

Article

Not peer-reviewed version

A Novel 3D Reconstruction Sensor Using a Diving Lamp and a Camera for Underwater Cave Exploration

[Quentin Massone](#) , [Sebastien Druon](#) , [Jean Triboulet](#) *

Posted Date: 2 May 2024

doi: 10.20944/preprints202405.0041.v1

Keywords: Structured light method; active 3D reconstruction; cone estimation; underwater karst aquifer



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

A Novel 3D Reconstruction Sensor Using a Diving Lamp and a Camera for Underwater Cave Exploration

Quentin Massone , Sébastien Druon  and Jean Triboulet * 

Laboratory of Informatics, Robotics and MicroElectronics (LIRMM) (UMR 5506 CNRS—UM), University of Montpellier, 161 rue Ada, CEDEX 5, 34392 Montpellier, France; quentin.massone@gmail.com (Q.M.); sebastien.druon@lirmm.fr (S.D.); jean.triboulet@lirmm.fr (J.T.)

* Correspondence: jean.triboulet@lirmm.fr

Abstract: Aquifer karstic structures, due to their complex nature, present significant challenges in accurately mapping their intricate features. Traditional methods often rely on invasive techniques or sophisticated equipment, limiting accessibility and feasibility. In this paper, we propose a new approach to non-invasive, low-cost 3D reconstruction using a camera that observes the light projection of a simple diving lamp. Our method capitalizes on the principles of structured light, leveraging the projection of light contours onto the karstic surfaces. By capturing the resultant light patterns with a camera, we reconstruct three-dimensional representations of the structures. The simplicity and portability of the equipment required make this method highly versatile, enabling deployment in diverse underwater environments. We validate our approach through extensive field experiments conducted in various aquifer karstic settings. The results demonstrate the efficacy of our method in accurately delineating intricate karstic features with remarkable detail and resolution. Furthermore, the non-destructive nature of this technique minimizes disturbance to delicate aquatic ecosystems while providing valuable insights into the subterranean landscape. This innovative methodology not only offers a cost-effective and non-invasive means of mapping aquifer karstic structures but also opens avenues for comprehensive environmental monitoring and resource management. Its potential applications span hydrogeological studies, environmental conservation efforts, and sustainable water resource management in karstic terrains worldwide.

Keywords: Structured light method; active 3D reconstruction; cone estimation; underwater karst aquifer

1. Introduction

Access to water has become one of the most pressing challenges facing humanity today. Yet water is abundant on a global scale: the oceans are an inexhaustible reserve, provided that salt and water are separated. Unfortunately, desalination methods are still too energy-intensive to be used profitably on a large scale. Pending a possible scientific breakthrough, we are therefore relying essentially on freshwater resources, which account for 2.5% of the world's water [1] (Figure 1).

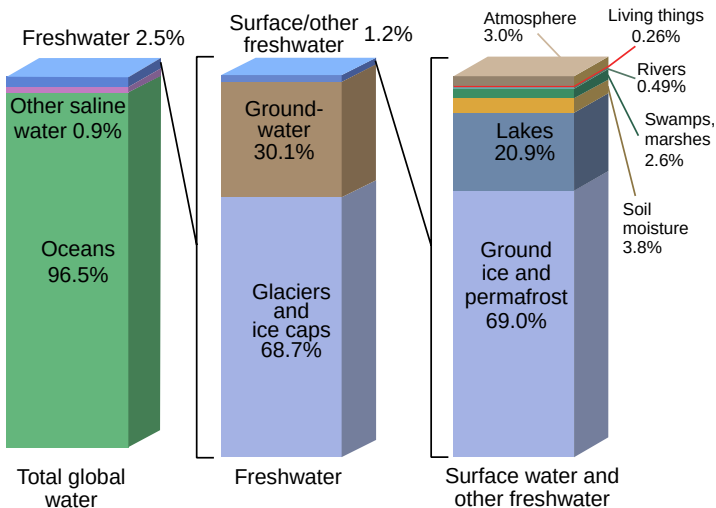


Figure 1. Earth's water distribution.

Only 1.2% of this freshwater is accessible at the surface, although it is now very polluted. Of the remainder, 68.7% is trapped in glaciers and ice caps and 30.1% is found underground. This groundwater, which represents 0.76% of the world’s resources, is extremely valuable. Groundwater is the result of a long and natural process of filtration through the various layers of soil, which means that its quality is generally considered to be very good. This groundwater is stored in aquifers, most of which are karstic.

Aquifer karstic structures, due to their complex and porous nature, present significant challenges in accurately mapping their intricate features. Traditional methods often rely on invasive techniques or sophisticated equipment, limiting accessibility and feasibility. In this paper, we propose a novel, non-invasive approach utilizing structured light projection in conjunction with a camera and a diving light to effectively map aquifer karstic formations. First part presents the main methods in underwater mapping. Second part details materials and methods of our original approach based on the principles of structured light, leveraging the precise projection of light contours onto the karstic surfaces. By capturing the resultant light patterns with a camera, we reconstruct detailed three-dimensional representations of the subsurface structures. Third part presents results to validate our approach by simulation of an aquifer galley first and finally through extensive outdoor experiments conducted in a particular equivalent aquifer karstic environment : The results demonstrate the efficacy of our method in accurately delineating intricate karstic features with remarkable detail and resolution. Furthermore, the non-destructive nature of this technique minimizes disturbance to delicate aquatic ecosystems while providing valuable insights into the subterranean landscape. This innovative methodology not only offers a cost-effective and non-invasive means of mapping aquifer karstic structures but also opens avenues for comprehensive environmental monitoring and resource management. Its potential applications span hydrogeological studies, environmental conservation efforts, and sustainable water resource management in karstic terrains worldwide.

2. Main Methods

Underwater mapping is a valuable tool in a wide range of applications, including marine biology, geology, archaeology, and the offshore industry. Various methods and sensors are employed to obtain 3D reconstructions of the environment, with data typically collected by divers, ROVs, or AUVs. In most cases, this involves performing surveys with downward-facing sensors and reconstructing the seafloor surface. The studies conducted by [2,3] provide comprehensive reviews of the various underwater 3D reconstruction methods. This synthesis of methods is illustrated in Figure 2.

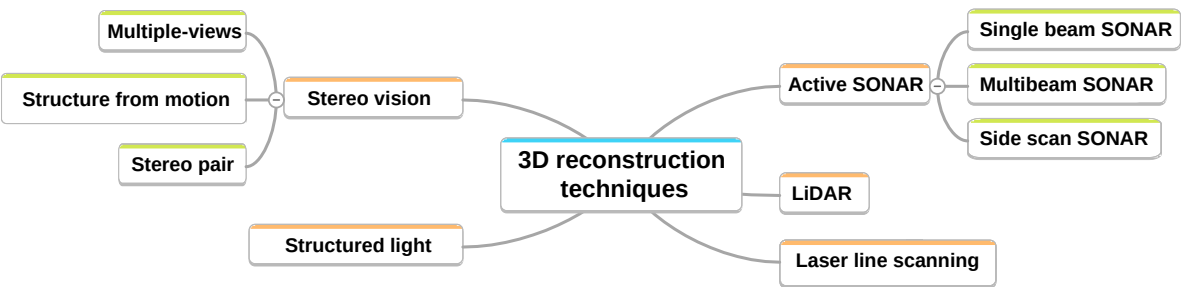


Figure 2. Review of 3D Reconstruction Methods in Underwater Environments.

Sensors used for 3D reconstruction can be classified into two main categories:

- active sensors: these sensors physically interact with the environment by emitting a signal or wave, such as sound waves in the case of sonars or light waves in the case of lidars.
- passive sensors: these sensors do not modify the environment, and the most commonly used ones for 3D reconstruction are camera-based.

Time-of-flight methods are the most direct methods for 3D reconstruction. They work on the same principle as a rangefinder: emitting a signal or wave and retrieving its echo. Knowing the speed of the

signal thanks to the nature of the wave and the propagation medium, it is possible to proportionally deduce the relative distance between the sensor and the observed element from the measurement of the time elapsed between the emission of the wave and the return of the echo.

Among time-of-flight methods, those based on acoustic waves are by far the most used because their propagation is favored by the medium. They offer an interesting coverage capacity in front of the size of the areas to be studied in underwater mapping.

In addition and to allow the localization of 3D data during the measurement, most underwater navigation algorithms are based on acoustic positioning systems such as Doppler Velocity Log (DVL) and Ultra-short baseline (USBL) [4,5]. Multibeam sonars (MBS) are on the other hand adapted to bathymetric mapping (measurement of the seabed relief) [6] but much less to structured (non-flat) environments such as karst aquifers.

[7,8] presents a 3D model of the Zacaton cenote (a water-filled cave at least 300 meters deep) using sonar data collected by the DEPTHX (Deep Phreatic THERmal eXplorer) vehicle. It is a NASA autonomous underwater vehicle measuring just over 2m and weighing over a ton, prefiguring among the robots that could explore underwater environments on other planets. Its sensors consist of a DVL, a sonar, an IMU and its movements are ensured by six motors allowing it to move in all directions.

Another method using acoustics is described by [9], which shows the results of an AUV performing limited penetration, inside an underwater cave using a mechanically scanned imaging sonar (MSIS for mechanically scanned imaging sonar). The results, obtained only for localization, show that it is possible to use this type of technology in our karst environment.

Efficient for obtaining bathymetry and macro-characteristics of habitats, these sensors nevertheless provide little information on fine characteristics (micro-bathymetry), do not allow to recover the texture of the environment and are very expensive.

Still in the class of time-of-flight methods, those based on electromagnetic waves, such as LiDAR, offer access to a very dense reconstruction of the seabed. However, on the other hand, they are reserved for very local observations, because the propagation of these waves are limited by the medium as we explained in the constraints. They are therefore not adapted to our situation.

The other class of methods is the triangulation method that we favor. These methods require at least two devices (or 2 different views) which will provide distance information from the same point, and thus form a triangle between the 2 devices and the point to be measured.

There are several ways to perform this triangulation, including:

- Multi-view triangulation involves taking multiple views of the same scene. By knowing the disparities of each viewpoint, it is possible to project the points of the scene onto different "image" planes, and thus triangulate the points of the scene to obtain a depth map. This technique is based on epipolar geometry explained in [10] with two possible approaches:
 - the use of a binocular stereo system where the relative positions of the two cameras are known. It is possible to make the system active, by using a structured light projector to add discriminant points or features related to the observed environment and thus facilitate the matching of points between the two images for triangulation.
 - the method known as Structure from Motion (SfM) involves taking a sequence of sequential images of an object or scene, typically from a single camera. To obtain a metric reconstruction, it is necessary to know the camera trajectory to resolve the scale ambiguity.
- structured light triangulation relies on an emitter-receiver system. The emitter is a light source (laser or not) that projects a pattern (distinctive patterns) onto a scene observed by the receiver, a camera. Using the positions of the camera and the projector, it is possible to triangulate the discriminant points brought by the light in the scene captured by the camera [11].

Regarding structured light triangulation methods, they are very accurate, but are generally only used to reconstruct small areas or objects. This is because light is quickly absorbed in water; therefore, we

must be close to the target to best detect the projected light. Additionally, structured light sensors are not yet well suited for mobility and are therefore mostly used statically for the moment.

Cameras, on the other hand, have the advantage of being inexpensive, providing dense data as well as texture. However, most camera-based methods operate in open areas with natural lighting (or with artificial lighting that completely illuminates the workspace) and generally try to reconstruct the ground or small objects.

For example, [12,13] propose a 3D reconstruction method based on a stereo pair for archaeological objects in shallow water and in very poor conditions. Special filters are implemented to solve the noise problems caused by these poor conditions. This static 3D measurement is fused with a 3D map of the excavation area obtained using a scanning sonar. [14] performs a dense reconstruction of submerged structures at very high depths (up to 6000 meters) with a stereo system mounted on a controlled manipulator arm. The imposition of a known trajectory allows to know the different positions of the cameras to calculate a precise 3D model of the scene.

[15] reconstructs coral reefs with a wide-baseline stereo system using high-resolution video and a dense reconstruction method.

Our context is different from these works since we are in total darkness in a structured and extended environment whose content appears as the diver or robot moves.

[16] presents a description of the problems related to metrology in the underwater environment. The importance of calibrating the stereo pair underwater is emphasized, taking into account the refraction problems related to the medium. The detection of interest points is done on sufficiently bright images allowing a robust matching then used for a 3D reconstruction. The comparison of the object measurements obtained is made with a reference metrology and gives good results but in statics.

The work of [17] is directly related to our theme. The objective is to reconstruct a karstic underwater tunnel of *Woodville Karst Plain* in Florida using a new approach to stereovision generating a 3D point cloud of the cave. In their method, after calibration and rectification of the highly distorted images, they extract the contours of the artificial light projected into the cave which creates a cone of light illuminating part of the walls. These contours are then used as inputs in their stereovision algorithm. The displacement is estimated by visual odometry (ORB-SLAM) and allows a 3D reconstruction of a portion of the cave of about one hundred meters but without specifying the comparison of the result with a known ground truth.

For an even more optimal mapping of this underwater gallery, [18] have developed a SLAM method where they fuse this stereo method with a sonar and an inertial measurement unit, thus improving the accuracy of their measurements.

[19] uses a SFM method coupled with multi-view stereo photography to perform measurements of silting areas at the outlet of water transport pipelines to the seabed. The accuracy of their method allows a robust estimation of these elements useful for monitoring these areas but also remains static measurement.

[20] provides an overview of calibration and underwater image processing methods in structured light. [21] proposes a method of 3D reconstruction in underwater environments combining a SLAM approach and dense reconstruction using a stereo system embedded on the Aqua2 robot whose IMU allows localization. The results are mainly on coral reefs and partially on the reconstruction of underwater cavities. The question of the accuracy of the measurements with respect to a ground truth remains. [22] proposes an application of 3D reconstruction for underwater archeology. The system consists of two monochrome cameras and a sinusoidal light pattern projector. A combination of 3D reconstruction in structured light and SLAM-based visual odometry is proposed. All mounted on an ROV whose IMU data complements that of visual odometry. The first results are obtained on objects in a controlled environment. A test is then carried out over a dozen meters to follow a pseudo pipeline. But the size of the system is too large for an application in underground galleries.

[23] uses Ariane's thread to enable the movement and guidance of a BlueRov robot. This learning method gives good results for localisation and assisted piloting. The next steps envisaged are autonomous navigation and then mapping of the cavities studied (but already explored by divers).

[24] proposes a real-time monocular 3D reconstruction method for underwater environments. It relies on optical flow tracking of characteristic seabed points and uses Delaunay triangulation to complete the seabed estimation. The method assumes a nearly planar seabed but remains unclear about the achieved metric accuracy.

