

The Pinax-model for accurate and efficient refraction correction of underwater cameras in flat-pane housings

Tomasz Łuczyński*, Max Pfingsthorn, Andreas Birk

Robotics, Computer Science and Electrical Engineering, Jacobs University Bremen, Campus Ring 1, 28759 Bremen, Germany

ARTICLE INFO

Keywords:

Underwater vision
Calibration
Rectification
Refraction correction
Camera model
Axial camera

ABSTRACT

The calibration and refraction correction process for underwater cameras with flat-pane interfaces is presented that is very easy and convenient to use in real world applications while yielding very accurate results. The correction is derived from an analysis of the axial camera model for underwater cameras, which is among others computationally hard to tackle. It is shown how realistic constraints on the distance of the camera to the window can be exploited, which leads to an approach dubbed Pinax Model as it combines aspects of a virtual pinhole model with the projection function from the axial camera model. It allows the pre-computation of a lookup-table for very fast refraction correction of the flat-pane with high accuracy. The model takes the refraction indices of water into account, especially with respect to salinity, and it is therefore sufficient to calibrate the underwater camera only once in air. It is demonstrated by real world experiments with several underwater cameras in different salt and sweet water conditions that the proposed process outperforms standard methods. Among others, it is shown how the presented method leads to accurate results with single in-air calibration and even with just estimated salinity values.

1. Introduction

Cameras are very important sensors for underwater applications in general. This includes ship hull, pipeline and other inspection missions (Hollinger et al., 2012; Kim and Eustice, 2013; Foresti, 2001; Asakawa et al., 2000; Negahdaripour and Firoozfam, 2006; McLeod et al., 2013; Galceran et al., 2014), habitat mapping (Davie et al., 2008; Bodenmann et al., 2013), vehicle station-keeping (Negahdaripour and Fox, 1991; Marks et al., 1994; Lots et al., 2000), archeology (Bingham et al., 2010; Chapman et al., 2010; Hue et al., 2011) or search and recovery missions (Purcell et al., 2011) just to name a few examples. A short overview with respect to underwater vision on unmanned underwater vehicles (UUV), e.g., is given in Horgan and Toal (2006).

Flat-panel glass windows are commonly used for underwater camera housings. While domes provide optical advantages, they have to be specially engineered to fit the camera and the integration is not trivial. Flat-pane windows are hence simply a much less expensive and more flexible choice. On the other hand, flat ports introduce significant distortions due to the refraction at the air-glass and glass-water interfaces. The predominant way to handle the distortions is to use a standard perspective projection model and to perform standard camera calibration in-situ, i.e., in the water or by including estimated correction factors, see e.g., Shortis and Harvey (1998), Gracías and Santos-

Victor (2000), Pessel et al. (2003), Pizarro et al. (2003), Lavest et al. (2003), Negahdaripour et al. (2006), Negahdaripour et al. (2007), Brandou et al. (2007), Sedlazeck et al. (2009), Johnson-Roberson et al. (2010), Kunz and Singh (2010), Beall et al. (2011), Kang et al. (2012).

Treibitz et al. (2008, 2012) show that flat port cameras do not possess a single viewpoint (SVP), i.e., the perspective projection model is invalid for flat ports. This is also supported by other works (Li et al., 1997; Kunz and Singh, 2008; Chari and Sturm, 2009; Gedge et al., 2011; Yamashita et al., 2011; Sedlazeck and Koch, 2011; Jordt-Sedlazeck and Koch, 2012, 2013; Agrawal et al., 2012; Servos et al., 2013; Chen and Yang, 2014; Yau et al., 2013).

In Kunz and Singh (2008) the errors caused by not compensating the refractive distortions are discussed in some detail and they are identified as significant, however no solution to this problem is presented. A mathematical model of underwater imaging through planar glass ports is introduced in Chari and Sturm (2009). Matrices corresponding to fundamental and homography matrices are derived. They, however, depend on the incident angle of the light ray corresponding to each image pixel, so they cannot be used directly for underwater vision methods. Since no continuation of this work was published, their results remain as theoretical considerations of conceptual value. Apart from analyzing the general problem from a largely theoretical perspective, Treibitz et al., (2008, 2012) provide an

* Corresponding author.

E-mail addresses: t.luczynski@jacobs-university.de (T. Łuczyński), m.pfingsthorn@jacobs-university.de (M. Pfingsthorn), a.birk@jacobs-university.de (A. Birk).

approach for a single refractive layer, i.e., when the window is negligibly thin and the problem can be reduced to only a single air-water interface.

Important insights into the problem and ways towards a solution are presented in Agrawal et al. (2012) where a flat port camera is identified to be in fact an axial camera. Agrawal et al. (2012) derive a 12th degree polynomial that must be solved to project a 3D point onto an image plane in this case. A method is proposed for calibration of the camera but it requires knowledge of the full 3D geometry of the calibration points in the environment - a requirement which is difficult if not impossible to fulfill in underwater applications. Furthermore, the underlying axial model does not allow for a rectification of single images as the axial model implies that the points are lying on complex curves. Correspondences across multiple images can, in principle, be exploited, but this is computationally very complex, as also pointed out in Jordt-Sedlazeck and Koch (2013).

When using multiview methods, the SVP model can lead to reasonable results as explicitly discussed in Kang et al. (2012). Nevertheless, Jordt-Sedlazeck and Koch (2013) rely on the results from Agrawal et al. (2012) by proposing a refractive Structure from Motion (SfM) method by augmenting the standard perspective SfM process by incorporating a new error function in the optimization and report clear improvements. In their further work Jordt et al. (2016) integrated (Jordt-Sedlazeck and Koch, 2013; Jordt-Sedlazeck et al., 2013) into a complete system for refractive reconstruction by improving the non-linear optimization. This allowed them to use their method on larger scenes. While this is an interesting approach, it requires sufficiently many images with sufficiently different views of the scene and it is still computationally very demanding. It should be noted that though we use stereo vision for validation purposes in the experiment section, we do not use any two- or multiview information. Stereo data is just used in our experiments as it facilitates a metric analysis of the errors. Our method is perfectly suited for the calibration and rectification of single camera images, i.e., the most general case of underwater vision. Our method can hence also be of interest for stereo or multiview approaches - including refractive ones like in Jordt-Sedlazeck and Koch (2013) - by providing excellent initial guesses for the camera parameters, hence allowing for faster convergence and possibly even more accurate results.

In this article we make two main contributions. Firstly, we discuss the problem of underwater camera modeling from a practitioner's viewpoint. We provide illustrative examples of the underlying effects and their relevance to real world applications. To some extent, this also bridges some apparent contradictions found in the literature that can be explained when contrasting theoretical considerations with typical application cases. Secondly, we provide an approach for calibration and refraction correction of underwater images that is very convenient to use in real world applications and that is at the same time very accurate. This *pinax* model is based on a virtual *pinhole* camera model - which is demonstrated herein to be applicable for real world underwater housings where the camera is relatively close to the flat-pane - while using the projection function of an *axial* camera. The *pinax* model incorporates the water refraction index, for which - as also experiments show - it is sufficient to derive it through (estimated) salinity to achieve accurate results. It is hence sufficient to calibrate the underwater camera only once in air, thus replacing tedious in-water calibrations before or during missions. For the rectification, a look-up table is generated using the projection function of the axial model, for which we show that it can be used in a significantly simplified fashion within the *pinax* model. The look-up table can be easily computed a priori and allows very fast real-time refraction correction of single images. Real world experiments with different cameras in different sweet and salt water environments show that the *pinax* model outperforms standard methods. The code for using the *pinax* model is provided in MATLAB, and in the Robot Operating System (ROS) and is freely usable for academic purposes.

2. Problem formulation and motivational examples

2.1. Flat-panel camera setup

The following setup is considered. A physical camera ${}^P\text{Cam}$ that follows the standard SVP (single viewpoint) model with an intrinsic matrix ${}^P\mathbf{K}$ is enclosed in a water sealed housing with a flat glass panel through which it observes the underwater environment. The glass panel is flat and both sides are parallel. The glass panel introduces distortions that are to be handled by a virtual camera model ${}^V\text{Cam}$ that interprets the environment scene from the physical camera ${}^P\text{Cam}$. The overall underwater setup of the physical camera plus its housing with a flat-pane window submerged into water is denoted as the underwater camera ${}^U\text{Cam}$. When the underwater camera is in air, e.g., for the calibration, it is denoted with ${}^A\text{Cam}$.

If not mentioned otherwise, in this article we use the term *camera* for the complete underwater set-up and use the terms *virtual camera* and *physical camera* to refer to the model ${}^V\text{Cam}$ of the glass-panel refraction, or to the in-air physical device ${}^P\text{Cam}$ inside the housing, respectively.

The main object of interest for this article is the virtual camera model ${}^V\text{Cam}$ to handle the refraction induced distortions. The related notations and a schematic view are presented in Fig. 1. Following parameters are used:

- d_0 - distance from the center of projection of ${}^P\text{Cam}$ to the glass window,
- d_1 - thickness of the glass,
- x - distance to point of intersection of the light ray with the camera axis,
- Δx - length of the focus section (discussed in more detail in Section 2.3),
- n_a, n_g, n_w - refraction indexes (scaled so that $n_a=1$),
- $\mathbf{n}$ - normal vector to the glass surface,
- α - incident angle.

