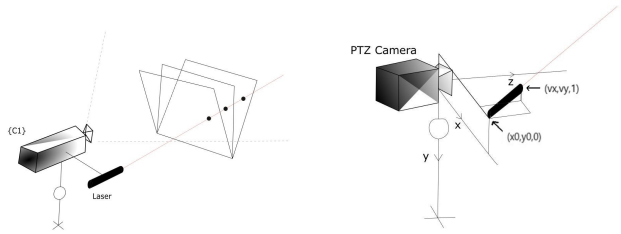


Fig. 4. Laser Reflection Points in the Chessboard at Different Poses (left). Laser not parallel to Oz (right).

4) Estimating the Laser Line: Given a set of 3D laser points $\{M_i\}$, the objective is to find the laser line. First are made 3 assumptions, (i) Points have been mapped to the camera coordinate system, $M_I = {}^{c}T_{F_i} \; {}^{F}M_i$, (ii) the laser is not perpendicular to the z axis of the camera, seen in the right image in Fig. 4 and finally (iii) coordinate of the laser z_0 = 0, possible by extending the laser line.

Let the laser line be represented as $[x_0 \;\; y_0 \;\; 0]^T + \alpha_i[v_x \; v_y \; 1]^T = M_i$, where α_i denotes a scale factor and M_i is computed from the previous homography. Given 4 unknowns, (x_0, y_0, v_x, v_y), one 3D laser point provides three equations plus one new unknown α_i. Note however that the unknown is a length along the z-axis:

$$\begin{cases} x_0 + \alpha_i v_x = M_{ix} \\ y_0 + \alpha_i v_y = M_{iy} \\ \alpha_i = M_{iz} \end{cases} \tag{6}$$

therefore, one has the matrix form:

$$\underbrace{\begin{bmatrix} 1 & 0 & M_{iz} & 0 \\ 0 & 1 & 0 & M_{iz} \end{bmatrix}}_{A_i} \begin{bmatrix} x_0 \\ y_0 \\ v_x \\ v_y \end{bmatrix} = \underbrace{\begin{bmatrix} M_{ix} \\ M_{iy} \end{bmatrix}}_{b_i} \tag{7}$$

The matrix form shows that each laser point provides two constraints on the four parameters. Considering $N \geq 2$ 3D points one estimates the laser line using least squares $[x_0 \ y_0 \ v_x \ v_y]^T = (A^T A)^{-1} A^T b$ where $A = [A_1 \ A_2 \ ... \ A_N]^T_{2N \times 4}$ and $b = [b_1 \ b_2 \ ... \ b_N]^T_{2N \times 4}$.

5) 3D Location of Laser Reflection Points: In order to obtain a good estimation of the 3D laser reflection points, M_i, a homography between a world point and image point is computed as follows. Given a specific chessboard pose, at a frame i that will be noted as F_i, with coordinates $(x_i, y_i, z_i = 0)$, can be seen in Fig. 5. To compute this homography a chessboard image is given in which the laser point can be seen. The first step consists in obtaining, four points in metric units, $m_i = [m_1 \ m_2 \ m_3 \ m_4]$, and four corresponding point matches to the same points in the image, $m_i^{'} = [m_1^{'} \ m_2^{'} \ m_3^{'} \ m_4^{'}]$ to compute the homography ${}^m H_p$. The image and world metric coordinates systems are defined as p and m, respectively. Note: Points from $m_{1:4}$ are points in the chessboard and point m_5 is the laser reflection point.

The relation between the points can be formulated as $m_m \sim {}^m H_p m_p$. Following the same methodology as in Section 2.2, multiplying m_m on both sides results in the left hand side of the equation being zero and thus, $[m_m]_\times \ {}^m H_p m_p = 0$, where $[m_m]_\times$ denotes the cross product operation as a skew-symmetric matrix based on m_m. Using the Kronecker properties, the previous equation can be rewritten as:

$$(m_p^T \otimes [m_m]_\times) vec({}^m H_p) = 0 \tag{8}$$

where $vec({}^m H_p)$ denotes the vectorization of the homography. To complete this direct mapping between image and world points is needed to compute the last desired point, m_5, with:

$$m_5 \sim \ {}^m H_p m_5^{'} \tag{9}$$

where ${}^m H_p$ is the previous computed homography.

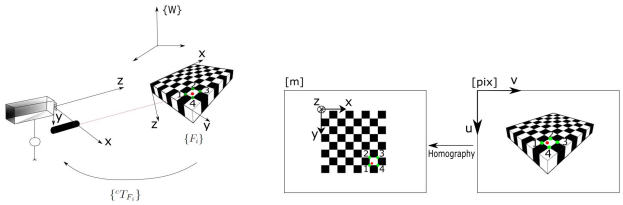


Fig. 5. Specific chessboard and 3D laser reflection point (left). Homography (right).