3. Materials and Methods

The complete development of our method is detailed in Figure 3 and is based on three main blocks :

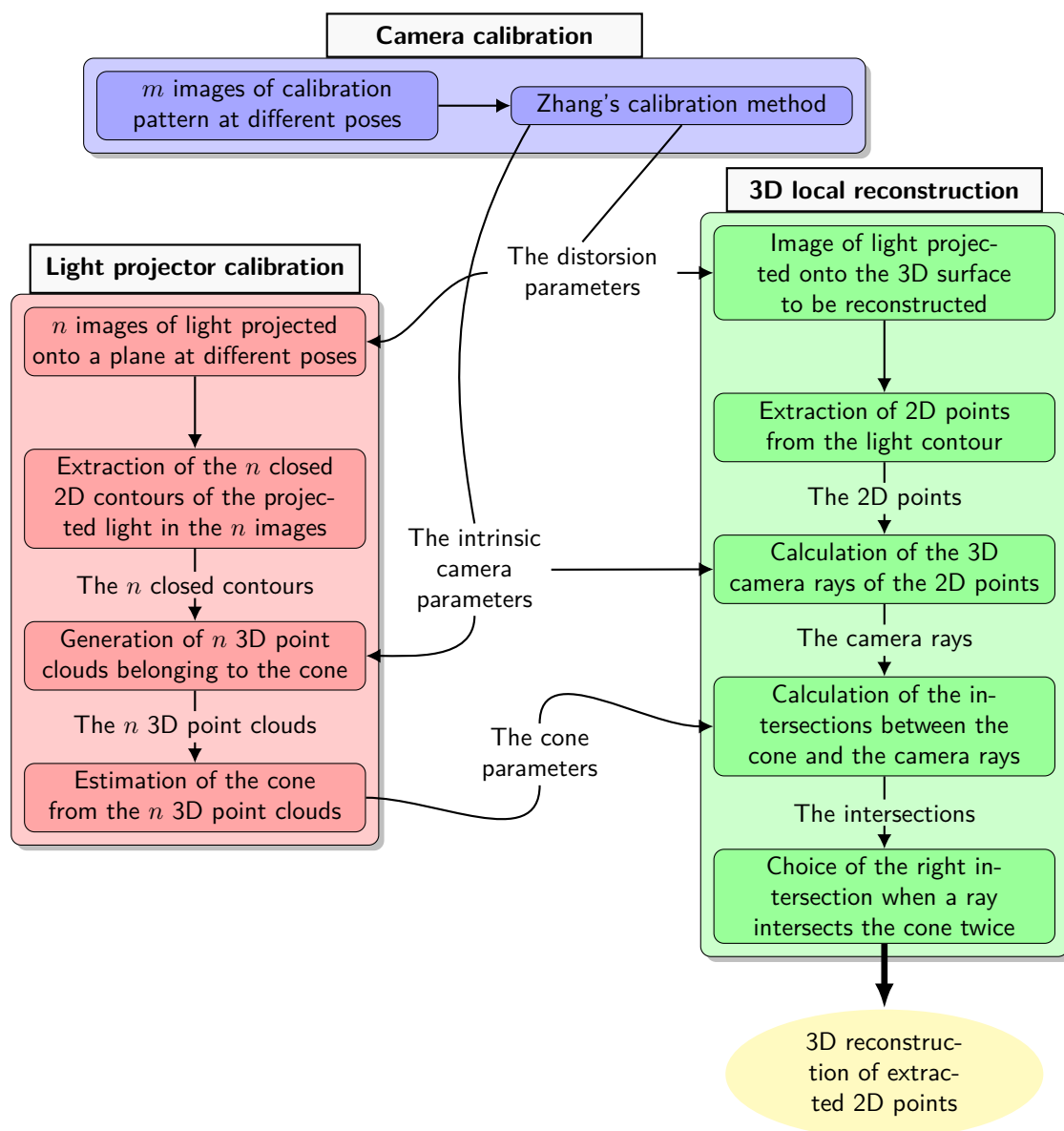


Figure 3. Flowchart of the camera + projector method.

- calibration of the camera is used to estimate its intrinsic parameters, including radial distortion,
- calibration of the projector consists of estimating the geometric parameters of the cone (vertex $O\mathcal{P}$, direction d , half-angle of opening α),

- local 3D reconstruction based on the camera / projector triangulation leading to a 3D point cloud expressed in \mathcal{C} , the camera frame.

This triangulation is illustrated in Figure 4. It shows our system, with one camera and light projector. To simplify the figure, the light is projected onto a plane, but you can imagine that the procedure has to be applied to non-planar surfaces such as the walls of a gallery. Once the light contour has been detected in the image, the associated camera ray is calculated for each point on the contour. The Figure 4 shows two points X' and Y' belonging to the contour detected in the image and the rays associated with them. In the configuration shown in the Figure 4, the ray associated with X' (respectively Y') intersects the cone at two points X_1 and X_2 (respectively Y_1 and Y_2). To calculate these intersections we need to know the angle of aperture α and the transformation between the references of the cone and the camera called $T_{\{\mathcal{P} \rightarrow \mathcal{C}\}}$.

Still on the Figure 4, the intersections which correspond to the 3D points of the contour (X_1 and Y_2) are shown in green and those which should not be taken into account (Y_1) are shown in red. The point X_2 is not shown, but we can imagine that it would be red and beyond the area of the plane in the Figure 4. For a given ray, it is therefore necessary to identify whether it is the first intersection or the second intersection that belongs to the observed light contour.

This method of selecting intersections requires a study of the geometric relationships between the cone and the camera.

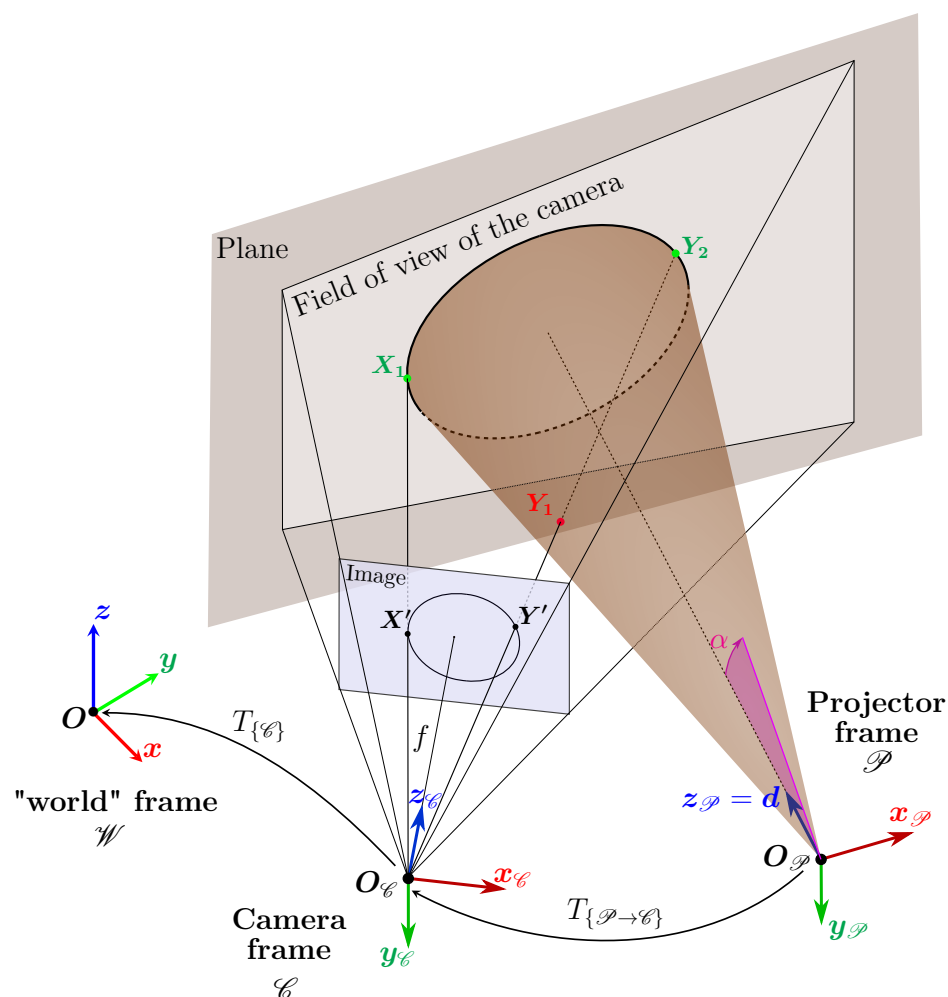


Figure 4. Diagram of the system consisting of the light projector represented by a cone of revolution and the camera observing the light projection on a plane.

3.1. Cone Parameters

3.1.1. Cone of Revolution

The cone of revolution, referred to as a cone in the remainder of this document, is associated with its surface of revolution and not with its solid, as may be the case in certain applications. The terms we will use to define the relationship between a point and the cone are as follows:

- a point belonging to the cone is a point which belongs to the surface of the cone,
- a point inside the cone is a point which belongs to the solid bounded by the surface of the cone,
- a point outside the cone is a point which does not belong to the solid bounded by the surface of the cone.

This surface is generated by the revolution of a line, called the generatrix, which we will note g and which passes through a vertex, in this case $O_{\mathcal{P}}$.

The revolution takes place around a fixed axis which also passes through the vertex and which turns out to be an axis of symmetry of the cone, the direction of which is defined by the unit vector d .

If a plane not passing through $O_{\mathcal{P}}$ is orthogonal to the axis of symmetry, then its intersection with the cone is a circle.

Thus, the angle formed by g and the axis of symmetry is constant and corresponds to the half-angle of the cone opening $\alpha \in]0; \frac{\pi}{2}[$.

The cone is represented in Figure 5 with the orthonormal base $(O_{\mathcal{P}}, x_{\mathcal{P}}, y_{\mathcal{P}}, z_{\mathcal{P}})$, oriented such that $z_{\mathcal{P}} = d$, which defines the \mathcal{P} reference frame associated with the cone.

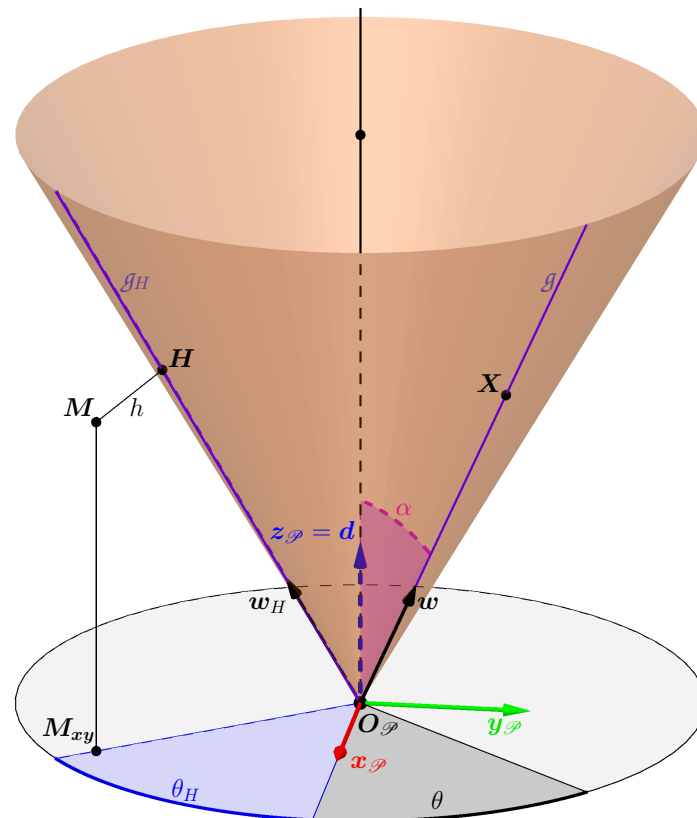


Figure 5. Parametrisation of the cone using its generatrices. With this parametrisation, we can find the closest generatrix to an external point M and thus find the orthogonal projection H of this point on the cone.

The generatrix g has the unit direction vector w which is set by a new angle $\theta \in [0; 2\pi[$ (Figure 5).

In the cone frame \mathcal{P} , we can simply express the vector $w_{\{\mathcal{P}\}}$ which then depends only on the angles θ and α :

$$w_{\{\mathcal{P}\}} = \begin{bmatrix} \sin(\alpha) \cos(\theta) \\ \sin(\alpha) \sin(\theta) \\ \cos(\alpha) \end{bmatrix} \quad (1)$$

This vector can be used to express all the points on the surface parametrically.

Let $X_{\{\mathcal{P}\}} = [x_{X_{\{\mathcal{P}\}}} \quad y_{X_{\{\mathcal{P}\}}} \quad z_{X_{\{\mathcal{P}\}}}]^T$ be a point X expressed in the reference frame \mathcal{P} . Then, if X belongs to the cone we have :

$$X_{\{\mathcal{P}\}} = l w_{\{\mathcal{P}\}} \Leftrightarrow \begin{cases} x_{X_{\{\mathcal{P}\}}} = l \sin(\alpha) \cos(\theta) \\ y_{X_{\{\mathcal{P}\}}} = l \sin(\alpha) \sin(\theta) \\ z_{X_{\{\mathcal{P}\}}} = l \cos(\alpha) \end{cases} \quad (2)$$

with l a real parameter whose absolute value defines the distance between $X_{\{\mathcal{P}\}}$ and the cone vertex. However, with $l \in \mathbb{R}$, the parametric equation defines a "double cone". However, since the cone models our light projector, we are only interested in the upper part of the "double cone", i.e. when the parameter $l \in \mathbb{R}^+$.

3.1.2. Orthogonal Distance

This parametrisation of the cone will help us to write the equation for the orthogonal distance between a point and the cone, an equation that will be useful for cone estimation.

Let M be a point outside the cone and H its orthogonal projection on the cone (see Figure 5).

The angle θ_H is the angle that gives the generatrix of the cone closest to M . This generatrix, called g_H , passes through H and has the unit direction vector w_H whose expression in the \mathcal{P} reference frame is :

$$w_{H\{\mathcal{P}\}} = \begin{bmatrix} \sin(\alpha) \cos(\theta_H) \\ \sin(\alpha) \sin(\theta_H) \\ \cos(\alpha) \end{bmatrix}$$

If we also express M in \mathcal{P} with $M_{\{\mathcal{P}\}} = [x_{M_{\{\mathcal{P}\}}} \quad y_{M_{\{\mathcal{P}\}}} \quad z_{M_{\{\mathcal{P}\}}}]^T$, we obtain the following expressions:

$$\begin{cases} \theta_H = \arctan\left(\frac{y_{M_{\{\mathcal{P}\}}}}{x_{M_{\{\mathcal{P}\}}}}\right) \\ H_{\{\mathcal{P}\}} = (M_{\{\mathcal{P}\}}^T \cdot w_{H\{\mathcal{P}\}}) w_{H\{\mathcal{P}\}} \end{cases} \quad (3)$$

If $M_{\{\mathcal{P}\}}^T \cdot w_{H\{\mathcal{P}\}} < 0$ then the projected point is in the lower part of the cone. As we are only working with the upper part of the cone, when this condition is true, the nearest point is the vertex of the $O_{\mathcal{P}}$ cone (see example Figure 6). The orthogonal distance h is therefore :

$$h = \begin{cases} \sqrt{(M_{\{\mathcal{P}\}} - H_{\{\mathcal{P}\}})^T (M_{\{\mathcal{P}\}} - H_{\{\mathcal{P}\}})} & \text{si } M_{\{\mathcal{P}\}}^T w_{H\{\mathcal{P}\}} \geq 0 \\ \sqrt{M_{\{\mathcal{P}\}}^T M_{\{\mathcal{P}\}}} & \text{si } M_{\{\mathcal{P}\}}^T w_{H\{\mathcal{P}\}} < 0 \end{cases} \quad (4)$$

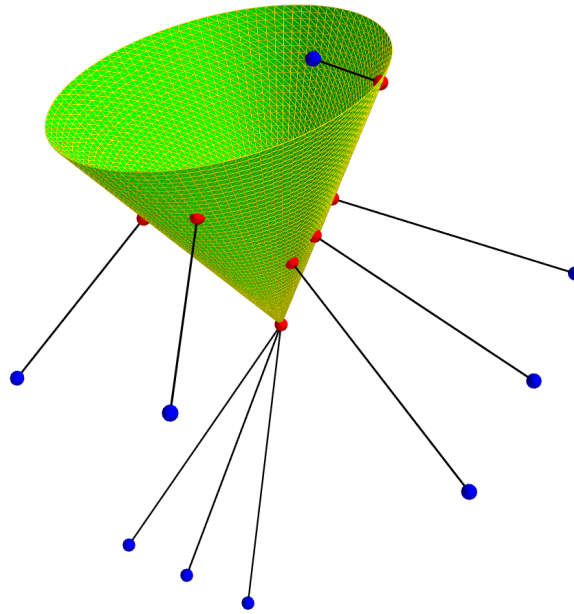


Figure 6. Example showing the closest points on the upper part of the cone (in red) to points outside the surface of the cone (in blue).