2.2. The flat port setup as an axial camera

As shown in Agrawal et al. (2012), the physically accurate model of a flat-port underwater camera corresponds to an axial camera model. So, light rays creating the image do not intersect in one point, as in the

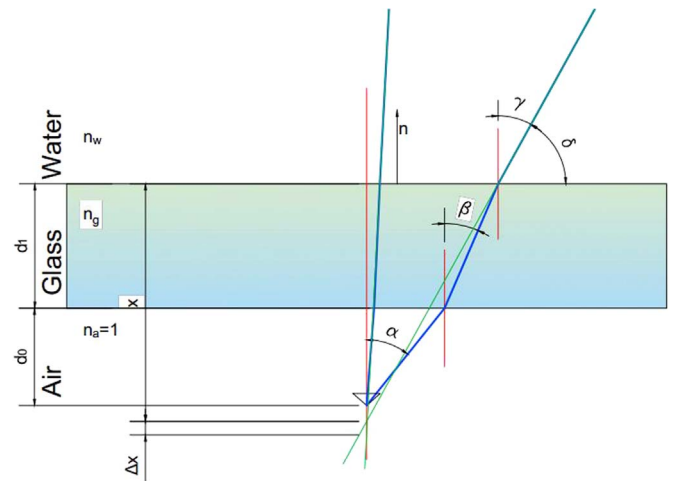


Fig. 1. Schematic view of a Flat Port setup: d_0 - distance from the center of projection to the glass window, d_1 - thickness of the glass, x - distance to point of intersection of the light ray with the camera axis, Δx - length of the focus section, n_a, n_g, n_w - refraction indexes, scaled so that $n_a=1$, $\mathbf{n}$ - normal vector to the glass surface, α - incident angle. The blue line represents the physically accurate light ray; the green line is the apparent ray traced back to the camera's optical axis. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

SVP pinhole model, but they all intersect one line, called the axis of the camera. Using the pinhole camera model thus requires approximating the focus section, i.e., the line segment on the axis on which rays cross, with a single point. The conclusion is that the quality of this approximation depends directly on the length Δx of this section. In the limit case, the pinhole camera can be seen as an axial camera where the focus section of the camera axis is infinitesimally short. To analyze the refraction, ray tracing through the air-glass-water interface and the apparent intersection of the rays in the water with the camera axis can be modelled (compare Fig. 1):

$$\beta = \arcsin \frac{\sin \alpha}{n_g}$$

$$\gamma = \arcsin \frac{\sin \alpha}{n_w}$$

$$\delta = \frac{\pi}{2} - \gamma$$

For the sake of simplicity, we assume that the refractive plane normal and therefore the camera axis in the axial model is parallel to the optical axis of the camera. This assumption is without loss of generality since the incident angle α , i.e., the only parameter related to camera rotation, is one of the inputs, which can be easily rotated by a fixed off-set. For the sake of completeness, the equations for finding incident angles α , given the camera pose in the housing is:

$$\mathbf{v}_0 = \mathbf{K}^{-1} \mathbf{p}$$

$$\alpha = \arccos \frac{\mathbf{v}_0^T \mathbf{n}}{\|\mathbf{v}_0\| \|\mathbf{n}\|}$$

where $\mathbf{K}$ is the intrinsic parameter matrix and $\mathbf{p}$ represents pixel coordinates on the image.

The focus distance x for each light ray (Fig. 1) can be computed as:

$$x = \tan \delta (d_0 \tan \alpha + d_1 \tan \beta)$$

2.3. Length of the focus section

Consider an example setup with a glass refraction index $n_g=1.5$, a water refraction index of $n_w=1.335$ and a glass thickness of $d_1 = 10$ mm. Plotting the change of x as a function of the incident angle α and of the distance d_0 illustrates a very important aspect (Fig. 2). As

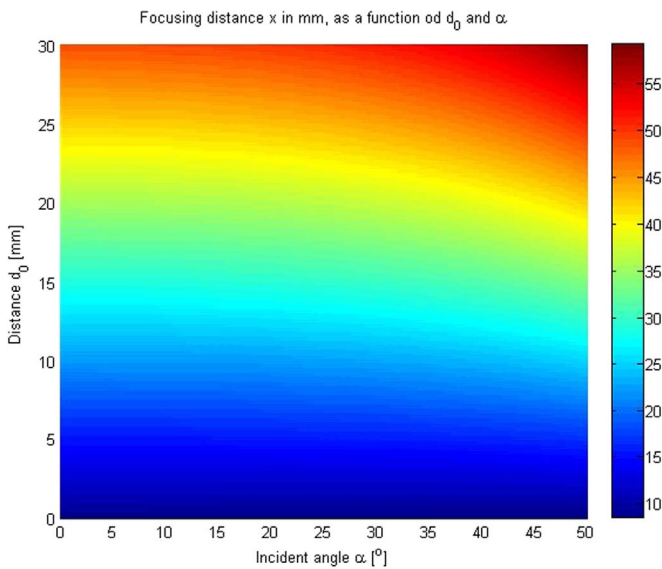


Fig. 2. The focus distance x (in mm) as a function of d_0 and α for an example setup with a glass refraction index of $n_g=1.5$, a water refraction index of $n_w=1.335$ and a glass thickness of $d_1 = 10$ mm. It can be seen that the changes in the focus distance x for different incident angles α become more significant for increasing distances d_0 of the physical camera to the flat-plane.

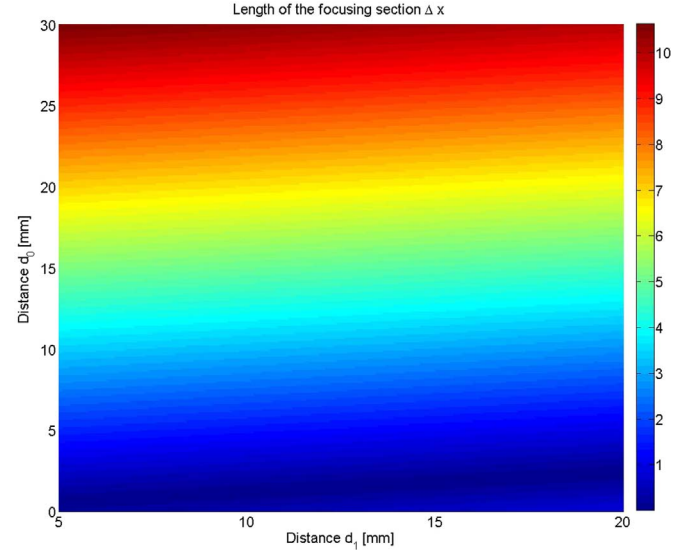


Fig. 3. The length of the focus section (Δx) as a function of d_0 and d_1 . It can be seen that the influence of the distance d_0 of the physical camera to the flat-plane is more significant than the thickness d_1 of the glass-plane.

d_0 grows, the changes in the focus distance depending on the incident angle α (along X axis) become more significant, i.e., there is a higher range of focus distances with increasing d_0 . This is further illustrated in the following.

Fig. 3 shows the length of the focus section Δx as a function of d_0 and d_1 . The plot shows that changes in d_0 are much more significant than changes in d_1 , i.e., the distance of the camera to the flat-plane window has a stronger effect than the thickness of the glass window. This effect is caused by a relatively small difference between the refraction index of glass (≈ 1.5) and the average water refraction index (≈ 1.33 , (Roswell et al., 1976)) compared to the more significant refraction on the glass-air interface. In Fig. 3 it can be also observed that the best approximation of the axial camera model with a pinhole model occurs for small values of d_0 .

To further motivate and illustrate this, Fig. 4 shows where the light rays in water cross the camera's optical axis for different values of d_0 . Each line on the graph corresponds to a different incident angle ranging from 0 to 35 degrees, i.e., a physical camera with a field of view of 70 degrees. It can be seen that they never cross the same spot,

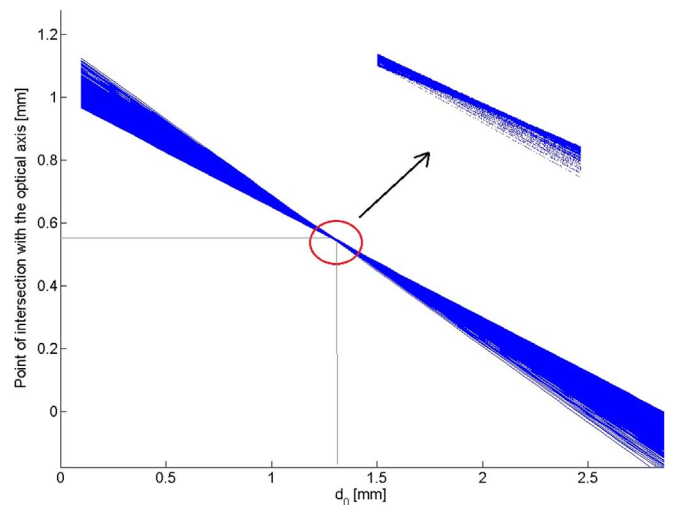


Fig. 4. Example distances where the light rays traced back from the water cross the optical axis of the camera depending on d_0 . Different lines correspond to different incident angles ranging from 0 to 35 degrees, i.e., a physical camera with a field of view of 70 degrees.

but for some optimal d_0 , they are very close to intersecting in one point. To find this optimal value for some given parameters the following method is used. We implemented ray tracing based on the above formulation of the model. Then non-linear optimization is used to minimize the length of the section where light rays back-traced from the water intersect with the camera optical axis. For example, for the case where $d_1 = 10$ mm, the glass refraction index $n_g = 1.5$, and the water refraction index $n_w = 1.335$, the method converges to $d_0 = 1.4282$ mm where all light rays intersect the optical axis on a section Δx that is only 0.0079 mm long, i.e., within a very good approximation of a single point. The result of this numerical analysis allows us to define the middle of this section Δx as a secondary center of projection placed 0.5851 mm away from the glass panel. For this case, the virtual camera can be treated as an SVP camera and represented with the pinhole model.