6) Surveillance Camera Calibration: The method to calibrate the surveillance camera goes through three points. Initially make use of the Matlab calibration toolbox and calibrate the camera using various photos of the chess pattern. This allows us to have a ground truth. Create a dataset of multiple images from both the PTZ camera and the Surveillance camera. They will be used for the detection of common points and consequently DLT calibration. A first DLT calibration of the camera using a user-dependent method.

7) Surveillance Camera Matlab Toolbox Calibration: About 50 images of the chessboard in different poses and at various distances were used in the Matlab's toolbox calibration.

8) Create a dataset with Images from PTZ and Surveillance: A dataset of images was taken to be able to detect information from laser points and perform DLT calibration with a large number of data. The process of capturing this dataset began by choosing a range of values for the pan and tilt angles, and for each pair (*pan*,*tilt*) an image was taken in both cameras with and without the laser on.

9) User-dependent DLT calibration: From the captured images in the dataset, 40 from both cameras were randomly chosen. The function *ginput* was used to store the value of the pixel of the laser point. We can compute the 3D points in the world frame by using the equation 16.

E. Improved Surveillance Camera Calibration

The main purpose of the improved method is to make the calibration and laser point detection algorithm user-free. To do this, we start by increasing data detection and doing it automatically. To reduce or eliminate outliers from Background Subtraction, it will be used the epipolar line and 3D reprojection to select regions where laser point can be. The automatic laser point detection in PTZ image uses the epipolar line as a guide line to know where is the area where the laser point is located.

Then, compute the difference between gray images with laser on and off of that region selected before. After this, converting the image to binary image, it allows us to identify the laser pixels. Find connected components, i.e, laser point pixels in binary image and compute the mean.

Since we have the laser points in the PTZ image, $(u, v)_P$, we can compute the 3D points in the world frame XYZ_{wf} using 16. In order to be able to detect the laser points in the images of the surveillance camera more easily, you take the 3D points in the world frame, XYZ_{wf}, and reproject them back in the images taken by the camera, $m_{rs} = K_s \times \begin{bmatrix} R_s & t_s \end{bmatrix} \times XYZ_{wf}$. Following the same logic as before, select region of interest of where m_{rs} is. To find the exact laser point, $(u, v)_S$, compute the difference between gray images of that area chosen and finally get binary image to find the pixels, $(u, v)_S$. In the end, with all the 2D points, $(u, v)_S$ and their corresponding 3D, XYZ_{wf}, is possible to perform a new DLT-Points calibration.

F. Relation between disparity and depth

In a stereo vision setup, the perception of depth arises from disparity of a given 3D point in your right and left images.

In this setup, we have two camera axes that are parallel with a baseline *(b)*. The depth, z, is given by $z = fb/d$ where f is the focal length and d is the disparity the difference of the same 3D point imaged in both images. Since the disparity is inversely proportional to depth we can immediately state that the disparity decreases as depth increases. An important note that we want to analyze is how can zoom improve the disparity.

In our case, our setup resembles stereo vision, the only difference is that the second camera is defined only by a laser ray, $P_2 = [R\ t]$, where R is the identity matrix and $t = [b\ 0\ 0]^T$.

V. Auxiliary Camera Pose Estimation

In this section we want to be able to prove if it is possible, with the setup that we built, to get poses along a certain path. To perform this estimation are used 3D lines of the environment where the camera is located together with algorithms such as procrustes.

Initially, the setup was positioned in two different locations, where in each of them 3D information was collected. For both positions, we choose two areas of interest to get points. In these areas of interest, the objective is to find 3D lines, called roof edges. These 3D lines are, in more detail, the result from the intersection of two planes.

Let us consider a simple scenario where a box near a wall is imaged from two viewpoints. One edge of the box and a floor-to-wall line form two *roof edges* which are used to estimate the camera relative motion between the two viewpoints. In a first step 3D points are reconstructed. The 3D points are used to compute four planes which define two *roof edges*. The planes are defined as $a_i \times x + b_i \times y + c_i \times z + d_i, i \in 1, 2, 3, 4$. The two intersection lines are represented as $M_{ij} + a_i \times v_i, i \in 1, 2, j \in 3, 4$.

Compute the new line, C, that defines the minimum distance between them A and B. Given M_{13}, M_{23}, v_1 and v_2 is possible to obtain two more 3D points along the lines A and B, as $M_{14} = M_{13} + v_1$ and $M_{24} = M_{23} + v_2$.

Being M_{P1} and M_{P2} two vectors that hold the 3D points along the roof edges defined of pose 1 and 2 respectively. These will be the input information of the procrustes algorithm that determines a linear transformation of the points in M_{P2} to best conform them to the points in M_{P1}.

VI. 3D Reconstruction

With the calibrated setup, this section tries to prove that the built-in setup is able to capture 3D data to make 3D reconstruction of a certain environment.