3.1.3. Quadratic form of the Cone

To obtain this relationship, we must first note that if the angle between the vectors $(X - O_{\mathcal{P}})$ and d or $(X - O_{\mathcal{P}})$ and $-d$ is equal to the angle α , then the point X belongs to the cone. This is equivalent to the following scalar product:

$$(X - O_{\mathcal{P}})^T d = \pm \|X - O_{\mathcal{P}}\| \cos(\alpha)$$

When squared, the expression becomes :

$$\begin{aligned} \left((X - O_{\mathcal{P}})^T d \right)^2 &= (\|X - O_{\mathcal{P}}\| \cos(\alpha))^2 \\ \Leftrightarrow ((X - O_{\mathcal{P}})^T d)((X - O_{\mathcal{P}})^T d) &= (X - O_{\mathcal{P}})^T (X - O_{\mathcal{P}}) \cos^2(\alpha) \\ \Leftrightarrow (X - O_{\mathcal{P}})^T d d^T (X - O_{\mathcal{P}}) - (X - O_{\mathcal{P}})^T (X - O_{\mathcal{P}}) \cos^2(\alpha) &= 0 \\ \Leftrightarrow (X - O_{\mathcal{P}})^T \underbrace{[d d^T - \cos^2(\alpha) I_3]}_Q (X - O_{\mathcal{P}}) &= 0 \end{aligned}$$

We thus obtain a quadratic function of the cone in the general case, directly showing the parameters d , α and $O_{\mathcal{P}}$ which define it:

$$Q(X) = (X - O_{\mathcal{P}})^T Q (X - O_{\mathcal{P}}) \quad (5)$$

This gives the relationships between the point X and the cone :

$$\bullet \text{ If } \mathcal{Q}(X) = 0 \text{ then } X \text{ belongs to the cone} \quad (6a)$$

$$\bullet \text{ If } \mathcal{Q}(X) < 0 \text{ then } X \text{ is inside the cone} \quad (6b)$$

$$\bullet \text{ If } \mathcal{Q}(X) > 0 \text{ then } X \text{ is outside the cone} \quad (6c)$$

$$\bullet \text{ If } \mathcal{Q}(X) = 0 \text{ and } (X - O_{\mathcal{P}})^T d > 0 \text{ then } X \text{ belongs to the upper part of the cone} \quad (6d)$$

$$\bullet \text{ If } \mathcal{Q}(X) < 0 \text{ and } (X - O_{\mathcal{P}})^T d > 0 \text{ then } X \text{ is inside the upper part of the cone} \quad (6e)$$

$$\bullet \text{ If } \mathcal{Q}(X) > 0 \text{ or } (X - O_{\mathcal{P}})^T d < 0 \text{ then } X \text{ is outside the upper part of the cone} \quad (6f)$$

3.2. Projector Calibration

Projector calibration is an essential step, since it involves estimating the parameters of the cone used in our method and in the equations we have seen so far. As a reminder, the parameters of the cone to be estimated are its vertex $O_{\mathcal{P}}$, its direction vector d and its half-angle of aperture α .

Since $O_{\mathcal{P}}$ and d represent the relative pose of the cone with respect to the camera, the projector must be fixed with respect to the camera during calibration and during any experimentation. Furthermore, if the projector has a variable half-angle aperture, it is also necessary to ensure that this is fixed.

3.2.1. 3D Point Generation for Cone Estimation

The first step in the calibration method is to generate a cloud of 3D points belonging to the cone. To achieve it, the chosen method is to capture several light projection images on a flat surface.

This light projection is the result of the intersection between the projector cone and the flat surface, which is by definition a conic. In our case, it will be an ellipse because we want a closed shape. The projection of this ellipse into the image is also an ellipse.

If we can estimate the relative pose of this surface with respect to the camera and extract the light contour in the image, which is an ellipse, then we can obtain the real ellipse from the ellipse in the image.

To estimate the relative pose, we attached a chessboard pattern to the surface to estimate its relative pose. This relative pose is obtained by estimating the homography between the calibration pattern and the image plane [25]. As the camera is calibrated, rotation and translation can be estimated from the homography.

To extract the ellipse from the image, we again use the contour detection method presented in [26,27].

Then, from the contour points obtained, we estimate the best ellipse using the method of [28] which is based on least squares optimisation.

It is therefore possible to obtain n elliptical sections of the cone from n images taken at different distances from the surface. For each ellipse extracted in an image, we obtain the ellipse in the surface via the estimated homography between the image plane and this surface. The estimated relative pose allows us to obtain the elliptical section, i.e. the set of 3D points expressed in the camera frame of reference belonging to the ellipse of the surface.

Figure 8 shows an example where the camera has captured three images of an ellipse surrounding a chessboard pattern on the flat surface in three different poses. This is equivalent to obtaining three elliptical sections of the same cone.

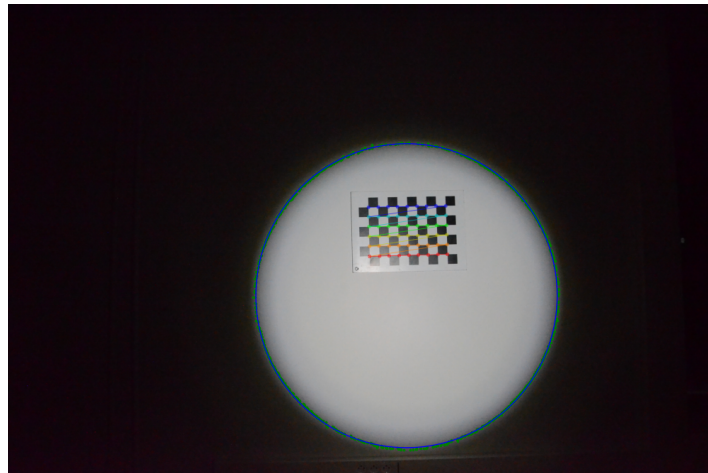


Figure 7. Example of detecting the contour of a halo of light on a wall using a chessboard to estimate the plane pose. The contour is in green and the adjusted ellipse in blue.

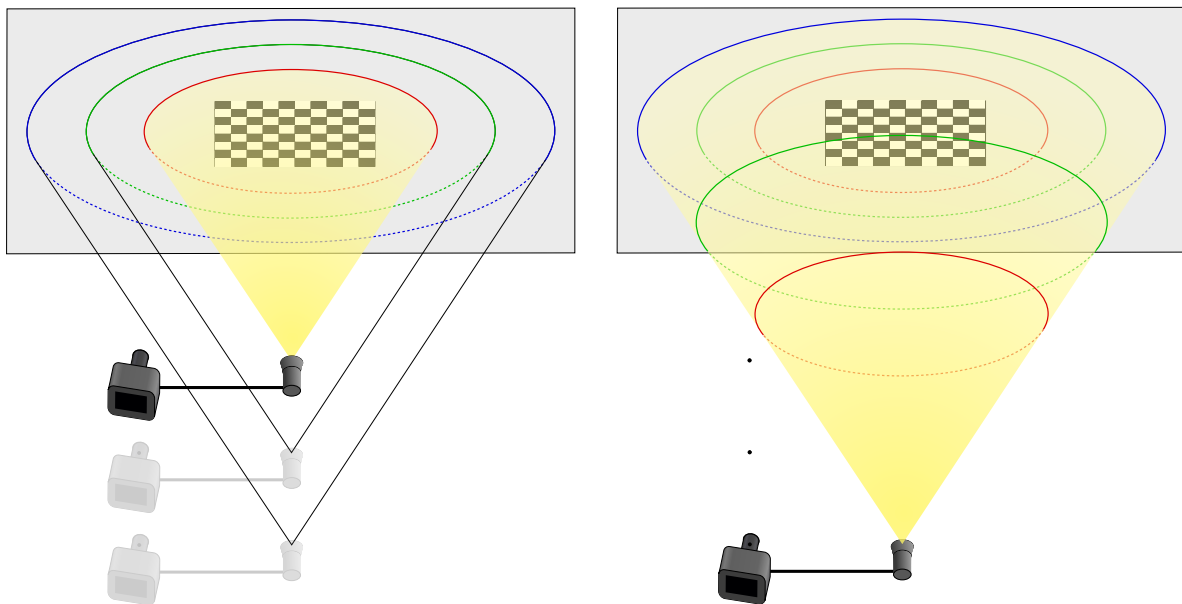


Figure 8. Method for obtaining several elliptical sections of the cone.

Consequently, if we have obtained n elliptical sections where m 3D points are extracted per section, we generate a cloud of $n \times m$ 3D points belonging to the cone.

3.2.2. Cone Estimation

The aim is now to estimate the cone that best approximates the 3D point cloud. This is done by geometric fitting, i.e. by minimising the orthogonal distances between the 3D points and the cone.

Let \mathbf{p} be the vector containing the six parameters of the cone, which are :

- the three coordinates of its vertex $\mathbf{O}_{\mathcal{P}}$,
- the two angles yaw and pitch in ZYX-Euler convention of its direction vector \mathbf{d} , the angle roll not being necessary since a cone has an axis of symmetry,
- its opening half-angle α .

To estimate the \mathbf{p} vector, we use the Levenberg-Marquardt (LM) iterative method [29]. For the first iteration, we need to choose an initial parameter vector \mathbf{p}_0 . This could be the null vector, for example. In practice, we will use measurements taken with a caliper for the various parameters of \mathbf{p}_0 .

At each iteration i , LM receives a solution vector \mathbf{p}_i and the function for calculating the vector of orthogonal distances to be minimised. Our function first calculates the rotation and translation of the cone relative to the camera from $\mathbf{O}_{\mathcal{D}}$ and the yaw and pitch angles of the vector \mathbf{p}_i . It then expresses the 3D points in the cone reference frame and calculates the orthogonal distance vector between the 3D points and the cone using the equation 4.

So from \mathbf{p}_i and this function, LM returns a new solution vector \mathbf{p}_{i+1} .

When the difference between the sum of the squared orthogonal distances obtained with \mathbf{p}_j and the calculus obtained with \mathbf{p}_{j+1} is less than an arbitrary threshold, this means that LM has reached convergence at iteration j . The best solution in the least squares sense is therefore the vector \mathbf{p}_j .

It is important to note that there are an infinite number of cones that share the same elliptical section. On the other hand, there is only one cone that passes through an elliptical section and through a point outside that section. Consequently, this minimisation can only work if we have at least two elliptical sections.

3.3. Intersection with a Ray

Now that we know how to determine the cone parameters using our calibration method, the next step is to be able to calculate the intersections between a camera ray and the cone. We are only interested in the intersections with the top of the cone.

A ray is a half-line that can be expressed parametrically using the following equation:

$$\mathbf{X}(t) = \mathbf{O} + t\mathbf{u} \quad (7)$$

with \mathbf{O} the starting point of the ray, \mathbf{u} its unit direction vector and t a positive real number such that $\mathbf{X}(0) = \mathbf{O}$. Note that we will only deal with cases where the point \mathbf{O} is outside the cone. This is because the starting point of the camera rays is the optical centre in our situation. The point \mathbf{O} can nevertheless be inside the lower part of the cone.

If $\mathbf{X}(t)$ is an intersection point, then it belongs to the cone. Therefore, to determine the intersections, we can simply calculate the values of t which cancel the general polynomial of the cone (see 5) after replacing \mathbf{X} by the expression for $\mathbf{X}(t)$:

$$\begin{aligned} \mathcal{Q}(\mathbf{X}(t)) &= (\mathbf{X}(t) - \mathbf{O}_{\mathcal{D}})^T \mathbf{Q}(\mathbf{X}(t) - \mathbf{O}_{\mathcal{D}}) = 0 \\ &\Leftrightarrow (\mathbf{O} + t\mathbf{u} - \mathbf{O}_{\mathcal{D}})^T \mathbf{Q}(\mathbf{O} + t\mathbf{u} - \mathbf{O}_{\mathcal{D}}) = 0 \\ &\Leftrightarrow t^2(\mathbf{u}^T \mathbf{Q} \mathbf{u}) + t(2\mathbf{u}^T \mathbf{Q}(\mathbf{O} - \mathbf{O}_{\mathcal{D}})) + (\mathbf{O} - \mathbf{O}_{\mathcal{D}})^T \mathbf{Q}(\mathbf{O} - \mathbf{O}_{\mathcal{D}}) = 0 \\ &\Leftrightarrow at^2 + bt + c = 0 \text{ with } \begin{cases} a = \mathbf{u}^T \mathbf{Q} \mathbf{u} \\ b = 2\mathbf{u}^T \mathbf{Q}(\mathbf{O} - \mathbf{O}_{\mathcal{D}}) \\ c = (\mathbf{O} - \mathbf{O}_{\mathcal{D}})^T \mathbf{Q}(\mathbf{O} - \mathbf{O}_{\mathcal{D}}) = \mathcal{Q}(\mathbf{O}) \end{cases} \end{aligned}$$

3.4. Projection of the Generatrices of the Cone in the Image

To understand the relationship between a point on the cone and its projection into the image, we need to look at the projections of the generatrices of the cone. Since the cone represents the projector, we will only consider the upper half of the cone, which means that the generatrices can be limited to the half-lines that start at the vertex $\mathbf{O}_{\mathcal{D}}$. Note that the study is limited to the case where the cone is oriented in the same direction as the camera, as required by our method. Geometrically, this condition is satisfied if the angle called μ between $\mathbf{z}_{\mathcal{C}}$ and $\mathbf{d} = \mathbf{z}_{\mathcal{D}}$ is in the interval $]\frac{\pi}{2} - \alpha; -\frac{\pi}{2} + \alpha[$ (see Figure 9). In practice, for the camera to be able to see the light projection, the angle μ will be close to 0.

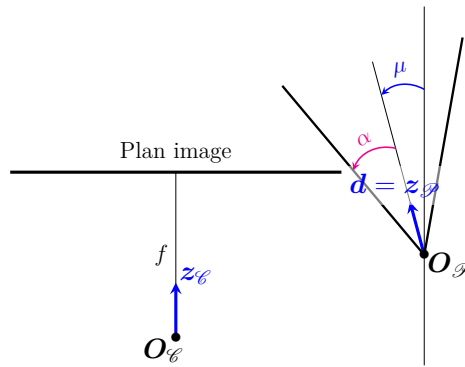


Figure 9. Representation of the camera / cone pair in an orthogonal projection 2D view where the projection axis is perpendicular to z_P and z_C . The relative orientation of the cone with the camera can be defined here by a single angle called μ . This is only true if we consider that the camera has an infinite field of view and is therefore symmetrical about its axis defined by z_C (the cone is basically symmetrical about its axis defined by $z_P = d$).

3.4.1. Calculating the Projections of the Generatrices of the Cone

Let us call Π_g the plane which passes through the optical centre O_C and through at least one generatrix of the cone. The intersection of this plane with the image plane is the line ℓ_g .

Let X' be a point in the image such that $X' = K\mathcal{C}\{u\}$ with u the direction vector of the camera ray associated with X' . If X' is the projection into the image of a 3D point belonging to a generatrix resulting from the intersection between the cone and the plane Π_g , then X' belongs to the line ℓ_g . However, if X' belongs to the line ℓ_g , it is not necessarily the projection of a point on the cone. To obtain the equation of ℓ_g , the simplest way is to select two points belonging to one of the generatrices (or the generatrix) through which the Π_g passes, then project them into the image to find the parameters of the line. The same result would be obtained by projecting two points of the plane Π_g which do not belong to the same camera radius. However, knowing the line ℓ_g is not enough to understand the projection of a generatrix.