This example motivates that although the pinhole camera model does not represent the actual physical state, for the purpose of underwater vision it may be used as basis for a model if the distance between the center of projection of pCam and the glass plane is very small. This is a realistic assumption as there are no reasons to design excessive housing sizes, i.e., the physical camera inside a housing is usually placed quite close to the flat-plane window.

2.4. Influence of the distance of the camera to the flat-panel

The influence of the distance of the physical camera to the flat pane is now further illustrated in a other motivational example. The thickness of the glass panel d_1 is assumed to be constant at $d_1 = 10$ mm. For different values of d_0 , the camera is calibrated with a standard procedure. The calibration input data is based on 27 simulated checkerboards in 3D space (Fig. 5) by projecting the corner points to the image plane using the full physical model including refraction. This forward projection requires solving the twelve-degree polynomial introduced in Agrawal et al. (2012). This data is hence used to calibrate the camera as if it were underwater.

In a second step, a set of 100 random 3D points in front of the underwater camera is generated. This set is then projected onto the image plane twice for each d_0 . Firstly, the projection is performed with the full physical model to get the expected image coordinates and the second time with the camera matrix from calibration using the pinhole model including undistortion. Then the distance between corresponding points (also called reprojection error) is calculated and used to evaluate the pinhole approximation for the different d_0 . Figs. 6 and 7

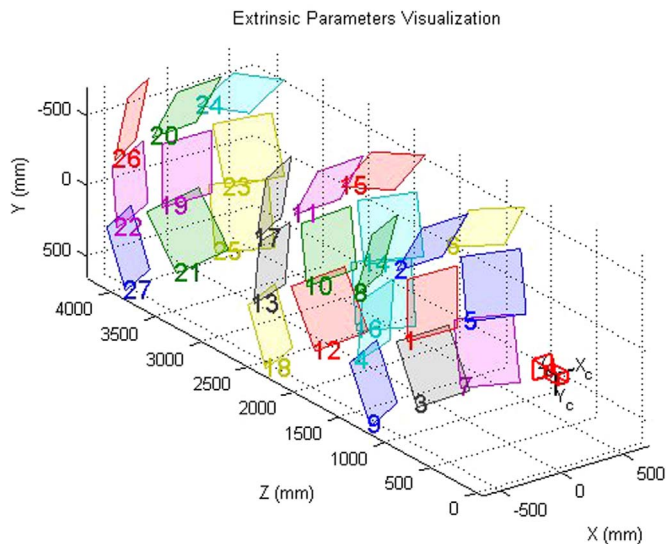


Fig. 5. Poses of the calibration patterns used for a further motivational experiment in simulation.

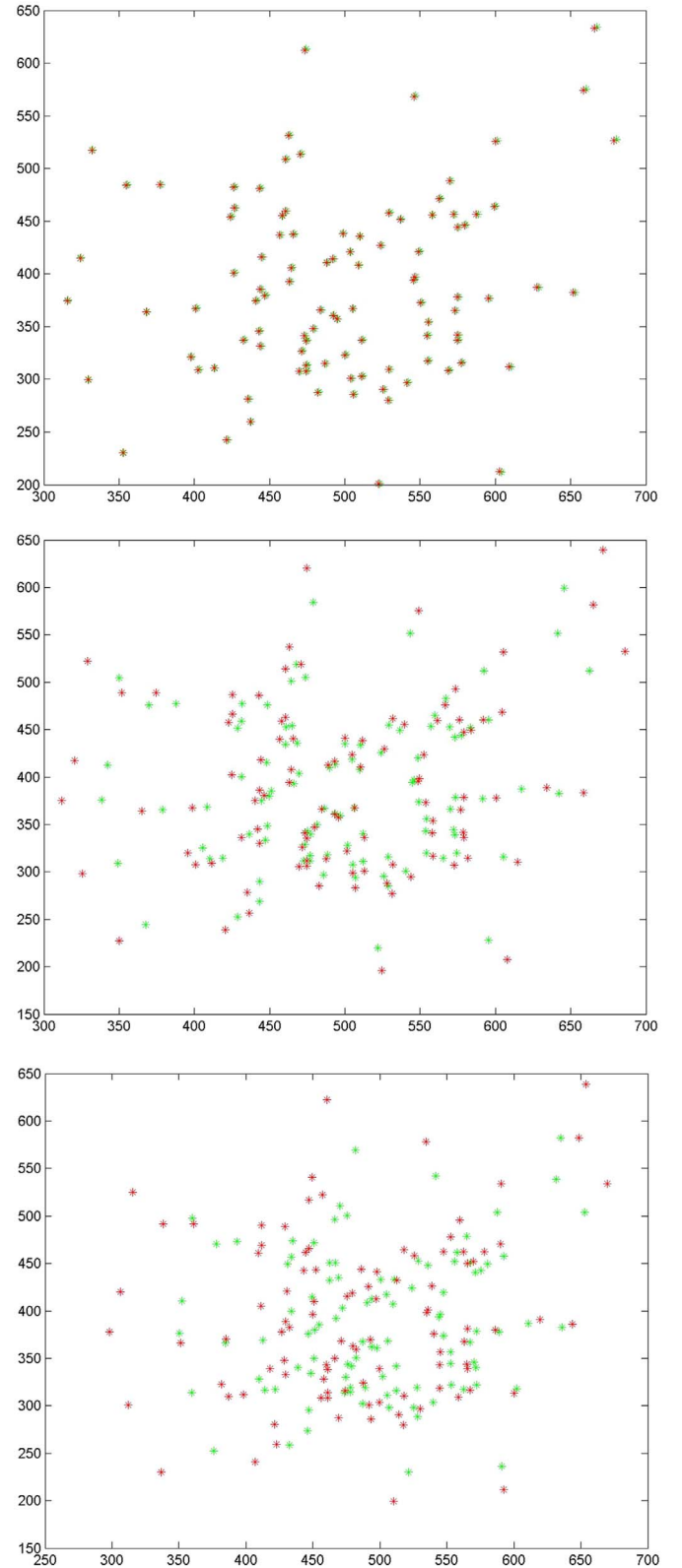


Fig. 6. An example with a refraction-based (green) and a pinhole (red) projection of random points in the scene for $d_0 = 1$ mm (top), 300 mm (center), and 500 mm (bottom) respectively. Please note the increasing deviations in the models with increasing d_0 . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

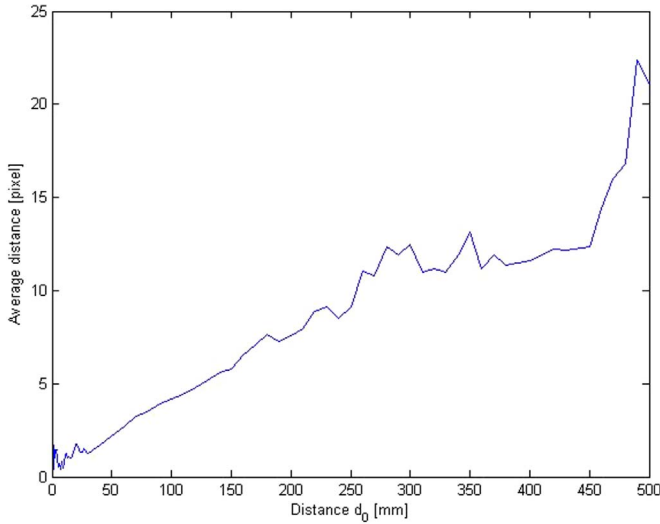


Fig. 7. An example of the reprojection error for changing d_0 . Note that for $d_0 = 0 - 10$ mm the errors caused by the SVP approximation can be neglected. The graph is not as smooth as may be expected, e.g., as in Fig. 3, because the simulated patterns were not always in the optimal positions for calibration, e.g., they did not always cover the whole field of view of the camera - which is a very natural effect that can also be observed in real world conditions.

show the results by plotting the average distance between corresponding points for the different d_0 values.

It should be noted that we consider here values of d_0 up to 500 mm. Values within this range can be found in the literature for experimental setups, especially in the highly relevant works of Agrawal et al. (2012) and Treibitz et al. (2012). Such big values are used because then the effects of the axial camera model are clearly visible and for example the position of the camera in the housing can be found with nonlinear optimization. There may be applications where the distance of the physical camera to the flat-plane is quite large, e.g., when observing objects in an aquarium and the physical camera needs to keep a significant clearance to the aquarium window for some reason. However, this scenario is very unrealistic for underwater cameras. Excessive housing sizes to allow for significant distances d_0 are neither necessary nor desirable for underwater applications.