Although the PTZ camera plus laser pointer behaves like a stereo vision setup, the calculations for getting a 3D point in the world frame differ from each other. Considering the case of stereo vision using the pin-hole model we have $m_1 \sim P_1 M$ and $m_2 \sim P_2 M$. The projection equations can be rewriten in matrix form:

$$\begin{bmatrix} [m_1]_\times P_1 M \\ [m_2]_\times P_2 M \end{bmatrix} = \underline{0} \Leftrightarrow A_{6\times4} M_{4\times1} = \underline{0}_{6\times1} \tag{10}$$

The 3D point, M, is obtained as the right singular vector corresponding to the least singular value of A.

Our setup, represented in Fig. 6, being camera $\{C_1\}$ plus laser $\{C_2\}$ is inspired in the stereo vision setup. Although equations follow the same idea, in this case we have two different approaches, (i) Laser ray parallel to z-axis, just has a baseline w.r.t the camera and does not have orientation so: $P_2 = [I\ t]$ and (ii) Laser ray not parallel to z-axis. In this case laser presents some orientation along x and y axis, so: $P_2 = [R\ t]$.

Although these approaches have different projection matrices, both must take into account that $m_2 = [0\ 0\ 1]^T$

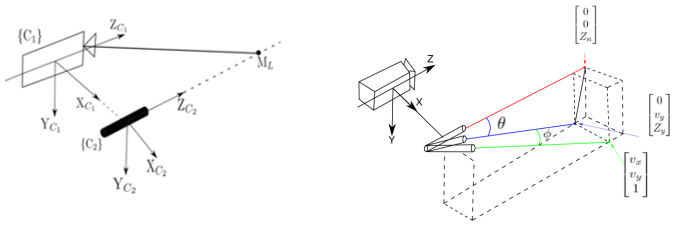


Fig. 6. PTZ Camera and Laser Setup (left). Rotation between laser projection and world referential (right).

For the **first approach**, the t translation vector denotes the baseline of the laser as $t = [-b\ 0\ 0]^T$. The 3D laser point is obtained as $M_{laser} = \begin{bmatrix} b & 0 & Z_i \end{bmatrix}^T$.

Imaged by the PTZ camera, C_1, we get:

$$\begin{bmatrix} \gamma u_1 \\ \gamma u_2 \\ \gamma \end{bmatrix} = \begin{bmatrix} fS_x & 0 & C_x \\ 0 & fS_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} b \\ 0 \\ Z_i \\ 1 \end{bmatrix} \tag{11}$$

For the **second** approach, the laser has translation and rotation applied as in our case, like in right image in the Fig. 6.

The rotation in both x and y axis, $R = R_y R_x = R_\phi R_\theta$ will affect the position of $[0\ 0\ Z_n]^T$ as $(R_\phi \cdot R_\theta) \cdot \begin{bmatrix} 0 & 0 & Z_n \end{bmatrix}^T = \begin{bmatrix} v_x & v_y & 1 \end{bmatrix}^T$.

Let us consider the baseline of the laser ray by $[x_0\ y_0\ 0]^T$, and the rotation $R_{wl} = R_\phi \cdot R_\theta$.

$$P2 = K \begin{bmatrix} R_{wl} & \begin{matrix} x_0 \\ y_0 \\ 0 \end{matrix} \end{bmatrix} \tag{12}$$

The points along the laser ray are given by $M_i = P_2 \begin{bmatrix} 0 & 0 & Z & 1 \end{bmatrix}^T$ where $Z = norm(Z_i \cdot \begin{bmatrix} v_x & v_y & 1 \end{bmatrix})$

The points projected by the laser can also be expressed in the world by $M_{laser} = \begin{bmatrix} x_0 + Z_i v_x & y_0 + Z_i v_y & Z_i \end{bmatrix}^T$.

Where v_x is the rotation along x-axis and v_y the rotation along y-axis. Imaged by PTZ camera, C_1 we get:

$$\begin{bmatrix} \gamma u_1 \\ \gamma u_2 \\ \gamma \end{bmatrix} = \begin{bmatrix} fS_x & 0 & C_x \\ 0 & fS_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_0 + Z_i v_x \\ y_0 + Z_i v_y \\ Z_i \\ 1 \end{bmatrix} \tag{13}$$

Initially, the camera is calibrated using the dataset created in [3]. Then, choose a range of pan and tilt values that cover the scenario with scenario to reconstruct. For each pair of values $(pan, tilt)$ take an image with the camera and using the background subtraction to get the pixel, (u, v), of the laser point. Repeating this process for all the pan and tilt values chosen, calculate the 3D coordinates of the laser points with the K matrix. First, we obtain the depth estimation as follows:

$$Z = \frac{f \cdot S_x \times x_0}{u_{pixel} - f \cdot S_x \times v_x - C_x} \tag{14}$$

By substituting the value Z_i in the equation 15 we compute the 3D point of the laser in the camera frame.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} x_0 + v_x \cdot Z \\ y_0 + v_y \cdot Z \\ Z_i \end{bmatrix} \tag{15}$$

In order to obtain these 3D points in the world frame, apply the R_{wc} rotation matrix as:

$$M_{p_i} = R_{wc_i} \cdot \begin{bmatrix} x_0 + v_x \cdot Z_i \\ y_0 + v_y \cdot Z_i \\ Z_i \end{bmatrix}, \tag{16}$$

where R_{wc_i} is the rotation matrix that contains the $(pan, tilt)_i$ pair values applied to the PTZ camera.