Observation When Π_g Is Not Tangent to the Cone

When the plane Π_g is not tangent to the cone, it passes through two generatrices g_1 and g_2 , with g_1 chosen as the generatrix closest to the optical centre O_C (see Figure 10).

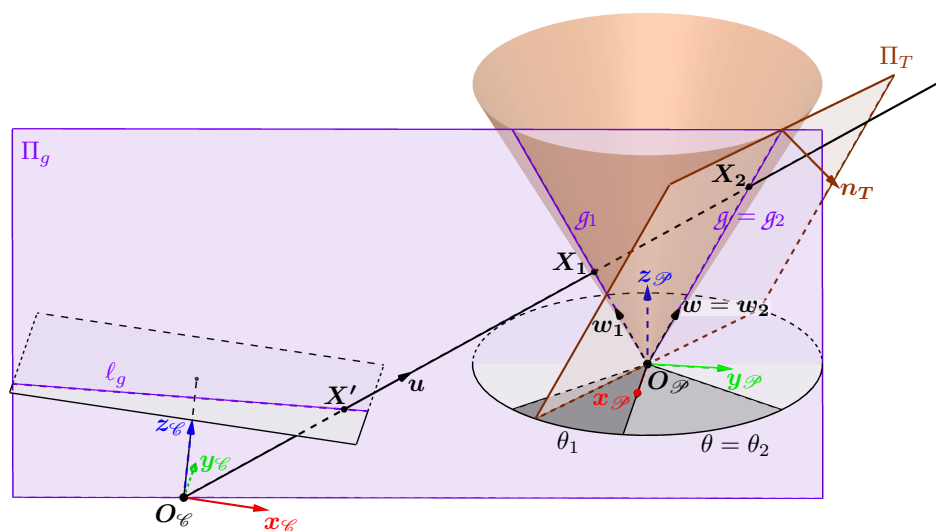


Figure 10. Representation of the plane Π_g when it is not tangent to the cone passing through the optical centre and the two generatrices g_1 and g_2 .

The direction vectors of these two generatrices are w_1 and w_2 such that :

$$w_{1\{\mathcal{P}\}} = \begin{bmatrix} \sin(\alpha) \cos(\theta_1) \\ \sin(\alpha) \sin(\theta_1) \\ \cos(\alpha) \end{bmatrix} \text{ et } w_{2\{\mathcal{P}\}} = \begin{bmatrix} \sin(\alpha) \cos(\theta_2) \\ \sin(\alpha) \sin(\theta_2) \\ \cos(\alpha) \end{bmatrix}$$

with θ_1 (respectively θ_2) the angle between the vector w_1 (respectively w_2) projected in the plane $x_{\mathcal{P}}y_{\mathcal{P}}$ and the vector $x_{\mathcal{P}}$ (see Figure 10). As the plane Π_g is chosen here as not tangent to the cone, we have $\theta_1 \neq \theta_2$. In this configuration, if X' is a point on the line ℓ_g then it can be both the projection of a point X_1 belonging to g_1 and of a point X_2 belonging to g_2 . The generatrix g_1 therefore contains the first intersections of the camera ray with the cone, while g_2 contains the second intersections.

Calculating the Two Special Generatrices When Π_g Is Tangent to the Cone

There are only two orientations for which the plane Π_g is tangent to the cone. When this happens, then $\theta_1 = \theta_2$ and the generatrices g_1 and g_2 are superimposed. The two possible cases are illustrated in Figure 11. The two angles θ_A and θ_B define the two generatrices g_A and g_B through which pass the two planes tangent to the cone Π_{g_A} and Π_{g_B} . It is important to note that the angles θ_A and θ_B depend solely on the parameters of the cone and its pose relative to the camera $T_{\{\mathcal{C} \rightarrow \mathcal{C}\}}$, which means that they are unique and fixed for a given configuration.

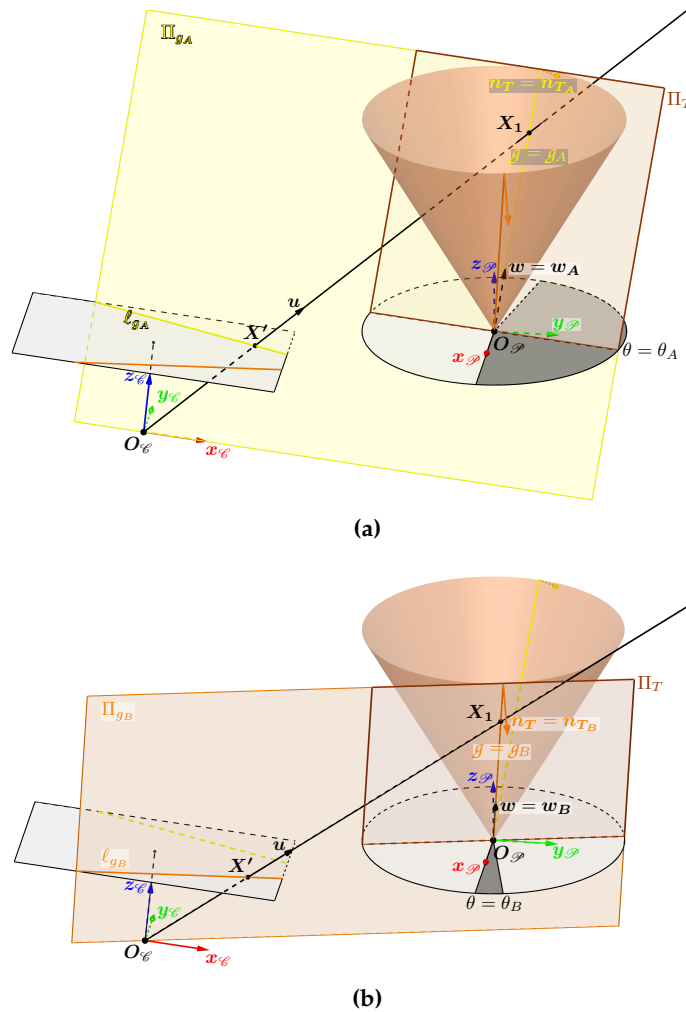


Figure 11. Representation of Π_{g_A} (a) and Π_{g_B} (b), the only two planes tangent to the cone passing through the optical centre.

To determine these two angles, we will use a plane tangent to the cone called Π_T which passes through a generatrix g whose direction vector is w defined by an angle θ . It is represented in Figure 10 passing through the generatrix g_2 purely for reasons of visibility. The orientation of the plane Π_T is characterised by the unit normal vector n_T such that :

$$n_{T\{\mathcal{P}\}} = \begin{bmatrix} \cos(\alpha) \cos(\theta) \\ \cos(\alpha) \sin(\theta) \\ -\sin(\alpha) \end{bmatrix} \quad (8)$$

We are therefore looking for the two solutions of θ so that the plane Π_T is superimposed on the plane Π_{g_A} or Π_{g_B} . This superposition only occurs when the plane Π_T passes through the optical centre $O_{\mathcal{C}}$ and therefore when the vector n_T is orthogonal to the vector $(O_{\mathcal{C}} - O_{\mathcal{P}})$. We are therefore looking for θ when :

$$\begin{aligned} O_{\mathcal{C}\{\mathcal{P}\}}^T n_{T\{\mathcal{P}\}} = 0 &\Leftrightarrow \begin{bmatrix} x_{O_{\mathcal{C}\{\mathcal{P}\}}} & y_{O_{\mathcal{C}\{\mathcal{P}\}}} & z_{O_{\mathcal{C}\{\mathcal{P}\}}} \end{bmatrix} \begin{bmatrix} \cos(\alpha) \cos(\theta) \\ \cos(\alpha) \sin(\theta) \\ -\sin(\alpha) \end{bmatrix} = 0 \\ &\Leftrightarrow x_{O_{\mathcal{C}\{\mathcal{P}\}}} \cos(\alpha) \cos(\theta) + y_{O_{\mathcal{C}\{\mathcal{P}\}}} \cos(\alpha) \sin(\theta) - z_{O_{\mathcal{C}\{\mathcal{P}\}}} \sin(\alpha) = 0 \end{aligned} \quad (9)$$

with $O_{\mathcal{C}\{\mathcal{P}\}}$ the optical centre expressed in the reference frame of the \mathcal{P} projector. To solve the equation 9 we pose :

$$s = \cos(\alpha) \begin{bmatrix} x_{O_{\mathcal{C}\{\mathcal{P}\}}} \\ y_{O_{\mathcal{C}\{\mathcal{P}\}}} \end{bmatrix} \quad \text{and} \quad t = \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix} \quad (10)$$

We can rewrite the equation 9 using the scalar product between the vectors s and t and the angles defined in Figure 12 :

$$s^T t = \overbrace{\cos(\alpha) \sqrt{x_{O_{\mathcal{C}\{\mathcal{P}\}}}^2 + y_{O_{\mathcal{C}\{\mathcal{P}\}}}^2}}^{\|s\| \|t\| = \|s\|} \cos(\gamma) = z_{O_{\mathcal{C}\{\mathcal{P}\}}} \sin(\alpha)$$

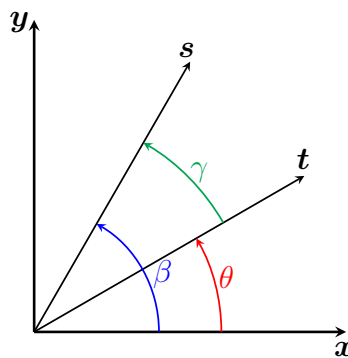


Figure 12. Visualisation of the vectors t and s and their angles.

The angle γ between the vectors s and t is therefore :

$$\gamma = \pm \arccos\left(\frac{z_{O_{\mathcal{C}\{\mathcal{P}\}}} \sin(\alpha)}{\cos(\alpha) \sqrt{x_{O_{\mathcal{C}\{\mathcal{P}\}}}^2 + y_{O_{\mathcal{C}\{\mathcal{P}\}}}^2}}\right) = \pm \arccos\left(\frac{z_{O_{\mathcal{C}\{\mathcal{P}\}}}}{\sqrt{x_{O_{\mathcal{C}\{\mathcal{P}\}}}^2 + y_{O_{\mathcal{C}\{\mathcal{P}\}}}^2}} \tan(\alpha)\right)$$

As for the angle β which defines the orientation of the vector \mathbf{s} , its expression is :

$$\beta = \text{atan2}(y_{O_{\mathcal{C}}\{\mathcal{P}\}}, x_{O_{\mathcal{C}}\{\mathcal{P}\}})$$

The solution to equation 9 is therefore :

$$\theta_{A,B} = \beta \pm \gamma = \text{atan2}(y_{O_{\mathcal{C}}\{\mathcal{P}\}}, x_{O_{\mathcal{C}}\{\mathcal{P}\}}) \pm \arccos\left(\frac{z_{O_{\mathcal{C}}\{\mathcal{P}\}}}{\sqrt{x_{O_{\mathcal{C}}\{\mathcal{P}\}}^2 + y_{O_{\mathcal{C}}\{\mathcal{P}\}}^2}} \tan(\alpha)\right) \quad (11)$$

Note that the two angles θ_A and θ_B only depend on the half-angle of the opening α of the cone and the translation between the cone and the camera (translation expressed in the equation by $O_{\mathcal{C}}\{\mathcal{P}\}$). From the expression of these two angles, we can obtain the expression of the two unit direction vectors \mathbf{w}_A and \mathbf{w}_B of the two special generatrices g_A and g_B . The projections of these two generatrices belong to the lines ℓ_{g_A} and ℓ_{g_B} which are the intersections of the image plane with the planes Π_{g_A} and Π_{g_B} .

Splitting the Cone into Two Areas Using the Two Special Generatrices

Using the generatrices g_A and g_B we can divide the cone into two surfaces as shown in Figure 13. The cyan surface named S_1 (resp. magenta surface named S_2) contains the set of generatrices containing all the first (resp. second) possible intersections between a camera beam and the cone, generatrices previously named g_1 (resp. g_2).

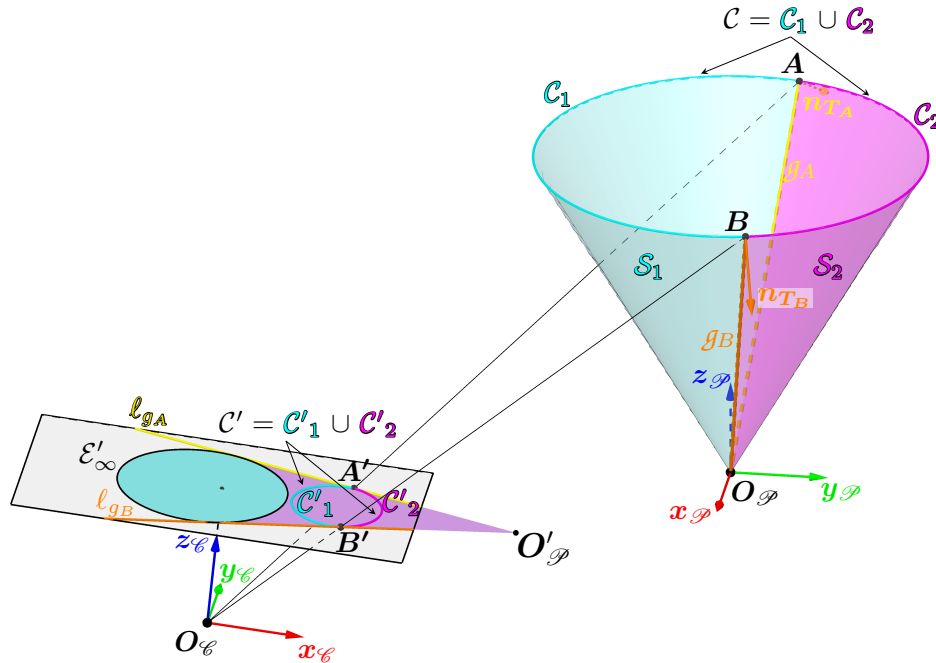


Figure 13. Representation in the image plane and in the cone of the areas containing the first intersections (in cyan) and the second intersections (in magenta).

In the image plane, this shows that we can delimit the areas where the camera rays can intersect the cone. The cyan area, surrounded by the ellipse \mathcal{E}'_{∞} , contains the projections of part of the set of 3D points belonging to the surface S_1 . The magenta area contains the projections of the other part of the set of 3D points belonging to the surface S_1 and contains the projections of the set of 3D points belonging to the surface S_2 . These areas are bounded by :

- the lines ℓ_{g_A} and ℓ_{g_B} .
- the ellipse \mathcal{E}'_{∞} which is the set of vanishing points of the generatrices. These points define a first bound on all the projections of the generatrices.

- the projection of the vertex $O_{\mathcal{P}}$ into the image plane, named $O'_{\mathcal{P}}$. It defines a second bound on all the projections of the generatrices.

In the configuration illustrated in Figure 13, each generatrix projection is therefore a segment whose limits are the vanishing point of the generatrix belonging to the ellipse \mathcal{E}'_{∞} and the point $O'_{\mathcal{P}}$.

However, there are other configurations where the projections of the generatrices onto the image plane are not segments but lines. Indeed, if $z_{O_{\mathcal{P}}\{\mathcal{C}\}} = 0$, then the line passing through $O_{\mathcal{P}}$ and $O_{\mathcal{C}}$ is parallel to the plane, which implies that the projection of the vertex $O_{\mathcal{P}}$ into the image plane is a point at infinity (or ideal point) and that the lines ℓ_{g_A} and ℓ_{g_B} are parallel.