2.5. Rectification accuracy near the calibration distance for SVP

The following simulation example is designed to illustrate that regardless of the setup parameters, it is possible to get a reasonable approximation of the physical state with a standard SVP pinhole model, if the observed part of the scene is always observed from roughly the same distance D and the camera calibration was performed at about the same distance, i.e., the calibration pattern was moved underwater in front of the camera, roughly in the distance D as well. This illustrates that, e.g., for mosaicking with a vehicle camera in a (roughly) fixed distance over ground, good rectification results with a pinhole model can be achieved if the calibration pattern was moved in water at the roughly same distance. On the other hand, errors emerge once the camera is looking at parts of the scene that are closer or further than D .

The setup parameters are as follows: $d_0 = 80$ mm, $d_1 = 20$ mm, i.e., a significant amount of space between the focal point of the physical camera lens and the glass panel plus a relatively thick glass pane. So, the parameters, especially d_0 , are in this case relatively unfavorable.

In this illustrating example, the camera is calibrated with 27 pattern poses spread around a point 2 m away from the camera (Fig. 8). Then test points are generated again randomly but around a given distance from the camera and, using the same method as above, projected onto the image plane. The reprojection error against the point distance to the camera is shown in Fig. 9.

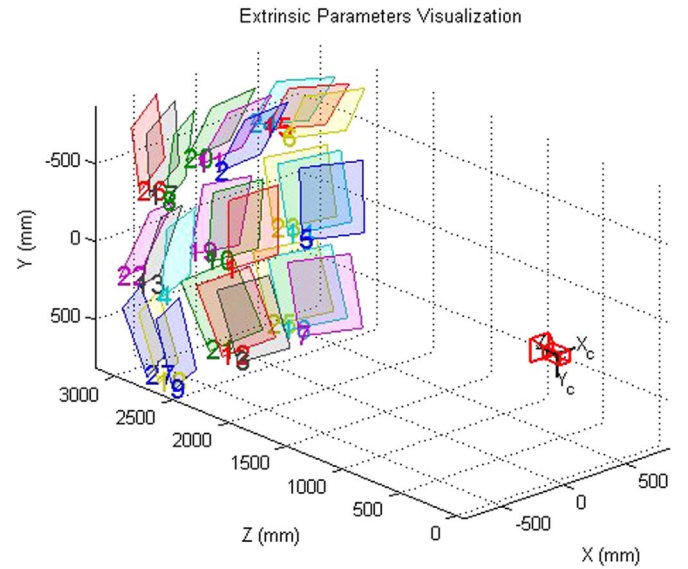


Fig. 8. Arrangement of calibration patterns used for calibrating the camera in an example illustrating the effects of the distance of the calibration in the pinhole model under unfavorable parameter conditions.

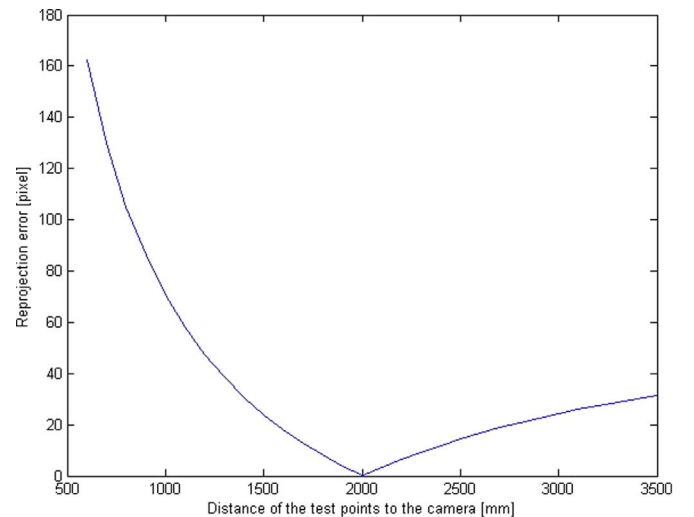


Fig. 9. The reprojection error for a changing distance of observed points to the camera as an example that the SVP pinhole model performs well if the observed points are close to the distance in which the camera was calibrated with an SVP model. In this simulation example, the camera was calibrated with patterns around 2 m away from the camera (Fig. 8), which is exactly the distance where the reprojection error is minimal.

The pinhole model holds very well only around the distance of calibration. This shows that for some specific applications, where minimizing d_0 is not possible, e.g., due to physical size of the lens as part of the camera subcomponent in the housing, the pinhole model can still be used effectively if the environment is observed from a known constant distance. As mentioned, one of the applications fulfilling this assumption can be seabed mosaicking with constant altitude control of the observing AUV.

This effect can also be observed in Kang et al. (2012) where the quality of Structure from Motion under an SVP model is investigated and good results are reported even for a larger distance of the camera to the window. The camera rig used in the experiments leads to a constant distance between the camera and the investigated object; hence the effect illustrated in this subsection takes place.

3. The Pinax model

3.1. Overview

Based on the previous considerations, we propose a system where a few setup assumptions are used to compensate for the refraction-based distortions of the image. Specifically, a transformation is computed to undistort and rectify the camera images. The resulting images can be directly used for example in stereo vision algorithms or for mosaicking, just to name two examples.

The following assumptions are made:

1. The distance d_0 between the glass and the center of projection is small and near the optimal spot d_0^* where the rays traced back from the water cross in a minimum focus section Δx^* .
2. The optical axis of the camera is perpendicular to the glass surface. If not, a correcting transformation can simply be applied, e.g., for converged stereo systems.
3. The glass thickness and its approximate refraction index are known, e.g., using standard refraction indexes for glass or plexiglass.
4. The water refraction index is approximately known from tables, e.g., from Roswell et al. (1976).

Fulfilling these assumptions allows us to assume a pinhole model for the virtual camera and hence allows us to model the refraction-based distortions very efficiently. It also makes it possible to omit any underwater calibration procedures. The first assumption in the above list is of course by far the strongest and most significant one. As motivated before, it is at least not unrealistic to assume that underwater housings are minimized in size and that hence the physical camera inside the housing is placed as closely as possible to the window. This assumption is also supported by the real world experiments presented later on.

3.2. In-air calibration

As the first step in our method, the physical camera ${}^p\text{Cam}$ is calibrated once in air, i.e., its intrinsic matrix ${}^p\mathbf{K}$ is determined using any standard calibration process (Hartley and Zisserman, 2003). From a practical viewpoint it is very interesting to note that the front window does not have to be removed from the housing. To be more precise, the physical camera ${}^p\text{Cam}$ can be calibrated by calibrating the underwater camera ${}^u\text{Cam}$ in air, i.e., by determining its intrinsic matrix ${}^u\mathbf{K}$.

The air-glass and glass-air refractions only lead to a change in scale in the images, which is part of the extrinsics, and the relative geometric relations between points in the scene are preserved. A calibration process of the underwater camera ${}^u\text{Cam}$ in air is hence the same as when calibrating ${}^p\text{Cam}$, i.e., ${}^p\mathbf{K} = {}^u\mathbf{K}$.

If the calibration of the physical camera was already done outside the housing, e.g., by the manufacturer, it is of course perfectly fine to use that data. The in-air calibration of the full underwater system is only an option that is very convenient to use for already existing complete camera systems. For high quality in-air calibration, the tool in CamOdoCal (Heng et al., 2013, 2014, 2015) is used in our experiments presented later on in Section 4.

3.3. Determining the optimal d_0^*

In an ideal scenario, the optimal distance d_0^* between the glass and the center of projection can be taken into account when designing a new underwater camera. More precisely, the optimal distance d_0^* of the physical camera should be taken into account, as the model of the virtual camera ${}^v\text{Cam}$ has its own, slightly different ${}^v d_0^*$ as discussed in more detail in the following section.

As already sketched in Section 2.3, ray tracing and non-linear optimization can be used to minimize the length of the section where light rays back-traced from the water intersect with the camera optical axis. The MATLAB code for these computations is provided as supplementary material to this article.

Table 1

Optimal $d_0^*/{}^v d_0^*$ of the centers of projection of the physical/virtual camera for different glass thicknesses and two common salinity values.

d_1 [mm]	$n_{w0}=1.333$ (sweet water)	$n_{w0}=1.342$ (salty water)
1	0.15 mm/0.06 mm	0.14 mm/0.06 mm
3	0.45 mm/0.18 mm	0.42 mm/0.17 mm
5	0.76 mm/0.31 mm	0.70 mm/0.29 mm
10	1.52 mm/0.61 mm	1.40 mm/0.58 mm
15	2.28 mm/0.92 mm	2.10 mm/0.87 mm
20	3.04 mm/1.22 mm	2.80 mm/1.15 mm

Using ${}^p d_0^*$ in a camera design is the ideal scenario and we only include its computation here for the sake of completeness. In most application cases, the underwater camera is an off-the-shelf system or an already finished design. Other design constraints on the housing or the physical camera/lens components may apply as well. But we consider it safe to assume that for any typical underwater housing the real distance $\widehat{d}_0$ is sufficiently close to ${}^p d_0^*$.

As illustrated in Table 1, ${}^p d_0^*$ tends to be in the order of a few millimeters and less. At the same time, the physical length of lenses tends to be in the order of their focal lengths, i.e., the center of projection tends to be at the front-end of the lenses of the camera device. Hence placing the physical camera as close as possible to the glass-pane with possibly a small air gap, i.e., using the standard default set-up for typical underwater cameras, leads to a close approximation of ${}^p d_0^*$ by $\widehat{d}_0$ with negligible errors. An exact quantification of the related errors is discussed below in the following sections.