VII. Imaging Noise Effect on Pose Accuracy

Applications for home users, require low cost alternatives to the nowadays expensive depth estimation setups as total stations or laser range finders. The MS Kinect gained acceptance as a depth measuring device after the study [6] has characterized the measurement error to be lesser that $4cm$ for the working range of the camera, about $20cm$ to $5m$. In this section we look also for a reconstruction error characterization for our color-point-depth device, PTZ and laser pointer, in order to assess whether our device can effectively be a low cost depth measuring alternative.

1) Starting the Study with the Real Setup: Considering the objective of assessing the calibration accuracy of our setup, one first step is to select a calibration dataset acquired by the setup. In M. Bento's work [3], a large dataset was created for the purpose of calibrating the PTZ camera combined with the laser pointer, and calibrating a surveillance. This dataset closely represents the real setup, but its noise prevents studying the effect of gradually larger noise levels, starting from a noiseless situation.

2) Creation of the Noiseless Calibration Data: Given the previous dataset, we want to create a similar dataset that is noiseless, and therefore allows studying the effect of gradually adding higher noise levels starting at the noiseless case.

An initial calibration is performed based on the original dataset and the calibration method DLT-Points. One obtains projection matrices for the PTZ and Surveillance cameras, P_1 and P_2, respectively. The noiseless dataset is created by assuming the 3D points $\{M\}$ and matrices P_1 and P_2 are correct (noise-free), and by computing the projected points

$$\{m_1\} = \{m_1 :\ m_1 = h^{-1}(P_1\ h(M)), \quad M \in \{M\}\} \tag{17}$$

$$\{m_2\} = \{m_2 :\ m_2 = h^{-1}(P_2\ h(M)), \quad M \in \{M\}\} \tag{18}$$

where $h(\cdot)$ and $h^{-1}(\cdot)$ denote point homogenization and dehomogenization, respectively.

If one would proceed backwards by doing a DLT-Points calibration for the PTZ camera given the points $\{M\}$ and $\{m_1\}$, that would lead to $P_1 = K_P[R_P\ t_P]$, equal to the result of the original calibration, with noisy data, as in essence $\{m_1\}$ were computed from P_1 and $\{M\}$. Similarly, $\{M\}$ and $\{m_2\}$, would lead to P_2.

The 3D data $\{M\}$ contains also laser reflection points, which in real operations are the only ones available to calibrate the surveillance camera. In the original dataset those reflection points are all on a single 3D line, as the calibration dataset has the PTZ standing still, it is a chessboard pattern being posed at various depths and orientations. Calibration of the surveillance camera requires 3D data not constrained to a single 3D line and not constrained to be in a single plane. Hence, in the following, are simulated small rotations of the PTZ camera, to obtain 3D data on multiple 3D lines and at multiple depths.

To calibrate the surveillance camera using DLT-points one needs characteristic 2D points which are projections of 3D points. In this case, the laser pointer reflection points in the original dataset are used but, as referred, it is necessary to create multiple PTZ and laser poses. Therefore, various rotations are applied to the PTZ Camera. The laser ray is rigidly mounted, i.e. moves with the camera. The intersection of the laser ray with the various chess patterns is obtained, resulting in 3D reflection points of the laser pointer with $M_i = R_i P_1 M, \quad i \in \{0, 1, 2, 3\}$. where M is in the set $\{M\}$ restricted to the laser reflection points, $R_0 = I_{3\times3}$, R_1 is a pan incremental rotation, R_2 is a tilt incremental rotation and R_3 is a pan followed by a tilt incremental rotations.

Each rotation implies a set of 3D points, $\{M_0\}$, $\{M_1\}$, $\{M_2\}$ and $\{M_3\}$, and their images on the surveillance camera. The noiseless 2D points images of the laser pointer reflections are obtained with

$$m_{si} = P_2 M_i, \quad i \in \{0, 1, 2, 3\}, \tag{19}$$

forming the surveillance camera images, sets of 2D points $\{m_{s0}\}$, $\{m_{s1}\}$, $\{m_{s2}\}$ and $\{m_{s3}\}$. The projection matrix, $P_2 = K_S[R_S\ t_S]$, is equal to the one from the original dataset. Finally, the calibration of the surveillance camera is obtained with DLT-Points given $\{(m_s, M_s)\}$ where

$$m_s \in \{m_{s0}\} \cup \{m_{s1}\} \cup \{m_{s2}\} \cup \{m_{s3}\} \tag{20}$$

$$M_s \in \{M_0\} \cup \{M_1\} \cup \{M_2\} \cup \{M_3\} \quad . \tag{21}$$

After the steps detailed in this section we have the noiseless input data for both cameras.