The case where $z_{O_{\mathcal{P}}\{\mathcal{C}\}} < 0$ is illustrated in Figure 14 which shows the lower part of the cone behind the camera. This part is shown in brown and its projection in the image plane is also in brown. This projection is obviously fictitious, as this part cannot be seen by the camera even if it had an infinite field of view. The point $O'_{\mathcal{P}}$, also fictitious, is still the intersection of the lines ℓ_{g_A} and ℓ_{g_B} , but this time it is on the other side in comparison of the Figure 13.

As for the magenta and cyan areas in the image plane, they are still delimited by the lines ℓ_{g_A} and ℓ_{g_B} and the ellipse \mathcal{E}'_{∞} but extend towards infinity on the side where the lines ℓ_{g_A} and ℓ_{g_B} diverge.

So if $z_{O_{\mathcal{P}}\{\mathcal{C}\}} \leq 0$ then each generatrix projection is a half-line whose only boundary is its vanishing point belonging to the ellipse \mathcal{E}'_{∞} . However, as we are working with real cameras and not with the image plane, which is infinite, each generatrix will have a segment as its projection into the image.

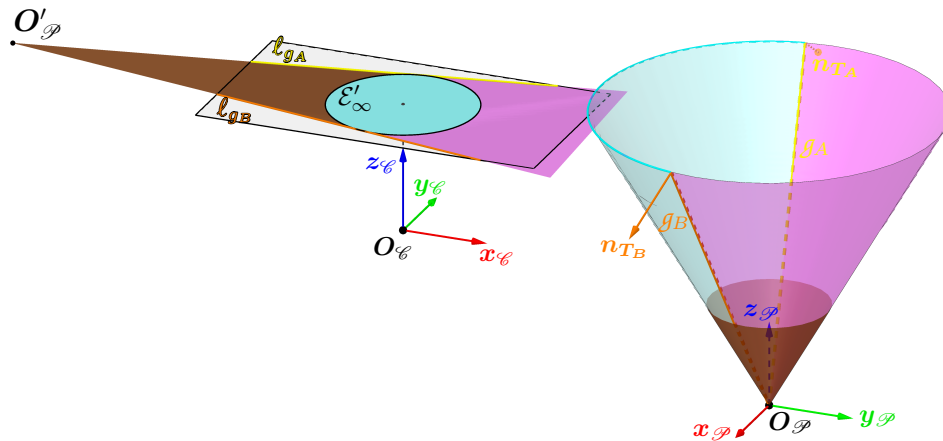


Figure 14. Representation in the image plane and in the cone of the areas containing the first intersections (in cyan) and the second intersections (in magenta) in the case where part of the cone (in brown) is behind the camera.

This division of the cone into two surfaces leads to several results:

- Any ray associated with a point in the cyan area has as its unique intersection with the cone a 3D point belonging to the surface \mathcal{S}_1 .
- Any ray associated with a point in the magenta area has two intersections with the cone, the first of which belongs to the surface \mathcal{S}_1 and the second to the surface \mathcal{S}_2 . These two intersections are superimposed if the point in the magenta area belongs to the line ℓ_{g_A} or to the line ℓ_{g_B} .
- Any ray associated with a point outside the magenta area and the cyan area does not have any intersection with the cone.

3.5. Intersection Selection and Triangulation

We now have everything we need to develop a method for determining which intersection to choose when the ray associated with a pixel in the contour has two intersections with the cone. To better understand the various stages of the procedure, let's take as an example the 3D contour of light whose associated 3D curve is circular and results from the intersection between a plane and the cone.

The example of the circular contour is illustrated in Figure 13 which contains the following elements:

- \mathcal{C}' is the closed 2D curve corresponding to the contour in the image with X' a 2D point on the curve.
- \mathcal{C} is our circular 3D curve corresponding to the contour of the light in the scene, with X a 3D point on the curve which has X' as its projection. The point X therefore corresponds to the correct intersection between the cone and the camera ray associated with X' .
- The areas containing the first and second intersections with the cone are delimited by the two generatrices g_A and g_B which divide the cone into two surfaces \mathcal{S}_1 and \mathcal{S}_2 . They also divide \mathcal{C} into two curves, one in cyan called \mathcal{C}_1 and the other in magenta called \mathcal{C}_2 .
- A and B are respectively the two intersections of g_A and g_B with the curve \mathcal{C} and are thus the only two 3D points common to the curves \mathcal{C}_1 and \mathcal{C}_2 .
- The cyan curve named \mathcal{C}'_1 and the magenta curve named \mathcal{C}'_2 are respectively the projections of the curves \mathcal{C}_1 and \mathcal{C}_2 .
- A' and B' are the projections of A and B and are therefore the only two points common to the curves \mathcal{C}'_1 and \mathcal{C}'_2 . They belong to both the curve \mathcal{C}' and the lines ℓ_{g_A} and ℓ_{g_B} . So, in theory, they correspond to the two unique points of tangency of the lines ℓ_{g_A} and ℓ_{g_B} with \mathcal{C}' .

So, if we get the points of tangency A' and B' , we will be able to divide the 2D curve \mathcal{C}' in two via the line $(A'B')$. However, in practice :

- the curve \mathcal{C}' is discrete,
- the light contours are extracted in a perfectible way,
- the camera calibration and the cone estimation have uncertainties, so the lines ℓ_{g_A} and ℓ_{g_B} also have uncertainties.

All this means that the lines ℓ_{g_A} and ℓ_{g_B} are not exactly tangent to the curve \mathcal{C}' . The method for determining the points A' and B' must therefore be adapted to this constraint.

Once these two points have been obtained and the curve \mathcal{C}' has been divided in two, it remains to determine which of the two curves is \mathcal{C}'_1 or \mathcal{C}'_2 . To do this, we need to look at the relative position of the vertex $O_{\mathcal{P}}$ in relation to the camera. Figure 15 shows that the vertex $O_{\mathcal{P}}$ and the curve \mathcal{C}'_2 lie in the same half-space bounded by the plane passing through $O_{\mathcal{C}}$ and the line $(A'B')$. This property is always true whatever the pose of the cone.

In the image plane, this property can be expressed by defining a point p_{ref} such that p_{ref} and the curve \mathcal{C}'_2 lie in the same half-plane bounded by the line $(A'B')$.

There are an infinite number of solutions for p_{ref} in the image plane which satisfy this condition. We have therefore forced p_{ref} to belong to the circumscribed circle of the rectangle delimiting the image, as can be seen in Figure 15. The segment between p_{ref} and the centre of the image forms an angle with the horizontal axis of the image called ν . This gives :

$$p_{ref} = \sqrt{w^2 + h^2} \begin{bmatrix} \cos(\nu) \\ \sin(\nu) \end{bmatrix} \quad (12)$$

with w and h respectively the width and height of the image in pixels. For the condition to be true, one possible solution for ν is to indicate the direction of $O_{\mathcal{P}}$ relative to the optical centre $O_{\mathcal{C}}$. To obtain this angle, we orthogonally projected $O_{\mathcal{P}}$ onto the plane normal to the camera axis passing through $O_{\mathcal{C}}$, which gives the point $H_{\mathcal{P}}$ illustrated in Figure 15. This gives ν which is the angle formed by the vectors $(H_{\mathcal{P}} - O_{\mathcal{C}})$ and $x_{\mathcal{C}}$:

$$(H_{\mathcal{P}} - O_{\mathcal{C}}) = \begin{bmatrix} x_{O_{\mathcal{P}}\{\mathcal{C}\}} \\ y_{O_{\mathcal{P}}\{\mathcal{C}\}} \\ 0 \end{bmatrix} \text{ donc } \nu = \arctan\left(\frac{y_{O_{\mathcal{P}}\{\mathcal{C}\}}}{x_{O_{\mathcal{P}}\{\mathcal{C}\}}}\right) \quad (13)$$

Calculating this angle ν and the point p_{ref} allows us to deduce which of the two curves is C'_2 .

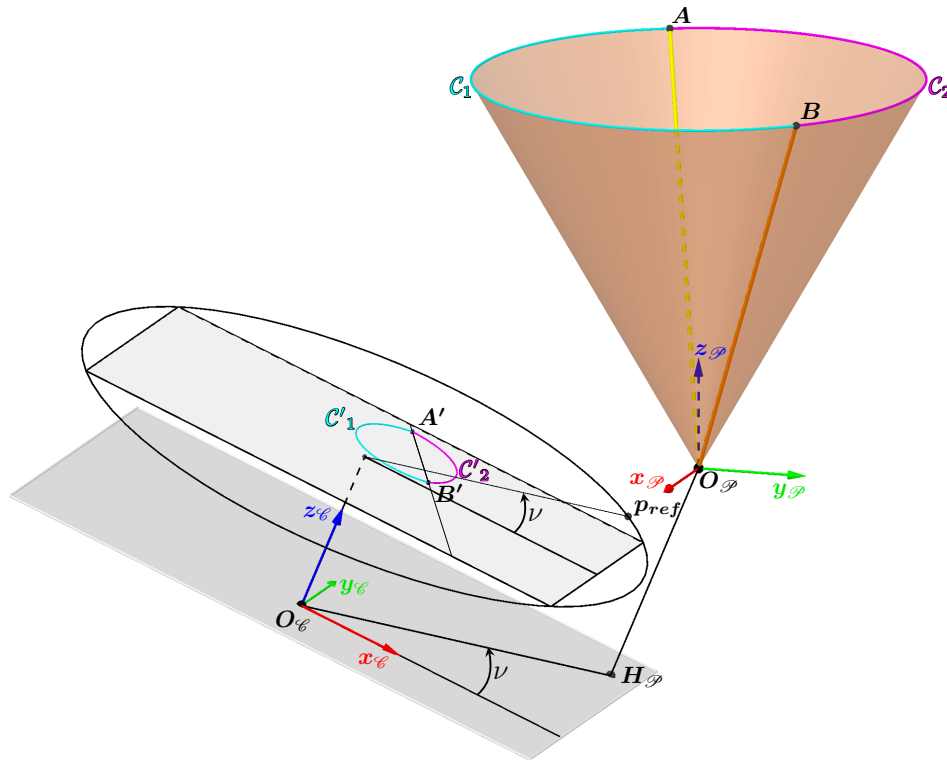


Figure 15. Illustrates how to obtain p_{ref} , our reference point in the image plane which indicates the relative position of the cone and which is used to obtain the curves C'_1 and C'_2 .

Once the curves C'_1 and C'_2 have been obtained, it is then possible to obtain the 3D contour. Indeed, if a point X' belongs to C'_1 , then the first intersection of the camera ray associated with it must be chosen, and if it belongs to C'_2 , the second must be chosen. This is because any point X' of C'_1 is the image of a 3D point X of C_1 , and any point X' of C'_2 is the image of a 3D point X of C_2 .

3.6. Test of the Method in Simulation

Contour Simulation

To illustrate the different stages of the method, we will use a gallery model. Figure 16 shows the camera represented by a pyramid and the projector represented by a cone positioned inside our model. In our example, the axis of the cone is parallel to the axis of the camera. The vertex O_P is positioned above and to the right of the camera; it belongs to the plane which passes through the optical centre O_C and which is orthogonal to the camera axis. Consequently, the projection of O_P in the image plane is a point at infinity (an ideal point) and the projections of the generatrices all belong to lines parallel to each other.

The 3D curve C is the intersection between the cone and the model. The set of 3D points on this curve is thus obtained from the intersections of the generatrices of the cone with the model, and they are shown in black in Figure 16. The number of 3D points is equal to the number of generatrices chosen to represent the cone. These 3D points are then projected onto the image to obtain a set of 2D points belonging to the curve $curveProjCone$. These 2D points are shown in black in Figure 16.

The idea now is to obtain the 3D points of the curve C from the 2D points of the curve C' . We will assume that the parameters of the cone and the camera are known.



Figure 16. The camera (represented by a pyramid) and the projector (represented by a cone) arranged inside the model of our gallery to simulate a contour of light in an image. This contour is obtained by projecting the intersections between the generatrices of the cone and the model into the image.

Calculating the Generatrices and the Lines Containing Their Projection

The first step is to calculate the direction vectors w_A and w_B of the generatrices g_A and g_B . To do this, we need to calculate the angles θ_A and θ_B using the equation 11.

Once the direction vectors have been obtained, we select two 3D points on g_A and g_B , project them into the image and compute the parameters of the lines ℓ_{g_A} and ℓ_{g_B} . The two lines are defined by their respective unit direction vectors u_A and u_B and by A'_1 and B'_1 , two points belonging respectively to the two lines.

Determining the Points A' and B'

We wish to obtain the points A' and B' which are theoretically the points of tangency of the lines ℓ_{g_A} and ℓ_{g_B} with the curve C' . We will only present the approach for the point A' since it will be the same for B' .

A first solution would be to say that A' is the point belonging to the line ℓ_{g_A} closest to the curve C' . However, we have chosen a slightly more generic approach, the result of which is illustrated in Figure 17. It consists of considering A' as the point belonging to the line ℓ_{g_A} which minimises the sum of the squared distances between itself and the $x\%$ of the points on the curve C' closest to the line ℓ_{g_A} . Let t be the parameter of the line. We are therefore looking for t such that :

$$t = \underset{t}{\operatorname{argmin}} \left\{ \sum_{i=1}^n \frac{(A' - X'_i)^T (A' - X'_i)}{f_i(X'_i)} \right\} \text{ with } A' = A'_1 + t u_A \quad (14)$$

Where :

- X'_i are the $x\%$ of the points on the curve C' closest to the line ℓ_{g_A} .
- $f_i : \mathbb{R}^3 \rightarrow \mathbb{R}^+$ is a function whose aim is to reduce the impact of the points X'_i far from the line ℓ_{g_A} in minimisation.

Figure 17 shows the points A' and B' obtained, with 40% of the points closest to ℓ_{g_A} in yellow and 40% of the points closest to ℓ_{g_B} in orange. Developing the sum to be minimised in 14 gives us :

$$\begin{aligned}
\sum_{i=1}^n \frac{(A' - X'_i)^T (A' - X'_i)}{\hat{h}(X'_i)} &= \sum_{i=1}^n \frac{(A'_1 + tu_A - X'_i)^T (A'_1 + tu_A - X'_i)}{\hat{h}(X'_i)} \\
&= \sum_{i=1}^n \frac{(tu_A + v_i)^T (tu_A + v_i)}{\hat{h}(X'_i)} \text{ with } v_i = A'_1 - X'_i \\
&= \sum_{i=1}^n \frac{(t^2 u_A^T u_A + 2tu_A^T v_i + v_i^T v_i)}{\hat{h}(X'_i)} \\
&= t^2 u_A^T u_A \sum_{i=1}^n \frac{1}{\hat{h}(X'_i)} + t2 \sum_{i=1}^n \frac{u_A^T v_i}{\hat{h}(X'_i)} + \sum_{i=1}^n \frac{v_i^T v_i}{\hat{h}(X'_i)} \\
&= at^2 + 2bt + c
\end{aligned}$$

avec

$$\begin{cases} a = u_A^T u_A \sum_{i=1}^n \frac{1}{\hat{h}(X'_i)} \\ b = \sum_{i=1}^n \frac{u_A^T v_i}{\hat{h}(X'_i)} \\ c = \sum_{i=1}^n \frac{v_i^T v_i}{\hat{h}(X'_i)} \end{cases}$$

Since a must be positive, the polynomial $at^2 + 2bt + c$ has a minimum when its derivative cancels, i.e. when :

$$2at + 2b = 0 \Rightarrow t = -\frac{b}{a}$$

This gives :

$$A' = A'_1 - \frac{b}{a} u_A \quad (15)$$

Using the same procedure, we can obtain the point B' . This produces the line $(A'B')$ (see Figure 17) which will be used to divide the curve C' in two.