3.4. Refraction and lens distortion modeling with maps

The main conclusion from the assumptions in Section 3.1, especially from assumption 1 about the distance of the physical camera to the window, is that a pinhole camera model can be used for the virtual camera model with a negligible error. To be more precise, we exploit this insight by defining a virtual pinax plane $\mathbf{p}_{pa} = (d_{pa}, \mathbf{n}_{pa})$ that is assumed to be at distance d_{pa} in the scene with a normal vector $\mathbf{n}_{pa}$ that is parallel to the camera axis. The distance d_{pa} is set fixed to 5 m as this is considered a typical viewing distance; but as discussed below, the exact value is of minor interest as points on pinax planes at different distances behave similar due to the virtual pinhole camera property that follows from small values of d_0 . ${}^v\mathbf{p}$ and ${}^p\mathbf{p}$ denote homogeneous image pixel coordinates. The intensity or color value of a given pixel is denoted as $I(\mathbf{p})$.

Each point ${}^v\mathbf{p}$ from the image vI of the virtual camera ${}^v\text{Cam}$ is projected onto the pinax plane $\mathbf{p}_{pa}$ using a pinhole camera projection. Then this point $\mathbf{m}_w$ is projected forward to the inside surface of the glass panel (point $\mathbf{m}_a$) using the method derived from Agrawal et al. (2012). Now $\mathbf{m}_a$ may be transformed to ${}^p\mathbf{p}$ with the in air calibration parameters of ${}^p\text{Cam}$ to obtain pixel coordinates in the distorted image pI . This last step may be performed using any camera and lens distortion model, referenced in Algorithm 1 with the subroutine `project3dToImage()`. When the calibration of the physical camera is based on a pinhole camera model with no lens distortion, this is:

$${}^p\mathbf{p} = {}^p\mathbf{K} \cdot \mathbf{m}_a$$

$${}^p\mathbf{p} = {}^p\mathbf{p} \cdot \frac{1}{p_z}$$

In order to find the point $\mathbf{q}_1$ (point corresponding to $\mathbf{m}_a$, expressed in coordinate frame z_1, z_2) as shown in Fig. 10 (compare also Fig. 1), the twelfth-degree polynomial method derived in Agrawal et al. (2012) is used. For the sake of completeness we shortly present the most important findings of Agrawal et al. (2012) here. As discussed before, it can be shown that a camera behind a flat glass panel is an axial camera. The camera axis is assumed to be identical to the optical axis. When tracing the light path all the refractions happen in one plane, called plane of refraction (POR), so the analysis can be conducted in 2D. To do this $\mathbf{m}_w$ must be projected to POR. The new coordinate system is defined as follows. Axis z_1 is identical to the

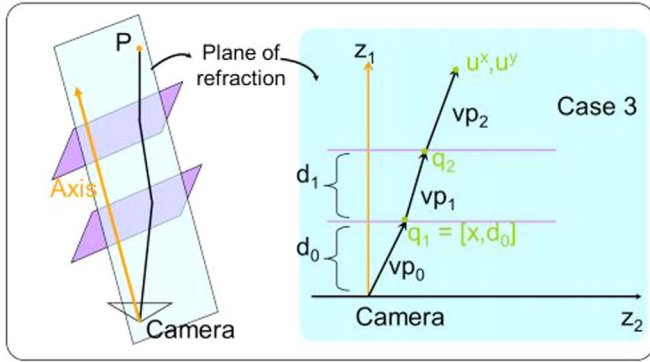


Fig. 10. Parameters of the analytical forward projection through a flat refractive panel with a 12th degree polynomial (From supplementary materials to Agrawal et al., 2012).

camera axis, z_2 is orthogonal to z_1 and lays on POR. This way $\mathbf{m}_w$ projected to $[u^x, u^y]$ may be used for calculations.

$$(k_1^2 D_1 + k_3^2 D_2 - k_2^2 D_1 D_2)^2 - 4k_1^2 k_3^2 D_1 D_2 = 0$$

where

$$k_1 = x(d_0 + d_1 - u^y)$$

$$k_2 = (u^x - x)$$

$$k_3 = -d_1 x$$

$$D_1 = d_0^2 n_g^2 + n_g^2 x^2 - x^2$$

$$D_2 = d_0^2 n_w^2 + n_w^2 x^2 - x^2$$

Algorithm 1. Creating correction maps in the Pinax Model (compare Fig. 11).

let M be an associative array

for $\mathbf{p} \in \mathcal{I}$ **do**

```

 $\mathbf{m}_w = {}^v K^{-1} \cdot \mathbf{p} \cdot d_{pa}$ 
 $\mathbf{z}_1 = (0, 0, 1)^T$ 
 $\mathbf{z}_2 = \mathbf{z}_1 \times (\mathbf{z}_1 \times \mathbf{m}_w)$ 
 $u^x = \mathbf{z}_2 \cdot \mathbf{m}_w$ 
 $u^y = \mathbf{z}_1 \cdot \mathbf{m}_w$ 
 $q_1 = \text{solve12thDegPoly}(\text{setupParams}, [u^x, u^y])$ 
 $\mathbf{m}_a = q_{1,x} \cdot \mathbf{z}_1 + q_{1,d} \cdot \mathbf{z}_2$ 
 $\mathbf{p} = \text{project3dToImage}(\text{CameraAndLensModel}, \mathbf{m}_a)$ 
store key–value pair  $(\mathbf{p}, \mathbf{p})$  in  $M$ 

```

end

Algorithm 2. Applying Pinax correction maps.

Let M be an associative container created with Algorithm 1

for $\mathbf{p} \in \mathcal{I}$ **do**

```

look up value  $\mathbf{p}$  for key  $\mathbf{p}$  in  $M$ 
 $\mathbf{a} = \text{floor}(\mathbf{p})$ 
 $b_x = \mathbf{p}_x - \mathbf{a}_x$ 
 $b_y = \mathbf{p}_y - \mathbf{a}_y$ 
 $c_1 = b_x \cdot {}^p I(\mathbf{a}) + (1 - b_x) \cdot {}^p I(\mathbf{a} + (1, 0, 0)^T)$ 
 $c_2 = b_x \cdot {}^p I(\mathbf{a} + (0, 1, 0)^T) + (1 - b_x) \cdot {}^p I(\mathbf{a} + (1, 1, 0)^T)$ 
 ${}^v I(\mathbf{p}) = b_y \cdot c_1 + (1 - b_y) \cdot c_2$ 

```

end

The method solving this polynomial to find q_1 is referenced with the subroutine *solve12thDegPoly()* in Algorithm 1.

This procedure that combines a pinhole forward and an axial backward projection has to be computed only once and leads to image transformation for undistortion and rectification stored in a lookup table (compare Algorithms 1 and 2 and Fig. 11). The main contribution in the context of the pinax model is of course Algorithm 1, i.e., the way the correction map is created, while Algorithm 2 is just the standard procedure for applying correction maps for rectification, which is included here for the sake of completeness.

3.5. Numerical error analysis

The essential assumption in our model is that the correction computed for points in the pinax plane also generalizes for other points in the scene. Furthermore, we postulate that minor variations in $\hat{d}_0$ are negligible and that typical underwater cameras are already designed in a way that allows to fulfill near optimal conditions. Fig. 12 shows the maximum errors between look up pixel values for the optimal *d_0 and a pinax plane distance of 5 m and scene points that are at different distances than the pinax plane, respectively if in addition d_0 deviates from the optimal *d_0 .

Fig. 12 shows that the errors are very small, i.e., in the order of at most a few millimeters over some meters distances, even with significant deviations of d_0 from *d_0 of up to 40 mm, i.e., under the presence of severe air gaps between the camera and the front panel. Only if the physical camera is significantly placed away from the glass panel pronounced errors occur. If d_0 is quite close *d_0 , i.e., if the air gap is small, the theoretical errors are even negligible considering realistic camera parameters. It can also be noted that the error becomes smaller for larger distances of the scene points.