3) Add Noise to the Calibration Data: Our objective is to study how a model responds to inputs having Gaussian noise, that is, to study the parameters of the cameras when adding noise to the input data that enter the calibration functions.

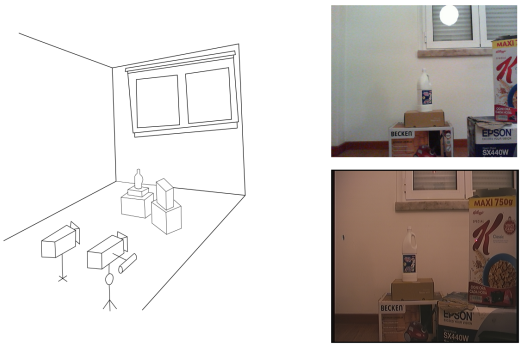


Fig. 7. Scenario and fixed and auxiliary cameras (left). Surveillance Camera Image (right top) and PTZ Camera Image (right bottom).

Given the noiseless 2D points, expressions 17 and 20, Gaussian noise is added with zero mean and variance, σ_G^2 that varies for both PTZ and Surveillance.

$$\begin{aligned} m_{1n} &= m_1 + \eta_1, \quad \eta_1 \sim N(0, \sigma_{\eta 1}^2) \\ m_{sn} &= m_s + \eta_s, \quad \eta_s \sim N(0, \sigma_{\eta s}^2) \end{aligned} \tag{22}$$

The image input data, to which noise will be added, is $\{m_1\} \cup \{m_s\}$, denoting image points on the PTZ and surveillance cameras, respectively.

By introducing noise in the input data, noise is caused in the estimates of PTZ and Surveillance poses. The 3D data can also contribute to uncertainty on pose estimates. More in detail, two levels of study can be considered here, (i) we can assume the 3D data is provided noiseless to each of the cameras, expression 21, and (ii) similarly to our setup, 3D data used by the second camera (the surveillance camera) is reconstructed by the first camera.

In order to see the cascaded noise effect in the surveillance camera, we need to work on 3D noise. More in detail, we need to move M_s to PTZ coordinate system as:

$$M_{s_2} = R_{PTZ_n} M_s + t_{PTZ_n}, \tag{23}$$

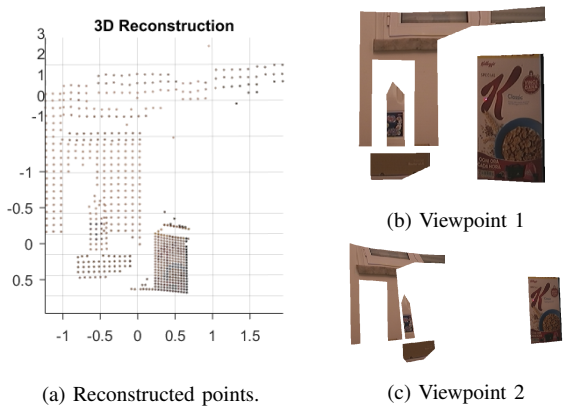
where R_{PTZ_n} and t_{PTZ_n} are the rotation and translation matrices obtained from the noisy PTZ calibration and M_{s_2} are the M_s in the PTZ coordinates frame.

VIII. Experiments and Results

In this section is presented the results obtained from the experiments that were performed in order to validate the methodologies and techniques presented in section III.

A. 3D Reconstruction

In order to assess the capabilities of the auxiliary camera for three-dimensional reconstruction an experiment was carried out. This experiment consists of putting the camera at a distance of approximately 2 m from a wall with some objects to reconstruct (Fig. 7). The experiments starts by choosing a range of pan and tilt values to cover the scene to reconstruct in Fig. 7. Use Background Subtraction to obtain the laser reflection points. Since we have, K matrix and Laser Poses, is possible to compute 3D coordinates of the correspondent laser point. We can then apply the rotation matrix $R_{wc} = R_{pan} \cdot R_{tilt}$ to the 3D Points we can compute the laser point in the world frame. Finally, create a 3D point cloud with respective RGB values.



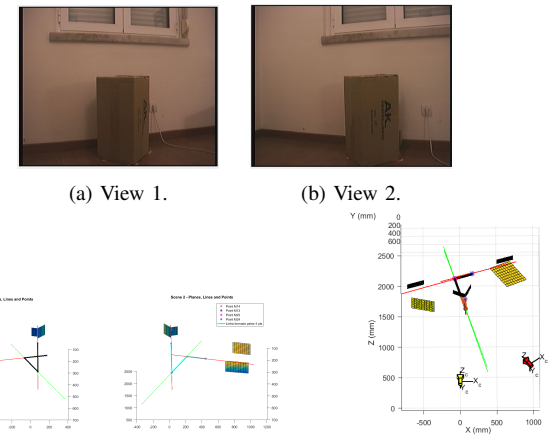
(a) Reconstructed points. (b) Viewpoint 1 (c) Viewpoint 2

Fig. 8. 3D Reconstruction using the auxiliary color-depth camera, consisting of one pan-tilt-zoom camera combined with one laser pointer.