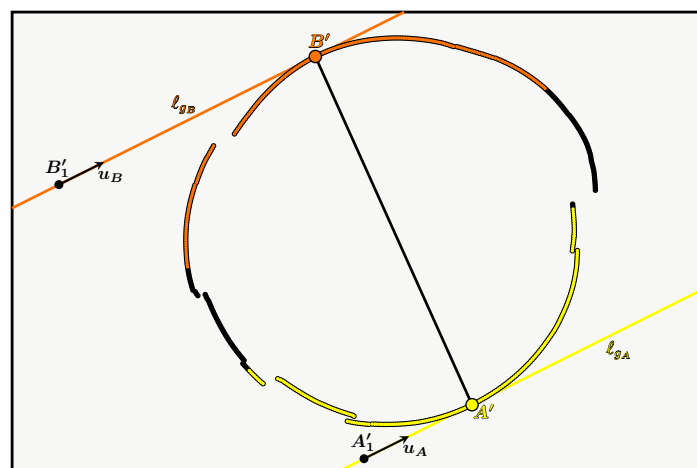


Figure 17. Determining the points A' and B' for our previous example contour.

Separation of the Curve C' for Intersection Selection

The points A' and B' are determined in a perfectible way in reality. We have therefore defined a minimum distance around the line $(A'B')$ called e which the points of the curve C' must respect. Points whose distance orthogonal to the line is less than e are considered indeterminate and will therefore not be taken into account in the 3D reconstruction of the contour. In practice, the value of e will depend on the quality of the measurements taken.

This separation of the C' curve into two curves is illustrated in Figure 18 with C'_1 in cyan and C'_2 in magenta.

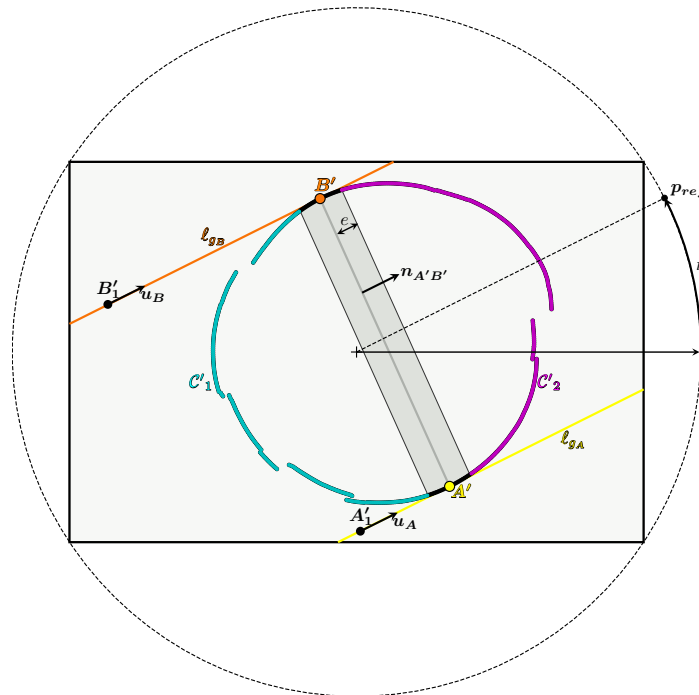


Figure 18. Determining the curves C'_1 and C'_2 for our previous example contour.

To obtain C'_1 or C'_2 , we had to use the point p_{ref} (see equation 12) obtained via the angle ν (see equation 13) and apply the following two conditions:

$$\begin{cases} X' \in C'_2 \text{ if } \text{sign}\left(n_{A'B'}^T(X' - A')\right) = \text{sign}\left(n_{A'B'}^T(p_{ref} - A')\right) \\ X' \in C'_1 \text{ if } \text{sign}\left(n_{A'B'}^T(X' - A')\right) = -\text{sign}\left(n_{A'B'}^T(p_{ref} - A')\right) \end{cases}$$

with $n_{A'B'}$, the normal vector to the line $(A'B')$.

3D reconstruction

We now know which points X' on the curve C' belong to the curve C'_1 or the curve C'_2 . We can therefore apply what was presented in the sections 3.3 and 3.5 to reconstruct these points in 3D.

Figure 19 takes the 3D scene from Figure 16 and adds the reconstructed 3D points. The calculated differences between these reconstructed 3D points and the initial 3D points of our simulation (obtained via the intersection between the generatrices of the cone and the gallery model) are below 10^{-8}m , which is very small. These deviations are due to numerical errors in the calculations at the various stages. This proves that these reconstructed 3D points are the same as the initial 3D points and that our method therefore works. We can now test the method with real data.

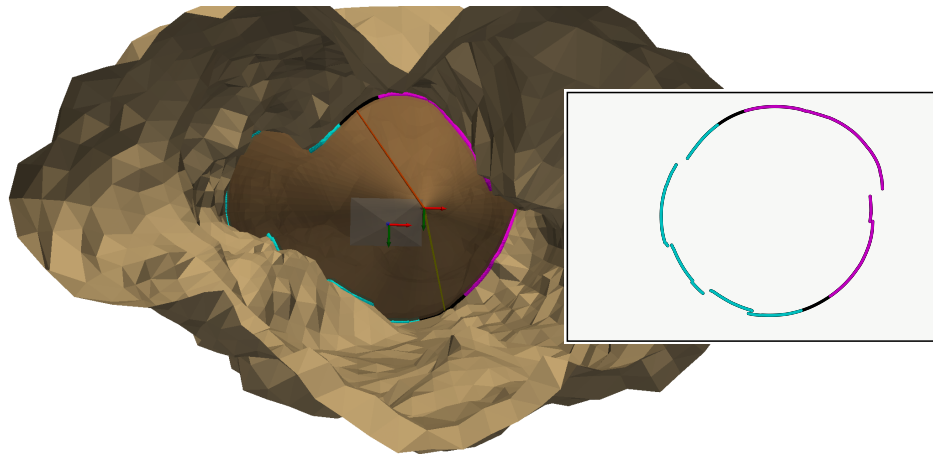


Figure 19. The reconstructed 3D points added to the 3D scene of Figure 19 with the first intersections in cyan and the second intersections in magenta.

4. Experiments in a Waterless Aqueduct

The local heritage offers us access, between two of our laboratory buildings, to the Saint-Clément aqueduct (more commonly known as the Arceaux aqueduct) built in the 18th century. It is no longer in use, but in dry periods it is an ideal experimental platform. The gallery is accessible via trapdoors, which once closed plunge you into total darkness. This confined space recreates experimental conditions similar to those in a karstic environment, but in a dry environment whose dimensions are easy to measure. Indeed, the aqueduct is simply a long corridor whose left and right walls are parallel and separated by a width of 62cm, which we measured with a laser rangefinder.

4.1. Our System

The camera used is the Nikon D7000 reflex and we set it to a focal length of 18 mm and a resolution of 2464×1632 .

The projector is the DIVEPRO M35 dive light with an aperture angle of 145° in air and 90° in water. As the aperture angle is too large to capture the full projection of light in our experiment, a custom-made plastic cylinder (obtained from the 3D printer) was added to reduce it, as can be seen in Figure 21. Its internal diameter is the same as the diameter of the projector, i.e. $d = 53,8$ mm. The cylinder protrudes from the projector by $l = 93$ mm (see Figure 20). We can therefore calculate the half-angle of aperture α as a function of the lengths l and d :

$$\alpha = \arctan\left(\frac{d/2}{l}\right) = \arctan\left(\frac{26,9}{93}\right) = 16,13^\circ$$

We will use this value to check the order of magnitude of future α values when we estimate the cone.

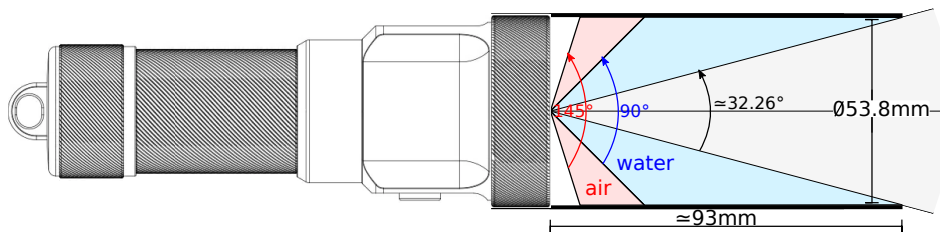


Figure 20. The diving lamp and its angle of aperture.

The camera and projector are placed on a rigid support as shown in Figure 21. They had to be fixed to the support because the slightest rotation of one in relation to the other alters the transformation between the camera and the projector, which is estimated during the projector calibration.

To get an idea of the magnitude of this transformation, we used a caliper to measure the translations between the camera and the projector along the camera's x , y and z axes. The rotation between the two is supposed to be small because we have chosen an identical orientation for both. Here is a summary of the various measurements of the projector parameters (lengths in m):

$$\left\{ \begin{array}{l} \alpha \approx 16,13^\circ \\ \mathbf{O}_{\mathcal{P}\{\mathcal{C}\}} \approx [0,2 \quad 0 \quad -0,07]^T \\ \|\mathbf{O}_{\mathcal{P}\{\mathcal{C}\}}\| \approx 0,212m \\ \mathbf{d}_{\{\mathcal{C}\}} \approx [0 \quad 0 \quad 1]^T \end{array} \right.$$

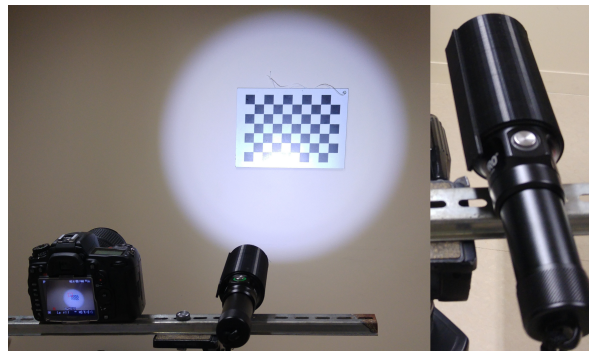


Figure 21. The system used, consisting of a camera and a conical-shaped projector to which a black cylinder has been added.

4.2. Camera Calibration Results

To calibrate our camera, we use the method of [30] with the help of eight images of a flat chessboard test pattern measuring 7×10 squares and 36mm wide (see Figure 22). In each of the eight images, the position of the test pattern relative to the camera is different, as can be seen in Figure 23, which is a 3D representation of the test pattern in eight different positions. Here are the camera parameters estimated via calibration:

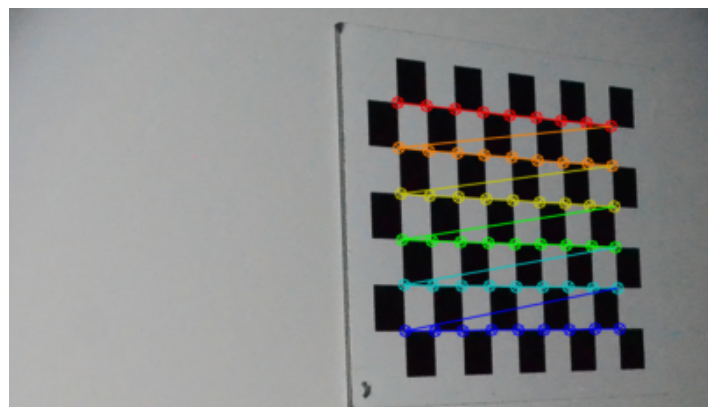


Figure 22. Image of one of the eight chessboard patterns taken at different poses for camera calibration using Zhang's method.

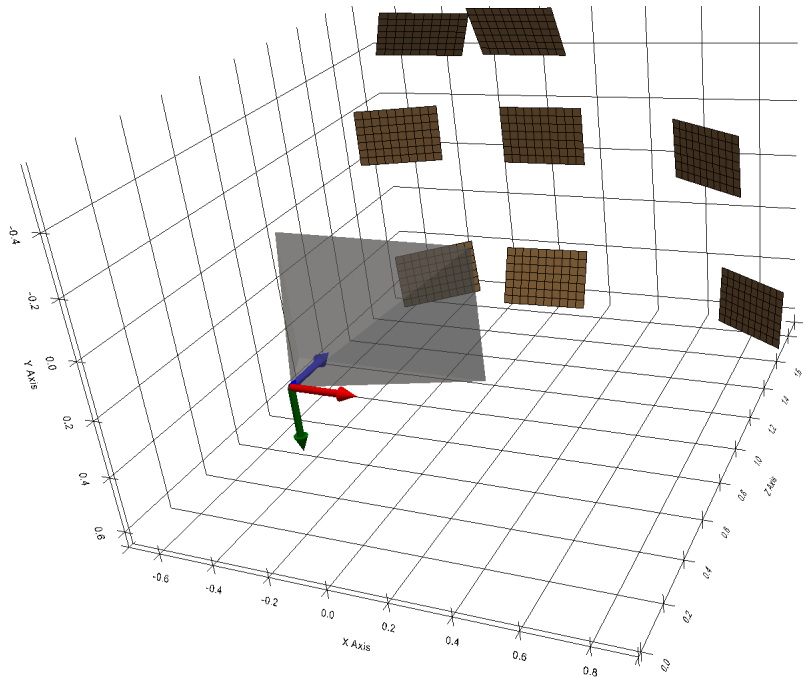


Figure 23. 3D view of the chessboard in its eight poses in relation to the camera.

- the intrinsic matrix K obtained is :

$$K = \begin{bmatrix} 1908,56 & 0,0 & 1227,53 \\ 0,0 & 1909,94 & 832,48 \\ 0,0 & 0,0 & 1,0 \end{bmatrix}$$

- the resulting focal length is 18.2 mm,

It can be seen that the orders of magnitude of the various camera parameters are consistent with the manufacturer's parameters. In addition, the average reprojection errors are of the order of 1/10 of a pixel. This level of error is quite satisfactory for all eight shots, each containing 56 calibration points.

4.3. Projector Calibration Results

To calibrate the projector, i.e. estimate the parameters of the cone, we used a white wall on which a chessboard pattern was hung. The first step is to obtain 3D points belonging to the cone by detecting the contour of the light projected onto the wall, as explained in the section entitled 3.2. To maximise the contrast of the light projected onto the white wall, the only light source in the room is the projector.

For this calibration, we captured five images of light projected onto the wall with the axis of the cone almost orthogonal to the wall (see Figure 24). The shape of the projected light therefore resembles a circle.

Each detected contour is transformed into an ellipse, shown in blue in Figure 24. To obtain the ellipses we use the method detailed in [28] but iteratively with the RANSAC algorithm [31] to eliminate certain outlying contour points. The contour points eliminated by RANSAC are shown in red in Figure 24 while the points used to obtain the ellipses are in green.

Using the pattern detected in each image, we obtain the equations for the five planes. From these five planes and the five ellipses obtained, the five elliptical sections are calculated. They are represented in 3D in Figure 25 with the camera and the test pattern.

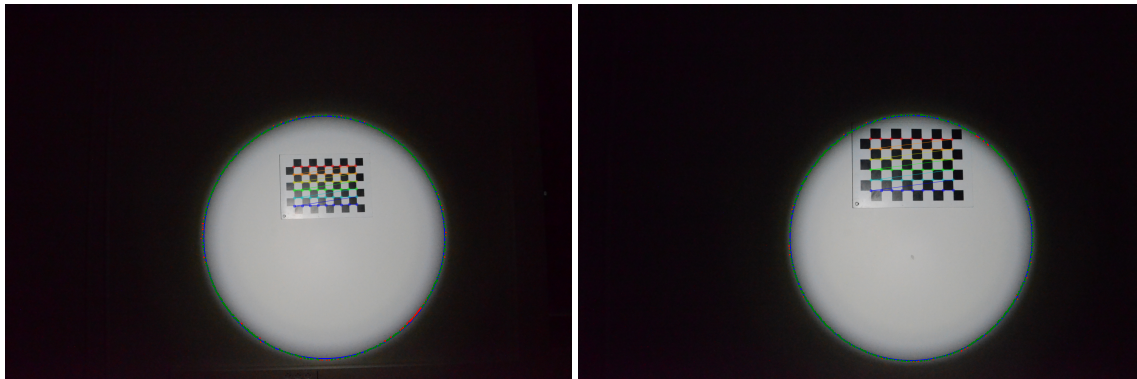


Figure 24. 2 images of light projected onto the wall from the five where the axis of the projector is almost orthogonal to the wall (images 1 and 5).