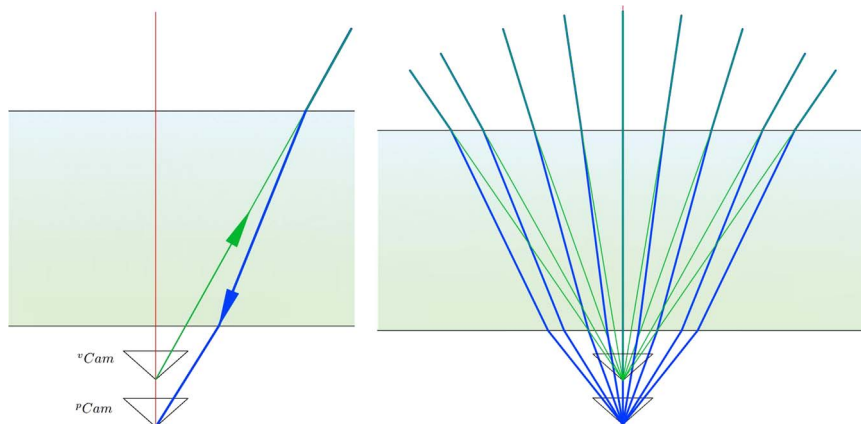


Fig. 11. Left: The map creation in the Pinax model that combines a projection from the virtual pinhole camera to the pinax plane (green ray) and back with an axial projection to the physical camera (blue ray). Right: The virtual (green) rays are good approximations of the physical rays (blue) once they cross from the glass panel into water - and the small d_0 assumption is fulfilled. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

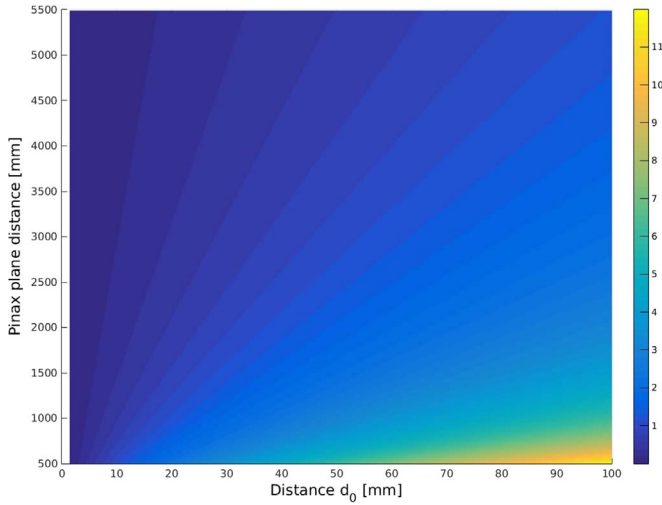


Fig. 12. Errors between the look up pixel value for the optimal d_0 and the pinax plane distance of 5 m and scene points that are at different distances than the pinax plane, respectively for which in addition d_0 deviates from the optimal d_0 . Note that as long d_0 is close to d_0 , the location of the point in the scene has no influence.

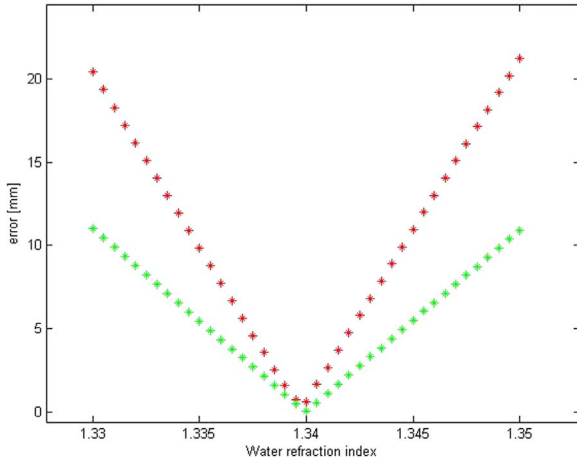


Fig. 13. The maximum (red) and average (green) error caused by changes of the water refraction index in an example camera set-up. The error is measured as the displacement of the estimated light ray from the proper one at a 5 m distance from the camera. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3.6. The role of changes in the water refraction index

The computation of the pinax correction map takes the refraction index of the water into account. It is important to note that even seemingly small changes from a nominal value of $n_w=1.34$ have noticeable effects. Consider our standard example set-up in combination with the water refraction indexes $n_w=1.33$ and $n_w=1.35$, i.e., only about $\pm 0.75\%$ from the nominal value. Fig. 13 shows the errors for the different refraction indexes in this example. The errors are computed as the displacement of the estimated light ray from the proper one at a 5 m distance from the camera. Both the maximum error, which occurs for rays under the maximum incident angle, as well as the average error over all rays are shown. The errors are substantial, i.e., though there is only a very small change in the water refraction index, it is very beneficial to take it into account. This also holds for other methods in general as shown in the experimental results section. While it is sufficient to simply recompute the correction map in the pinax model, which can be done very fast and without the need of gathering any additional vision data, the standard in water calibration approach requires a new recording of in-situ data to avoid errors.

In our experiments presented below, we simply use estimated salinity and the related refraction indexes from tables (Roswell et al., 1976), which we found sufficiently precise to accommodate for the effects of changing water refraction indexes. Nevertheless, the exact water refraction index can also be computed from physical parameters, e.g., by using the formulas from Millard and Seaver (1990) or Quan and Fry (1995). The predominant factor is the influence of salinity followed, to a much lesser extent, by temperature. Very commonly used CTD sensors provide exactly this information, i.e., it is very simple to get an exact indirect measurement of the water refraction index if needed. As the computation of the correction map is relatively fast (see Section 4.1), this even allows an online re-computation of the correction map during the mission if the conditions change, e.g., if the camera on a vehicle operating at sea passes an sweet water inflow, or if a mission ranges from warmer shallow waters to much colder deep waters.

4. Experiments and results

4.1. Overview

In this section, we complement the previous theoretical discussions and numerical analyzes of the pinax model with quantitative evaluations with real underwater camera systems. The underwater systems are based on various components received from different third parties. None of the systems or system components was designed with knowledge of the pinax model. We use stereo cameras here for the only reason that they allow an easier qualitative and quantitative evaluation of the errors in the rectification. The pinax calibration and rectification itself is of course directly applicable to single monocular cameras and does not use any stereo- or multi-view information.

Firstly, run-times for generating and applying the refraction correction maps are presented in Section 4.2. Especially the use of a correction map for rectification is extremely fast and can hence be applied in real-time on a video stream. Secondly, qualitative results from field work where the pinax model is used for in-air calibration are presented in Section 4.3. The qualitative results are based on several third party systems including a custom-made underwater camera on the Ifremer vehicle Vortex Brignone et al. (2011) and several COTS cameras in custom-made underwater housings, e.g., on the AUV Sparus Mallios et al. (2011) from the University of Girona and the AUV Seacat Enchev et al. (2015) from ATLAS Elektronik. Thirdly, quantitative evaluations are presented in Section 4.4 where the pinax model is compared to state-of-the-art underwater calibration. The experiments are conducted with a Bumblebee XB3 with dual stereo, i.e., three monocular cameras at two different baselines, in a custom-made underwater housing and with a GoPro Hero3+ Black Edition stereo rig in a consumer housing from GoPro. The accuracy of underwater stereo computations on artificial checker-board patterns is used in the quantitative evaluations as a metric of rectification accuracy.

4.2. Run-times for generating and applying the refraction correction maps

One of the strengths of the pinax model is its computational efficiency. The refraction correction is done via maps, i.e., simple look-up tables for image rectification which lead to very efficient operations very well suited for real-time performance. The computations of the maps themselves are also relatively fast and can be done just once offline. The following runtimes are benchmarked on an Intel Core i7-3610QM CPU running at 2.3 GHz, i.e., a mobile CPU that is used in an embedded system suited for integration on robotic vehicles, even within the camera system itself.

The experiments are done once with MATLAB R2014a on Windows 7 and once with the Robot Operating System (ROS) Hydro on Ubuntu 12.04. It should be noted that no optimization is used, and especially that no parallelization is employed. Both the computation of the

Table 2

Computation times for generating the correction maps.

Camera	Resolution	Time	
		MATLAB (h:mm:ss)	ROS (mm:ss)
Bumblebee2	1024×768	0:11:47	0:20
Bumblebee XB3/Vortex cam	1280×960	0:18:25	0:32
GoPro Hero3+ Black Ed.	4096×2160	2:12:36	3:50
Fuji FinePix 3DW3	3648×2736	2:29:35	4:20

correction map itself as well as its application for rectification can be easily speeded up by parallel computation, e.g., through multithreaded or CUDA programming if higher processing speeds are required.

Table 2 shows the computation times of the correction maps for different cameras, respectively image resolutions. The computation is linear in the number of pixels and takes about 0.89925 msec/pixel on MATLAB, respectively 0.026042 msec/pixel on ROS. The computation of a pinax correction map has to be done only once. It can hence be simply computed offline.

Each pinax correction map depends - in addition to the in-air calibration map of the underlying physical camera - on the water refraction index, i.e., especially salinity. In the experiments reported later on, only two different correction maps are used across a wide range of different field experiments, namely one for salty water and one for sweet water. As discussed in more detail in the relevant sections, we found two maps to be sufficient. But for even more accurate image rectifications, it is possible to use a CTD sensor to determine the salinity of the water directly at the location of the mission, respectively even during the mission if the salinity changes. This then allows to either instantaneously switch between several pre-computed maps, respectively to even compute a perfectly fitting map online, which takes in the order of at most a few minutes under ROS. The underlying algorithm is also well suited for parallel computation allowing faster processing if needed through multithreading or CUDA programming.

Table 3 shows the computation times to apply the correction maps, i.e., to perform an image rectification, for different cameras, respectively image resolutions. The computation is just a look-up operation and hence very fast and very well suited for real-time operation. The underlying algorithm is again also well suited for parallel computation; hence it is easy to further speed it up if necessary through multithreading or CUDA programming.

4.3. Qualitative results

In this section, we report qualitative results with a range of different underwater camera systems that illustrate the usefulness of the method introduced in this article for real world applications. Table 4 gives an overview of seven different systems where the pinax-model was used on, i.e., the cameras in each system were calibrated just once in air and the pinax correction tables were used for rectification of the images. The correction tables were computed with two different refraction indices, namely $n_w=1.333$ for sweet water, $n_w=1.342$ for salty water, respectively. Depending on the environment conditions, e.g., experi-

Table 3

Computation times for applying the correction maps, i.e., for rectification.