Is possible to create the virtual reality model of a 2D image by having the corresponding 3D points of what we want to rebuild. In our case, one can also get a 3D model. The quality is not the best but from the texture of the image and the 3D points obtained to make the reconstruction one can obtain the result in Fig. 8 (b) and (c).

B. Auxiliary Camera Pose Estimation

In the following figure we can see the results of the PTZ Pose Estimation process explained in section III, for both camera positions (scene 1 and scene 2).



(a) View 1. (b) View 2. (c) Pose 1. (d) Pose 2. (e) Poses 1 & 2

Fig. 9. Pose estimation. Images acquired at zero pan and tilt angles at two viewpoints (a,b). Reconstructed planes and lines generated for the two viewpoints (c,d). Fusion of the 3D data acquired at the two viewpoints, including the two camera poses (e).

$$R_{proc1} = \begin{bmatrix} 0.7958 & -0.0079 & 0.6056 \\ 0.0295 & 0.9992 & -0.0257 \\ -0.6049 & 0.0383 & 0.7954 \end{bmatrix}, \; t_{proc1} = \begin{bmatrix} -942.37 \\ -65.94 \\ 326.66 \end{bmatrix} \tag{24}$$

Applying the rotation matrix and translation vector from the procrustes algorithm to the planes, lines and points from Scene 2 give the results seen in the bottom image in the Fig. 9. Note that there are two cameras plotted. The yellow and red cameras are defined as $P_1 = K[I_1\ 0]$ and $P_2 = K[R_{proc2}\ t_{proc2}]$, respectively, with

$$R_{proc2} = \begin{bmatrix} 0.7958 & 0.0295 & -0.6049 \\ -0.0079 & 0.9992 & 0.0383 \\ 0.6056 & -0.0257 & 0.7954 \end{bmatrix}, \ t_{proc2} = \begin{bmatrix} -942.37 \\ -65.94 \\ 326.66 \end{bmatrix} \tag{25}$$

computed from $R_{proc2} = R_{proc1}^{-1}$ and $t_{proc2} = -R_{proc1}^{-1} t_{proc1}$. Looking at 25, we get a pose estimation of the red camera. This camera is approximately 94.94 cm w.r.t the yellow camera in the x-axis, 4.59 cm in the y-axis and 30.92 cm in the z-axis.

C. Surveillance Camera Calibration

1) Surveillance Camera Matlab Toolbox Calibration: Placing the 50 images in the Matlab's toolbox gives the intrinsic matrix:

$$K_{surv} = \begin{bmatrix} 845.19 & 0 & 427.76 \\ 0 & 844.56 & 285.53 \\ 0 & 0 & 1 \end{bmatrix} \tag{26}$$

2) User-dependent DLT calibration: Having the 2D points from *ginput* and the 3D point correspondences perform a first DLT Calibration, from which the following matrices K, R and t were obtained:

$$K_{surv} = \begin{bmatrix} 837.34 & -0.37 & 395.42 \\ 0 & 872.07 & 331.20 \\ 0 & 0 & 1 \end{bmatrix} \tag{27}$$

$$R_{surv} = \begin{bmatrix} 0.9995 & 0.0135 & 0.0295 \\ -0.0143 & 0.9995 & 0.0284 \\ -0.0291 & -0.0288 & 0.9992 \end{bmatrix}, \quad t_{surv} = \begin{bmatrix} 146.12 \\ -45.61 \\ -38.34 \end{bmatrix} \tag{28}$$

Making a first analysis with this calibration the resulting extrinsic parameters, t vector, presents an error in the x-axis of approximately 1.4 cm and 1.8 cm in the y-axis, to the ground truth measured. In Fig. 10 depicts the extrinsic parameters of both cameras.

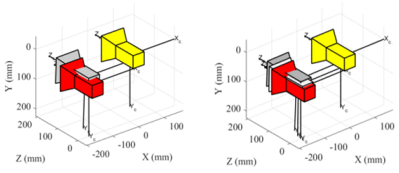


Fig. 10. Surveillance Camera Extrinsics for First DLT Calibration (left). Surveillance Camera Extrinsics for Second DLT Calibration (right). In yellow is the PTZ camera calibrated from Matlab's toolbox, the surveillance camera calibrated with Matlab's toolbox and from DLT-Points calibration are in red and grey, respectively. The white camera presented at the right image is the surveillance camera that resulted from the improved DLT-Point calibration.