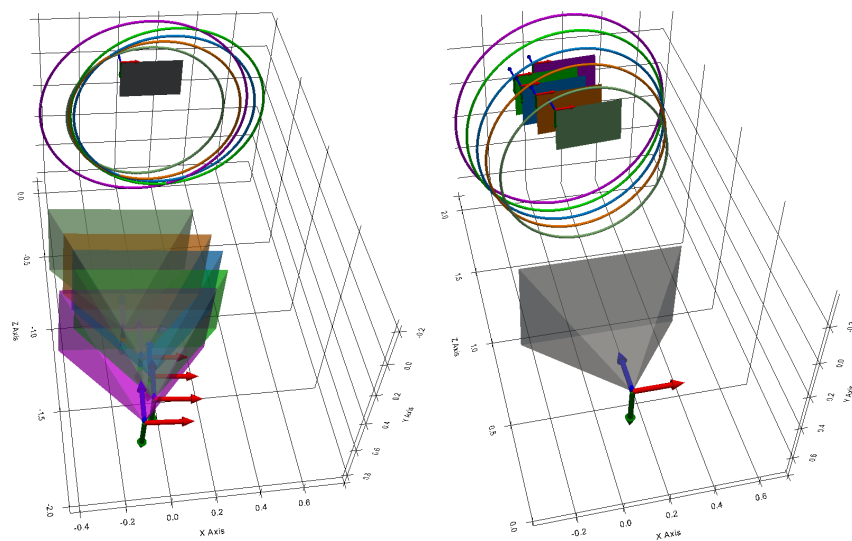


Figure 25. 3D view of the elliptical sections, the camera and the chessboard pattern. In the image on the left, the elements are expressed in the pattern frame. In the image on the right, the elements are expressed in the camera frame.

The image on the left shows all the elements expressed in the chessboard pattern frame, so you can see the different camera positions used to capture the five images.

The image on the right shows all the elements expressed in the camera frame. It is in this frame of reference that the elliptical sections must be expressed in order to estimate the cone.

We extract 3D points expressed in the camera frame of reference for each elliptical section. In our case, we recovered 200 points per section, making a total of 1000 3D points. Note that these 3D points are at a distance from the camera frame of between 1,50m and 2,17m.

Now that we have obtained 1000 3D points that are supposed to belong to the cone, the next step is to define an initial cone for our iterative minimisation algorithm. This is shown in red in Figure 26. We chose an initial cone that is far from the solution for greater visibility in the figure and also to prove that the method is capable of converging even when the initial cone is far from the solution. Obviously, this is an experimental result and not a mathematical proof. This optimisation problem is probably non-convex. Ideally, therefore, an initial cone close to the solution should be chosen using the aperture angle previously calculated and the relative pose measurement between the projector and the cone.

We therefore apply our algorithm and obtain the optimised cone shown in grey. Its estimated parameters (lengths in metres) are :

- $\alpha = 14,79^\circ$

- $O_{\mathcal{P}\{\mathcal{E}\}} = [0,1956 \quad 0,0092 \quad -0,0759]^T$
- $\|O_{\mathcal{P}\{\mathcal{E}\}}\| = 0,210m$
- $d_{\{\mathcal{E}\}} = [-0,013 \quad 0,080 \quad 0,997]^T$

To ensure the validity of the cone obtained, we begin by comparing the estimated parameters with the previously measured parameters. The Table 1 compares the estimated opening angle and the estimated gap with the measurements. As our measurements can be improved, this comparison at least proves that the order of magnitude of the estimated parameters is consistent.

Table 1. Comparison of estimated and measured cone parameters.

	Estimated values	Measured values	Relative deviation from measurement
α	14,79°	16,13°	1,34° (-8%)
$\ O_{\mathcal{P}\{\mathcal{E}\}}\ $	0,210m	0,212m	-0,002m (-0.9%)

Another point to check is the minimisation error for estimating the cone. In our case, this error is directly linked to the orthogonal distances between the 3D points of the elliptical sections and the cone, since it is the root mean square (RMS) of these distances that we are trying to minimise during estimation. In Figure 26, the colour of each 3D point is defined by its orthogonal distance from the cone via the colormap Jet (the colormap bar is shown in the figure) and the following function :

$$f : \mathbb{R}^+ \rightarrow [0,1]$$

$$d \mapsto \frac{d}{d + 0.1}$$

with d the orthogonal distance. m is chosen as the median of all these distances (here $m = 1,4mm$).

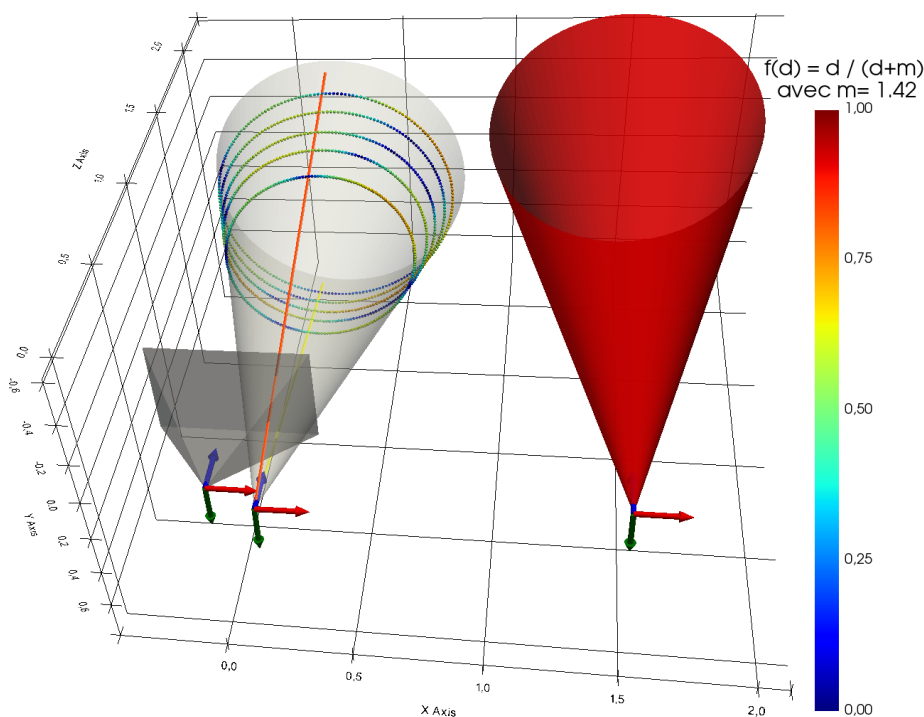


Figure 26. Representation of the estimated cone in relation to elliptical sections where the colour of the points depends on the distance orthogonal to the cone.

The statistics for these distances can be found in the Table 2. In particular, we have a maximum distance of 4,2mm and an overall mean of 1,5mm. The overall RMS is 1,9mm which corresponds to approximately 0,13% of 1,50m which is the smallest distance between a 3D point of an elliptical section and the camera frame.

In view of the analysis of the orthogonal distances, the minimisation is satisfactory, confirming once again that the estimated parameters of the cone are at least consistent with their true values.

Table 2. Statistics of the orthogonal distances in mm between the 3D points of each elliptical section and the estimated cone.

Dist. (mm)	Max.	Min.	Med.	Moy.	RMS
Section 0	4,2	0,0	1,4	1,5	2,0
Section 1	3,3	0,0	1,5	1,5	1,8
Section 2	4,1	0,0	1,4	1,6	2,0
Section 3	3,4	0,0	1,6	1,6	1,8
Section 4	3,6	0,0	1,3	1,5	1,8
All	4,2	0,0	1,4	1,5	1,9

Table 3. Distance statistics in mm between the 3D points of each elliptical section and the reconstructed 3D points.

Dist. (mm)	Max.	Min.	Med.	Moy.	RMS
Section 0	94,0	0,4	28,5	27,7	32,6
Section 1	149,6	0,0	23,6	30,9	42,8
Section 2	226,1	0,1	18,6	33,6	51,3
Section 3	179,3	0,1	18,6	30,0	44,7
Section 4	114,3	0,1	19,8	23,0	29,1
All	226,1	0,0	20,6	29,0	40,9

4.4. 3D Results

The shape of our aqueduct is a long narrow corridor, which makes it possible to fully distinguish the light projection in our images. We captured six images taken at different distances from a chessboard pattern placed in the aqueduct. For each image, the light contours were checked manually, as we wanted to isolate the triangulation method from the contour detection method, which is still in need of improvement. For each contour extracted, we manually delimited part of the contour points on the left wall and another part on the right wall. These delimitations can be seen in Figure 27. It shows an example of an extracted contour in two images, with contour points on the left wall in red, on the right wall in blue and the rest of the contour points in green.

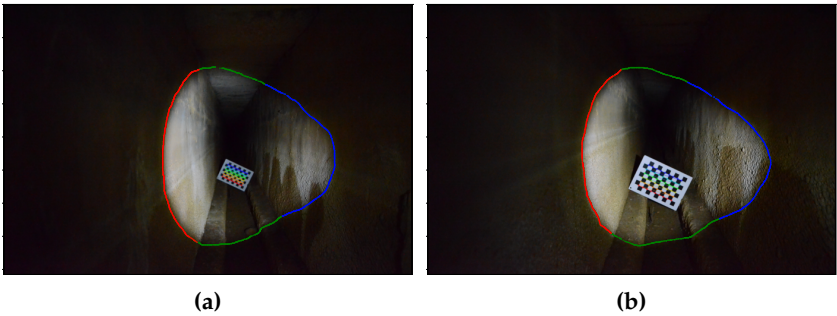


Figure 27. 2 images in the aqueduct with light contours.

Only the contours of the left and right walls will be used for triangulation, as this will allow us to check that the 3D points obtained respect the dimensions of the aqueduct corridor. To obtain these 3D points, it is necessary, as usual, to define which intersection is relevant for each radius associated with the contour points. In our situation, we can see that each contour point on the left wall implies a first intersection and each contour point on the right wall a second intersection. Figure 28 illustrates this for the two images in Figure 27.

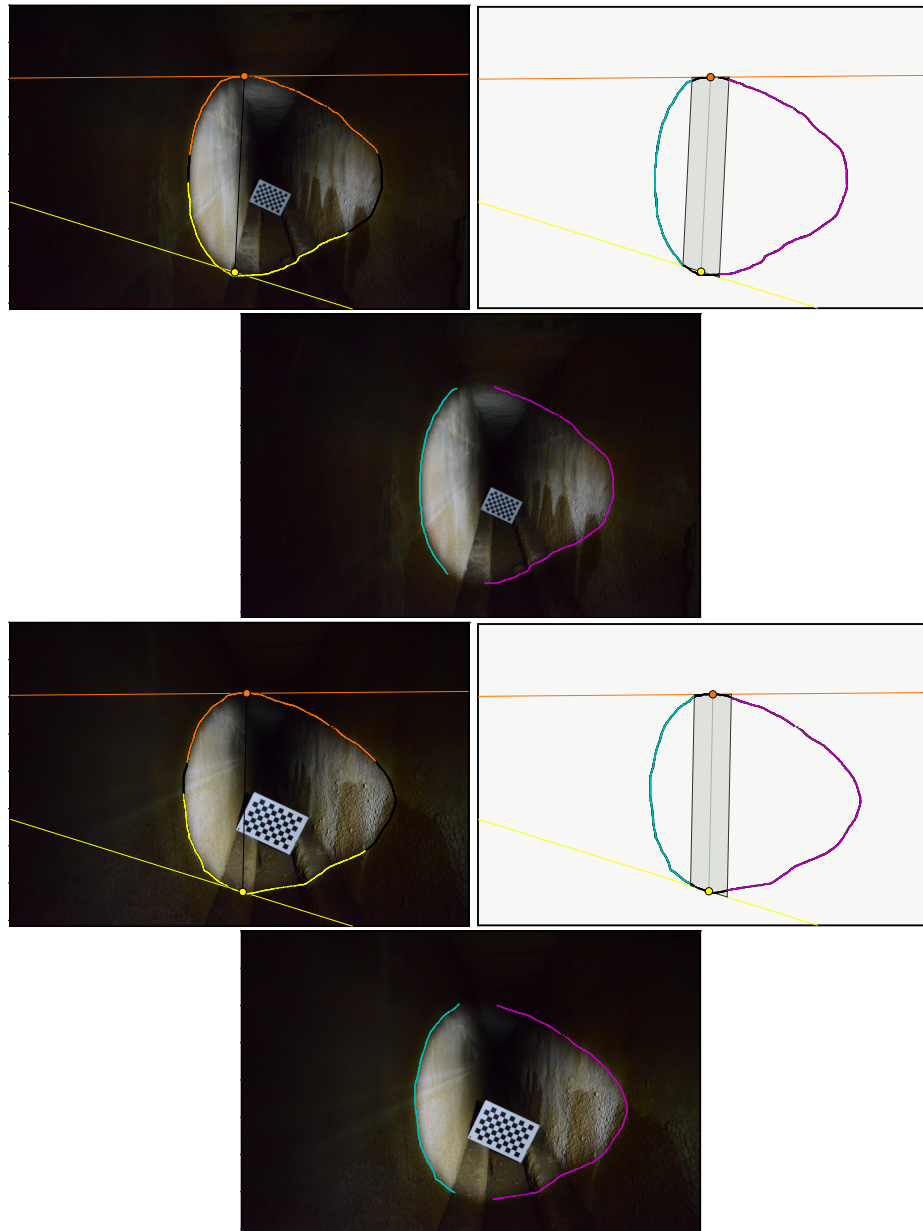


Figure 28. Intersection selection.

Figure 29 shows the result of the reconstruction of the contours of the six images. The shape of the 3D reconstructions of the contours is close to a hyperbola, which is consistent since the intersection between a plane and a cone in this configuration is supposed to be a hyperbola.