Camera	Resolution	Time	
		MATLAB (seconds)	ROS (seconds)
Bumblebee2	1024×768	0.025	0.007
Bumblebee XB3/ Vortex cam	1280×960	0.055	0.012
GoPro Hero3+ Black Ed.	4096×2160	0.412	0.085
Fuji FinePix 3DW3	3648×2736	0.453	0.094

Table 4

The different camera systems used in qualitative evaluations.

	Camera	Focal length (mm)	Imager		Housing provider
			Resolution	Size	
1	Bumblebee XB3 (Jacobs)	3.8	1280×960	1/3"	U.Zagreb
2	Bumblebee XB3 (IST)	3.8	1280×960	1/3"	U.Zagreb
3	Bumblebee2 (UdG)	2.5	1024×768	1/3"	UdG
4	Bumblebee2 (Jacobs)	2.5	1024×768	1/3"	ATLAS
5	Vortex Camera	4	1280×960	1/3"	Ifremer
6	Fuji FinePix 3DW3	6.3	3648×2736	1/2.3"	FantaSea
7	GoPro Hero3+ Black Ed.	2.65	4096×2160	1/2.3"	GoPro

ments in a pool or lake, or in the sea, the relevant map was chosen. The cameras have different technical parameters, especially with respect to focal length or d_1 , and they are mounted in different housings that were all designed by third parties without any knowledge of the pinax model.

The test systems are all stereo cameras. The advantage of stereo cameras is in this context that they not only provide metric information, which will be used for a quantitative analysis later on, but that their data also provides very good qualitative indicators of the calibration and rectification accuracy. Stereo processing is very sensitive to the accuracy of the image rectification due to the inherent use of the epipolar constraint. If there are distortions in the two cameras, matching pixel blocks do not lie on the same line in the two images anymore, i.e., the epipolar constraint is violated, and correspondences cannot be established leading to missing range values. Hence, rectification errors not only lead to metric errors in the range estimates but also to complete failures in the stereo computations.

The trials with different camera systems in different environment conditions show following main three qualitative results that are interesting for applying the method introduced in this article in real world applications:

1. *In-air calibration of underwater cameras with the pinax model is applicable to a range of systems and environment conditions.* We applied the method to seven different systems used in different environment conditions. The cameras and housings were from various 3rd parties. In each case, in-air calibration with the pinax model was successful and led to (at least) qualitatively comparable results to underwater calibration which was the previous state of practice for the systems.
2. *The quality of the in-air calibration matters.* The pinax-model allows for convenient in-air calibration that only has to be done once. The final result of the rectification is significantly influenced by the quality of this calibration.
3. *The water refraction index, especially due to salinity, matters but to a lesser extent.* Ignoring the influence of the changes in refraction of water due to environmental parameters, especially in form of salinity, leads to degradation in accuracy in the rectification.

Regarding aspect 1., the pinax model was successfully used on all seven systems. The in-air calibration and the related image rectification lead to high quality results in all cases as indicated by the density of the 3D point clouds generated by the stereo processing. Figs. 14 and 15 show two typical results as illustrative examples. It should be noted that the "holes" in the point cloud shown in the center of Fig. 14 are visible just due to the perspective view, i.e., due to occlusions in the scene. In addition to the density of the stereo results, there are also qualitative indications of the metric accuracy.

Fig. 16 illustrates the aspects 2. and 3. with respect to the relevance of the quality of the in-air calibration, depending on the water

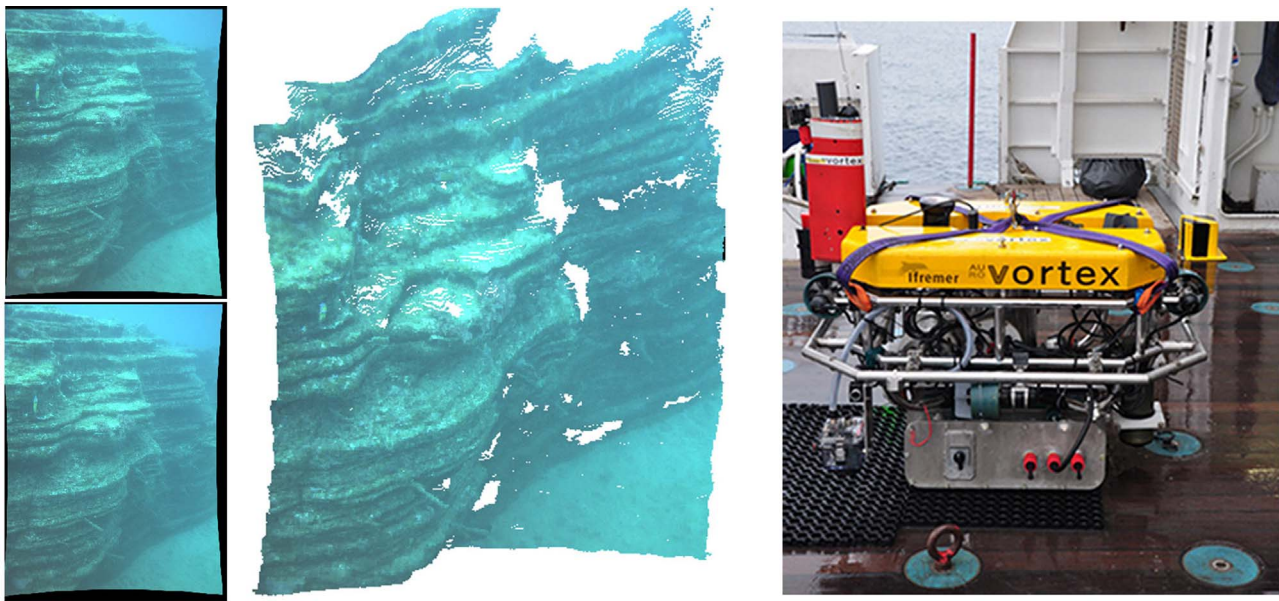


Fig. 14. A 2.5D colored point cloud (center) generated from images (left) from a custom-made underwater stereo camera on the Vortex vehicle (right) of the Institut français de recherche pour l'exploitation de la mer (Ifremer). The stereo point cloud is very dense, hence indicating very good rectification accuracy. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

refraction index. The point cloud PC_c shown on the right was generated in seawater by system 2 (Bumblebee XB3 (IST) with U.Zagreb housing) using our method with the proper factory in-air calibration file as input and our standard salt-water refraction estimation. The resulting point cloud density ∇PC_c provides a comparison baseline for a simple illustrative example.

The point cloud PC_a shown on the left uses the factory in-air calibration file from exactly the same type of camera, namely the Bumblebee XB3 owned by Jacobs with identical (nominal) parameters as the one owned by IST, and which is mounted in the same type of housing, namely the design by U.Zagreb. The proper salt-water refraction index is used. Nevertheless, the point cloud density ∇PC_a is just 19.7% of the density ∇PC_c . So, no correspondences can be found for a significant portion of the pixels in both images, i.e., the necessary epipolar constraint for stereo vision does not hold, and so the rectification process is highly unsuccessful in this case.

The point cloud PC_b shown in the center uses the correct factory in-air calibration file of this specific camera instance. But our standard sweet-water refraction index is here used in the pinax model though the data is collected in seawater. The point cloud density ∇PC_b degrades therefore to 93.2% of the density ∇PC_c in this example. It can be noticed that there is especially missing data at the sides of the point cloud, which is consistent with what is to be expected when the rectification quality degrades. The distortion effects due to refraction are most visible at the sides of the stereo images, hence violations of the epipolar constraint due to degraded rectification start taking effect from there.

4.4. Quantitative evaluation of the pinax accuracy

The numerical analyses of the pinax model as well as the qualitative experiences in the field indicate that it leads to very accurate calibration and rectification results. This is now further substantiated with quantitative evaluations of real cameras, namely a Bumblebee XB3 (Table 4, system 1) and a stereo rig consisting of two GoPro Hero3+ Black Edition (Table 4, system 7). Both systems are quite different and provide two interesting test cases.

The Bumblebee XB3 features three monocular cameras. This allows stereo processing with a short and with a wide baseline (Fig. 17). The GoPro stereo system consist of a standard set-up with two cameras (Fig. 18). There are hence five monocular cameras in total that are calibrated and rectified with the pinax model in the following experiments.

As it is difficult or even impossible to acquire ground truth data of natural underwater environments, the analysis is based on artificial checkerboard patterns where the exact distance between the black and the white fields is known. For the quantitative evaluations, the stereo systems are placed in a pool filled with sweet or salt water. The checkerboard pattern is then moved at different distances within the field of view of each camera. Stereo processing is conducted for each sequence of images acquired at the different distances. The metric stereo estimates of the distances between the checkerboard markers are finally compared to the ground truth distances, thus providing an error metric for the rectification accuracy.

Four different methods for calibration and rectification are evaluated, namely:



Fig. 15. A 3D map generated from 350 stereo based point clouds from a Bumblebee camera on a Sparus vehicle of the University of Girona. The consistently dense stereo data in all stereo computations indicates high rectification accuracy.