3) Improved Surveillance Camera Calibration: Doing the process for all the images we get 2021 points. With all 3D laser points in the world frame and the corresponding 2D laser points, performing DLT-Points calibration:

$$K_{surv} = \begin{bmatrix} 842.46 & -4.29 & 398.11 \\ 0 & 889.62 & 354.21 \\ 0 & 0 & 1 \end{bmatrix} \tag{29}$$

$$R_{surv} = \begin{bmatrix} 0.9997 & 0.0127 & 0.0210 \\ -0.0128 & 0.9990 & 0.0064 \\ -0.0210 & -0.0066 & 0.9998 \end{bmatrix}, \quad t_{surv} = \begin{bmatrix} 156.80 \\ -49.97 \\ -16.64 \end{bmatrix} \tag{30}$$

In the right image in Fig. 10, a white camera is defined by 30 is closer to the ground truth in red. Looking at the results of the improved DLT calibration, we can confirm that the error on the x axis has a error of approximately 0.4 cm and 1.4 cm on the y axis, i.e, a better estimation.

In Fig. 11 is represented a more complete pointcloud of the scenario (a) and respective depth map (b), from the data obtained with this process.

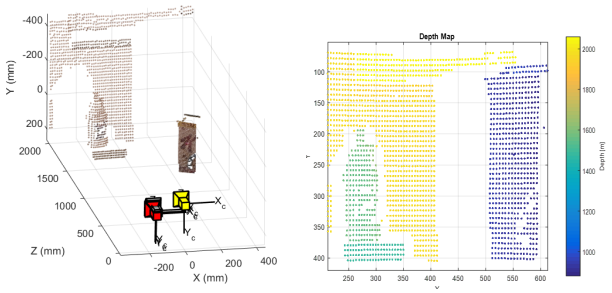


Fig. 11. PointCloud of the Scenario and Camera Positions (left). Depth Map (right).

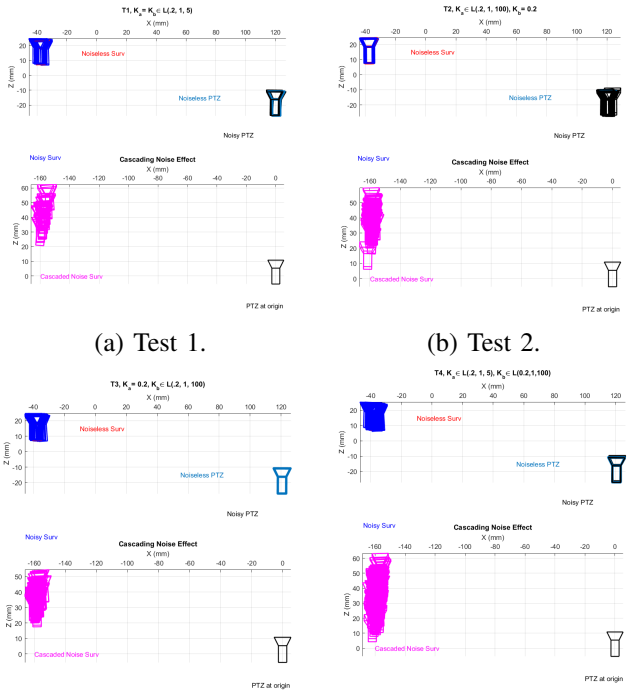
D. Imaging Noise Effect on Pose Accuracy

Initially, four different values were chosen for k_a and k_b to see how noise affects the extrinsic parameters of both cameras. For the four tests performed, the variance values were for Test $1 : k_a = k_b = L(0.2, 1, 5)$, Test $2 : k_a = L(0.2, 1, 100), k_b = 0.2$, Test $3 : k_a = 0.2, k_b = L(0.2, 1, 100)$ and finally for Test $4 : k_a = L(0.2, 1, 5), k_b = L(0.2, 1, 100)$.

In Fig. 12 we can see how the PTZ Camera noisy increases the noise in the surveillance camera pose estimation. In each image in the figure 12, there are two subplots. In the top subplot is the case where no cascading noise is used and in the bottom subplot is the case of cascading noise.

Analyzing the figure 12, it is stated that when bringing the noise that exists in the estimation of the PTZ camera pose and accumulating it in the pose of the estimated surveillance camera, greatly increases the estimation error of the surveillance pose. For each test, the magenta stain of the surveillance camera is dispersed more along the Z axis, which proves the effect of cascading noise.

To make a characterization or precision label of our setup, by choosing a pair of values (k_a, k_b) and repeat the process



(a) Test 1. (b) Test 2.

(c) Test 3. (d) Test 4.

Fig. 12. PTZ and Surveillance Cameras Extrinsic Parameters for Non Cascading and Cascading Noise.

performed for each test but this time it repeats N number of times to accumulate the errors and calculate the average.