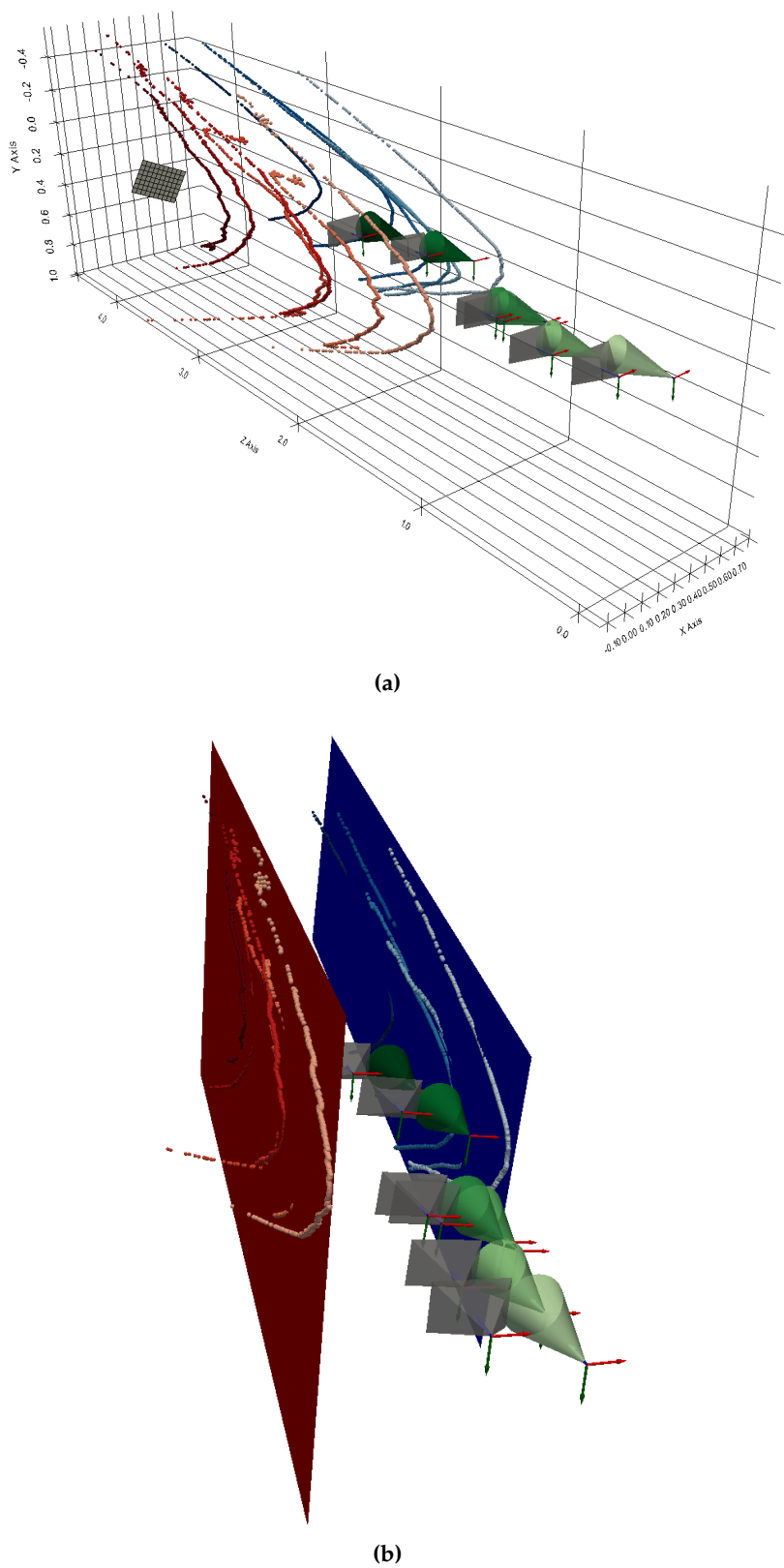


Figure 29. 3D reconstruction of the contours extracted from the six images.

We’re now going to compare this 3D reconstruction with reality. The aqueduct is a corridor whose left and right walls are parallel and separated by a width of 62cm. We can therefore check that :

- the 3D points resulting from the left part (respectively right part) of the contour belong to the same plane so they are supposed to be coplanar,
- If we estimate a plane from the left 3D points and a plane from the right 3D points, they must be parallel and separated by a distance of approximately 62cm.

As shown in Figure 29b, we get estimations of the left wall in red and the right wall in blue by using a least square method. To evaluate coplanarity, we compute the deviation between 3D points and their associated plane. For left points, we get an average deviation of 13,9mm and for right points 24,1mm (see Table 4).

To check the gap between the two estimated planes, we compute the distances between the left 3D points (respectively the right 3D points) and the right plane (respectively the left plane). The average distance between the left points and the right plane is 584,9mm. The average distance between the right points and the left plane is 583,6mm.

The two estimated planes are almost parallel since the angle calculated between the 2 normal vectors is 178,9°.

Table 4. Orthogonal distances between 3D points and estimated planes.

Dist. (mm)	Max.	Min.	Med.	Moy.	RMS
Left 3D points to left plane	220.5	0.0	11.0	13.9	19.9
Right 3D points to right plane	113.9	0.0	20.4	24.1	30.5
Left 3D points to right plane	778.4	492.1	581.9	584.9	585.2
Right 3D points to left plane	691.1	476.9	585.2	583.6	584.4

5. Discussion

Authors should discuss the results and how they can be interpreted from the perspective of previous studies and of the working hypotheses. The findings and their implications should be discussed in the broadest context possible. Future research directions may also be highlighted.

6. Conclusions

The aim of this paper was to propose a 3D reconstruction solution designed for underwater galleries found in karstic environments. The study of these environments represents a considerable challenge for the future, as they could provide part of our water needs. Exploration remains the best approach for acquiring reliable data on the structure, and is mainly carried out by qualified divers using topography methods with Ariane wires and section-by-section surveys. But on extensive karstic networks, this approach presents a major risk, as the duration of these surveys is so time-consuming, leading divers to long stops. The future is therefore more likely to lie with robotics, but there are still many technical challenges to be overcome before underwater drones that are sufficiently autonomous can be sent out to explore this type of environment. Our aim was to propose an original method of 3D scanning in this type of environment. Our approach uses a combination of a camera and a cone-shaped projector. The parameters of the light cone have been fixed: its centre, its direction vector and its angular aperture. A method for calibrating the cone was presented, enabling the position of the 3D points obtained by intersecting the visual rays of the contour of the projector halo as seen by the camera with the cone whose parameters have been estimated. The calibration of the camera is done thanks to a classical approach. A simulation of this method was presented and results validate the proposal. A first experimentation in real conditions was conducted in a disused aqueduct (without water) which had the advantage of reproducing an environment close to that of a karstic aquifer: narrow, without

light and above all without water to continue our evaluation. The camera calibration is validated with an error reconstruction less than 0.1 pixel and camera parameters close to technical values (see 4.2). The cone calibration leads to an angles error less than 2 degrees and distance with respect to the camera less than 2 mm. The results were validated by testing the coplanarity of the aqueduct walls to within 25mm and estimating the known distance between these walls to within 36mm (see Table 4). Experiments have been done in real karstic environment but due to complex meteorological conditions, visual acquisition was not possible. Thus, we aim to begin in a more controlled environment : a swimming pool and then finish in a real aquifer with clearer water.

To explore a new configuration of this system, we began to make some simulations with one projector and four cameras (Figure 30) to solve the error of the transition zone (see Figure 19). We also think about using several projectors with localisation sensors with respect to one or more cameras to have different configurations and measurement redundancy.

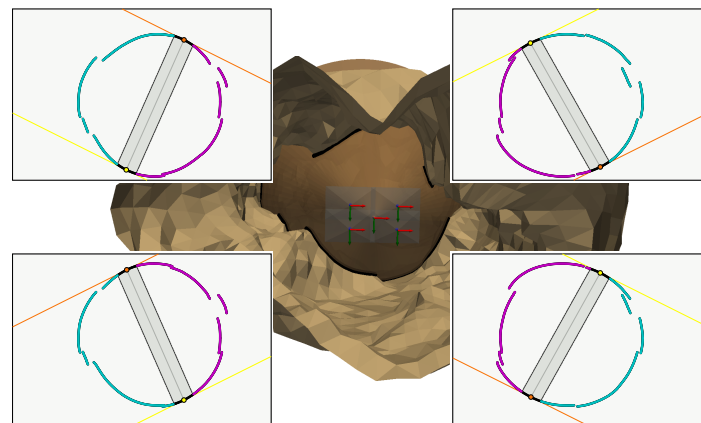


Figure 30. Configuration with 4 cameras.

References

1. Gleick, P.H. Water in crisis. *Pacific Institute for Studies in Dev., Environment & Security. Stockholm Env. Institute, Oxford Univ. Press.* 473p **1993**, 9, 1051–0761.
2. Massot-Campos, M.; Oliver-Codina, G. Optical sensors and methods for underwater 3D reconstruction. *Sensors* **2015**, *15*, 31525–31557. <https://doi.org/10.3390/s151229864>.
3. Hu, K.; Wang, T.; Shen, C.; Weng, C.; Zhou, F.; Xia, M.; Weng, L. Overview of Underwater 3D Reconstruction Technology Based on Optical Images. *Journal of Marine Science and Engineering*, *11*, 949. <https://doi.org/10.3390/jmse11050949>.
4. Snyder, J. Doppler Velocity Log (DVL) navigation for observation-class ROVs. In Proceedings of the OCEANS 2010 MTS/IEEE SEATTLE. IEEE, 2010, pp. 1–9. <https://doi.org/10.1109/oceans.2010.5664561>.
5. Lee, C.M.; Lee, P.M.; Hong, S.W.; Kim, S.M.; Seong, W.; et al. Underwater navigation system based on inertial sensor and doppler velocity log using indirect feedback kalman filter. *International Journal of Offshore and Polar Engineering* **2005**, *15*.
6. Brown, C.J.; Blondel, P. Developments in the application of multibeam sonar backscatter for seafloor habitat mapping. *Applied Acoustics* **2009**, *70*, 1242–1247. <https://doi.org/10.1016/j.apacoust.2008.08.004>.
7. Gary, M.; Fairfield, N.; Stone, W.C.; Wettergreen, D.; Kantor, G.; Sharp, Jr, J.M. 3D mapping and characterization of Sistema Zacatón from DEPTHX (Depth Permeable THermal Explorer). In *Sinkholes and the Engineering and Environmental Impacts of Karst*; 2008; pp. 202–212. [https://doi.org/10.1061/41003\(327\)20](https://doi.org/10.1061/41003(327)20).
8. Kantor, G.; Fairfield, N.; Jonak, D.; Wettergreen, D. Experiments in navigation and mapping with a hovering AUV. In Proceedings of the Field and Service Robotics. Springer, 2008, pp. 115–124. https://doi.org/10.1007/978-3-540-75404-6_11.
9. Mallios, A.; Ridao, P.; Ribas, D.; Carreras, M.; Camilli, R. Toward autonomous exploration in confined underwater environments. *Journal of Field Robotics* **2016**, *33*, 994–1012.

10. Hartley, R.I.; Sturm, P. Triangulation. *Computer vision and image understanding* **1997**, *68*, 146–157. <https://doi.org/10.1006/cviu.1997.0547>.
11. Massot-Campos, M.; Oliver-Codina, G. One-shot underwater 3D reconstruction. In Proceedings of the Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA). IEEE, 2014, pp. 1–4. <https://doi.org/10.1109/etfa.2014.7005282>.
12. Meline, A.; Triboulet, J.; Jouvencel, B. Comparative study of two 3D reconstruction methods for underwater archaeology. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2012, pp. 740–745. <https://doi.org/10.1109/iros.2012.6385711>.
13. Onmek, Y.; Triboulet, J.; Druon, S.; Meline, A.; Jouvencel, B. Evaluation of underwater 3D reconstruction methods for Archaeological Objects: Case study of Anchor at Mediterranean Sea. In Proceedings of the 2017 3rd International Conference on Control, Automation and Robotics (ICCAR). IEEE, 2017, pp. 394–398. <https://doi.org/10.1109/iccar.2017.7942725>.
14. Brandou, V.; Allais, A.G.; Perrier, M.; Malis, E.; Rives, P.; Sarrazin, J.; Sarradin, P.M. 3D reconstruction of natural underwater scenes using the stereovision system IRIS. In Proceedings of the OCEANS 2007-Europe. IEEE, 2007, pp. 1–6. <https://doi.org/10.1109/oceanse.2007.4302315>.
15. Beall, C.; Lawrence, B.J.; Ila, V.; Dellaert, F. 3D reconstruction of underwater structures. In Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2010, pp. 4418–4423. <https://doi.org/10.1109/iros.2010.5649213>.
16. Zhou, Y.; Li, Q.; Ye, Q.; Yu, D.; Yu, Z.; Liu, Y. A binocular vision-based underwater object size measurement paradigm: Calibration-Detection-Measurement (C-D-M). *Measurement*, *216*, 112997. <https://doi.org/10.1016/j.measurement.2023.112997>.
17. Weidner, N.; Rahman, S.; Li, A.Q.; Rekleitis, I. Underwater cave mapping using stereo vision. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2017, pp. 5709–5715. <https://doi.org/10.1109/icra.2017.7989672>.
18. Rahman, S.; Li, A.Q.; Rekleitis, I. Svin2: an underwater slam system using sonar, visual, inertial, and depth sensor. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2019, pp. 1861–1868. <https://doi.org/10.1109/iros40897.2019.8967703>.
19. Wu, X.; Wang, J.; Li, J.; Zhang, X. Retrieval of siltation 3D properties in artificially created water conveyance tunnels using image-based 3D reconstruction. *Measurement*, *211*, 112586. <https://doi.org/10.1016/j.measurement.2023.112586>.
20. Fan, J.; Wang, X.; Zhou, C.; Ou, Y.; Jing, F.; Hou, Z. Development, Calibration, and Image Processing of Underwater Structured Light Vision System: A Survey. *IEEE Transactions on Instrumentation and Measurement*, *72*, 1–18. <https://doi.org/10.1109/TIM.2023.3235420>.
21. Wang, W.; Joshi, B.; Burgdorfer, N.; Batsos, K.; Li, A.Q.; Mordohai, P.; Rekleitis, I. Real-Time Dense 3D Mapping of Underwater Environments. Technical report, arXiv, 2023. arXiv:2304.02704 [cs] type: article, <https://doi.org/10.48550/arXiv.2304.02704>.
22. Bräuer-Burchardt, C.; Munkelt, C.; Bleier, M.; Heinze, M.; Gebhart, I.; Kühmstedt, P.; Notni, G. Underwater 3D Scanning System for Cultural Heritage Documentation. *Remote Sensing*, *15*, 1864. <https://doi.org/10.3390/rs15071864>.
23. Yu, B.; Tibbetts, R.; Barua, T.; Morales, A.; Rekleitis, I.; Islam, M.J. Weakly Supervised Caveline Detection For AUV Navigation Inside Underwater Caves. <https://doi.org/10.48550/ARXIV.2303.03670>.
24. Hu, C.; Zhu, S.; Song, W. Real-time Underwater 3D Reconstruction Based on Monocular Image. In Proceedings of the 2022 IEEE International Conference on Robotics and Biomimetics (ROBIO), 2022, pp. 1239–1244. <https://doi.org/10.1109/ROBIO55434.2022.10011639>.
25. Marchand, E.; Uchiyama, H.; Spindler, F. Pose estimation for augmented reality: a hands-on survey. *IEEE transactions on visualization and computer graphics* **2015**, *22*, 2633–2651. <https://doi.org/10.1109/tvcg.2015.2513408>.
26. Massone, Q.; Druon, S.; Triboulet, J. An original 3D reconstruction method using a conical light and a camera in underwater caves. In Proceedings of the Proceedings of the 4th International Conference on Control and Computer Vision, 2021, pp. 126–134. <https://doi.org/10.1145/3484274.3484294>.
27. Massone, Q.; Druon, S.; Breux, Y.; Triboulet, J. Contour-based approach for 3D mapping of underwater galleries. In Proceedings of the Global Oceans 2020: Singapore–US Gulf Coast. IEEE, 2020, pp. 1–6. <https://doi.org/10.1109/ieeecnf38699.2020.9388982>.

28. Halir, R.; Flusser, J. Numerically stable direct least squares fitting of ellipses. In Proceedings of the Proc. 6th International Conference in Central Europe on Computer Graphics and Visualization. WSCG. Citeseer, 1998, Vol. 98, pp. 125–132.
29. Moré, J.J. The Levenberg-Marquardt algorithm: implementation and theory. In *Numerical analysis*; Springer, 1978; pp. 105–116. <https://doi.org/10.1007/bfb0067700>.
30. Zhang, Z. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence* **2000**, 22, 1330–1334. <https://doi.org/10.1109/34.888718>.
31. Fischler, M.A.; Bolles, R.C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM* **1981**, 24, 381–395. <https://doi.org/10.1016/b978-0-08-051581-6.50070-2>.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.