For $k_a = k_b = L(0, 2, 10)$, the experiment is repeated 100 times, obtaining the graphs in Fig. 13 (a). The Fig.13 (b) is a mesh plot of the average error of the surveillance camera pose as a function of the chosen variance values. The error of the estimated pose of the surveillance camera increases as the weight of k_a noise increases. This happens since the noise in the estimation of the PTZ camera had weight of k_a was accumulate in the surveillance camera. It can thus be said that if there are erroneous estimates in relation to the estimation of the PTZ camera of our setup, these errors will be reflected in the estimates of the network of cameras that our setup wants to calibrate.

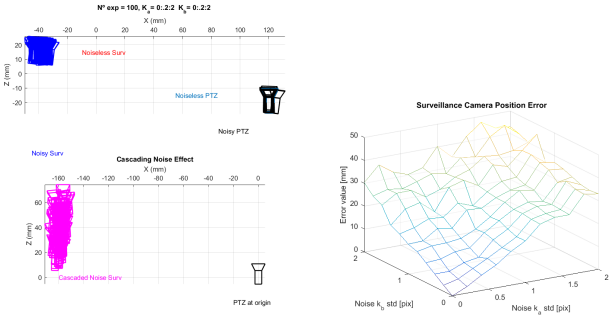


Fig. 13. PTZ and Surveillance Cameras Extrinsic Parameters for Non Cascading and Cascading Noise (left). Mean error of the Surveillance Camera Pose (right).

To finish, by analyzing Fig. 13 it can be stated that for maximum distances of $5m$, adding error of 0.2 [pix] to our setup causes error in fixed camera calibrations. This error is reflected in pose estimates of fixed cameras to be calibrated, with an uncertainty error of approximately $25mm$.

IX. Conclusion and Future Work

The work described in this thesis proposal focus in the calibration methodology of a fixed surveillance camera network with non-overlapping fields of view. Is used a mobile auxiliary camera which provides 3D calibration data and that is capable of establishing a universal coordinate system. Therefore, is possible that each camera from the network can give useful information.

The mobile auxiliary camera consists of a PTZ camera with a rigidly attached laser pointer. It was proposed automating the process of calibration by optimizing the data registration method that uses the algorithm DLT-Points.

The critical step that have been identified in this thesis report is how can the auxiliary camera help calibrate one surveillance camera at a time. In particular we detailed how to retrieve 3D data with the help of the laser while changing the pan and tilt degrees of freedom of the PTZ camera.

As future work, zoom could be further explored. It could be studied how zoom affects in 3D reconstruction, camera position estimation and setup characterization.

References

[1] A. Aziz and H. Karara. Direct linear transformation into object space coordinates in close-range photogrametry. *In Proc. of the Symposium on Close-Range Photogrammetry*, pages 1–18, 1971.

[2] Jean-Yves Bouguet. Camera calibration toolbox for matlab. http://www.vision.caltech.edu/bouguetj.

[3] Mário António da Silva Bento. Fast setting of networks of surveillance cameras. *Master's Thesis*, Instituto Superior Técnico, 2019.

[4] A. Fitzgibbon. Simultaneous linear estimation of multiple view geometry and lens distortion. *In Proc. IEEE Conf. Comp. Vision and Pattern Recognition*, VOL. 1, pages 125-132, 2001.

[5] R. I. Hartley. In defense of the eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(6):580-593, 1997.

[6] Kourosh Khoshelham and Sander Oude Elberink. Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors*, 2012.

[7] Nuno Miguel Pinto Leite. Calibra cão de uma rede de câmaras baseada em odometria visual. *Master's Thesis*, Instituto Superior Técnico, 2009.

[8] Helmut Lütkepohl. *Handbook of matrices*, volume 1. Wiley Chichester, 1996.

[9] José Mendes. Forensic use of mobile phone cameras:measuring the height of a person. *Master's Thesis*, Instituto Superior Técnico, 2017.

[10] Agustín Ortega, Manuel Silva, Ernesto H. Teniente, Ricardo Ferreira, Alexandre Bernardino, José Gaspar, and Juan Andrade-Cetto. Calibration of and outdoor distributed camera network with a 3d point cloud. *Sensors*, VOL. 14, pages 13708-13729, 2014.

[11] D. Samper, J. Santolaria, F. J. Brosed, A. C. Majarena, and J. J. Aguilar. Analysis of tsai calibration method using two and three dimensional calibration objects. *Mach. Vis. Appl.*, VOL. 24, pages 127-131, 2013.

[12] Manuel Silva. Camera calibration using a color-depth camera: Points and lines based dlt including radial distortion. *Master's Thesis*, Instituto Superior Técnico, 2012.

[13] Sudipta N. Sinha and Marc Pollefeys. Pan-tilt-zoom camera calibration and high-resolution mosaic generation. Department of Computer Science, University of North Carolina at Chapel Hill, 2008.

[14] Tsuyoshi Yokoya, Tsutomu Hasegawa, and Ryo Kurazume. Calibration of distributed vision network in unified coordinate system by mobile robots. *IEEE International Conference on Robotics and Automation, ICRA*, pages 1412-1417, 2008.

[15] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330 - 1334, 2002.