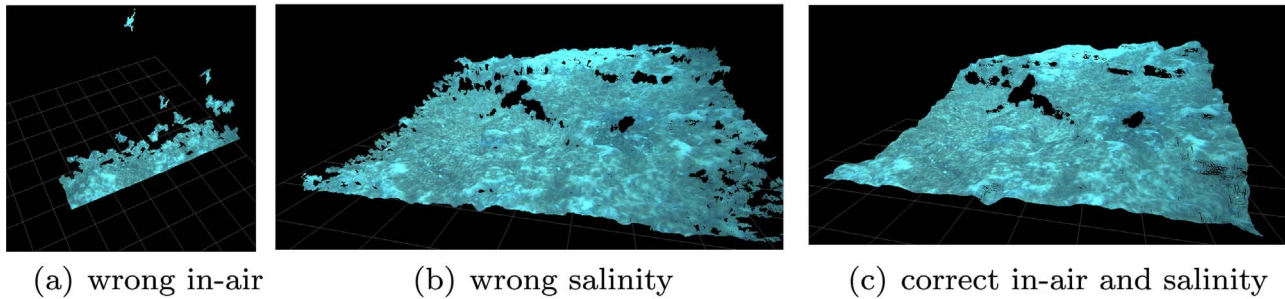


Fig. 16. An illustration of two of the three main qualitative observations related to our method, namely that 2. the quality of the in-air calibration matters (a) compared to c) and that 3. the salinity has an influence (b) compared to c)).

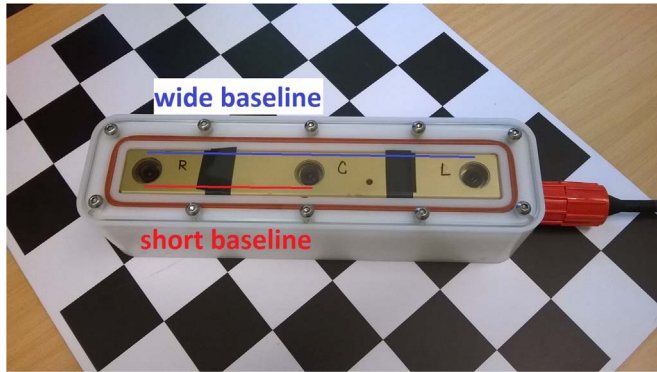


Fig. 17. The Point Grey Bumblebee XB3 has three monocular cameras that allow stereo processing with a short and with a wide baseline. The checkerboard pattern underneath the camera is used for the quantitative accuracy analysis.

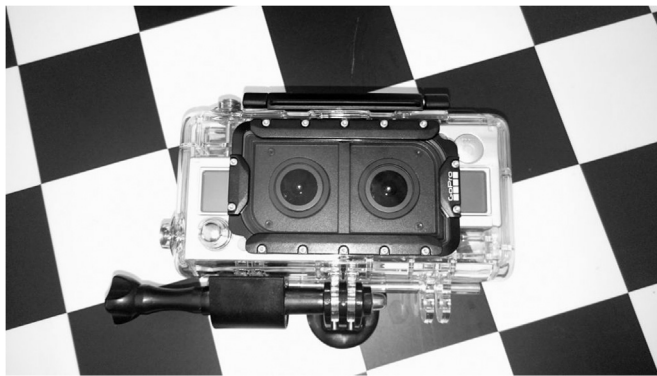


Fig. 18. The stereo system consisting of two GoPro Hero3+ Black Edition cameras in a GoPro Dual HERO underwater housing.

- standard *in-air* calibration and rectification
- state-of-the-art *underwater* calibration with a *correct* water refraction index (WRI), i.e., the calibration is performed in-situ in water at exactly the same salinity conditions as the recording of the evaluation data that is then rectified
- state-of-the-art *underwater* calibration with a *wrong* WRI, i.e., the calibration is performed in a sweet water pool while the recording of the rectified evaluation data is done in salty water
- *pinax* in-air calibration and rectification under arbitrary but roughly known (sweet or salty) water conditions

All calibrations are based on the popular SVP model by Zhang (2000) that can be found for example in MATLAB. We use here the according method from the CamOdoCal calibration package (Heng et al., 2013, 2014, 2015).

We can unfortunately not directly compare our results with the method of Agrawal et al. (2012) or methods that build on it, i.e., that

use the proper axial camera model to determine d_0 like for example Kawahara et al. (2013). The problem is that its non-linear optimization suffers from numerical instability when d_0 becomes small, i.e., when d_0 is in the order of several millimeters. More precisely, the underlying numerical optimization still performs well in these cases with perfect calibration data. However it diverges as soon as there is noise on the calibration data, especially when using any real world calibration data. It is interesting to note that the real world experiments presented in Agrawal et al. (2012) or in publications building on it use quite large values for d_0 , i.e., the cameras are placed at quite some distance, like even half a meter, away from the glass panel.

So in the region of interest for our method, i.e., with the camera being close to the window and d_0 being in the order of a few millimeters, the exact axial camera model fails. To substantiate this observation, we have conducted experiments with simulated data for different values of d_0 in our region of interest, namely for $d_0 = 1.5$ mm and $d_0 = 15$ mm. Simulated checkerboard points are projected from 3D to the image plane using the 12th degree polynomial for a proper axial camera model. Six levels of Gaussian noise from 0 to 0.5 pixel standard deviation are added to the projection. These points are used as input to the method from Agrawal et al. (2012) using their released code. 15 runs are performed for each noise level for each value of d_0 . The results of each run are shown in Fig. 19 as relative error of the result of the non-linear optimization over the ground truth. It can be observed that the method only leads to stable, correct results in the absolutely noise-free cases. For $d_0 = 1.5$ mm, the output varies significantly over each run and produces significant errors in the estimate of d_0 as soon as there is the slightest noise present. It can be observed that the situation improves a bit for $d_0 = 15$ mm, but a substantial instability still remains.

Fig. 20 shows the results of the evaluations on the three different stereo set-ups. All errors are normalized, i.e., they are plotted as percentage of the known, measured distance between checkerboard markers. For the GoPro test sequence, the evaluation of the in-air calibration is omitted as its rectification is performing so poorly that stereo processing is not possible anymore (Fig. 21). Also in the case of the two Bumblebee XB3 set-ups, stereo processing for several of the recorded image pairs could not be performed due to poor rectification results with the in-air calibration. These cases would have accordingly lead to significant metric errors; the reported average errors for the in-air calibration are hence a very optimistic, best case estimates.

It can be seen that this quantitative evaluation supports the previous numerical and qualitative observations. The pinax calibration and rectification leads in all cases to superior results. Most importantly, the errors are significantly smaller than using the state-of-the-art underwater calibration. In addition, pinax calibration is much more convenient to use as it is based on in-air calibration. The experiments also show that the salinity matters, i.e., if state-of-the-art underwater calibration is done for example in a sweet water pool and the camera is used in the sea, the rectification quality degrades. The pinax model takes the possible changes of the water salinity into account and is hence not affected by this.

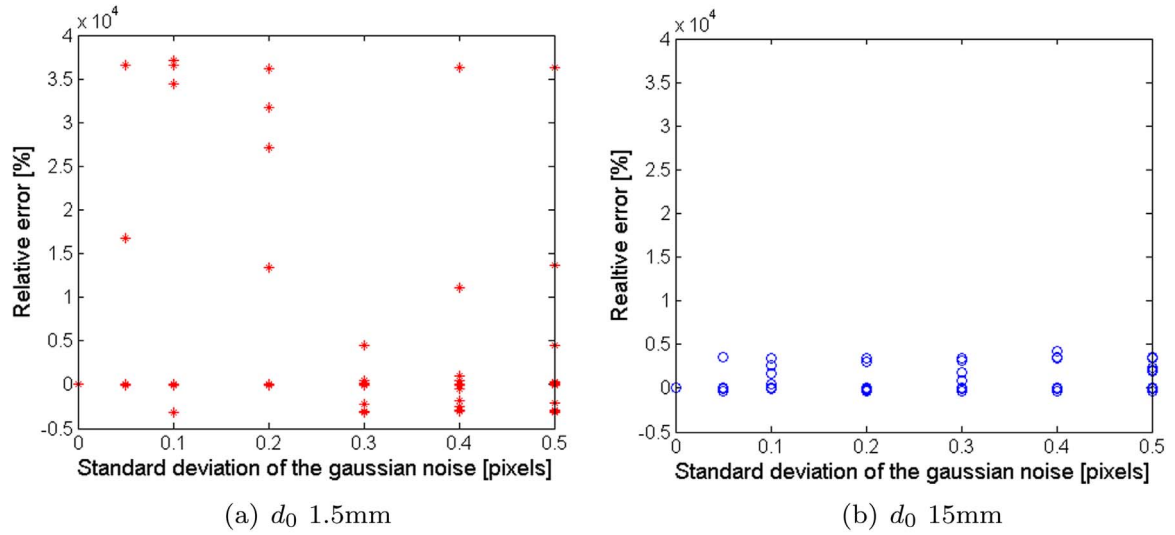


Fig. 19. The method from Agrawal et al. (2012) becomes numerically unstable for small d_0 under the presence of noise. This is illustrated here with two datasets for $d_0 = 1.5$ mm and $d_0 = 15$ mm. In each case, six noise levels from 0 to 0.5 pixel standard deviation are investigated. For each noise level, 15 runs of the method are performed on simulated data, i.e., with known ground truth d_0 .

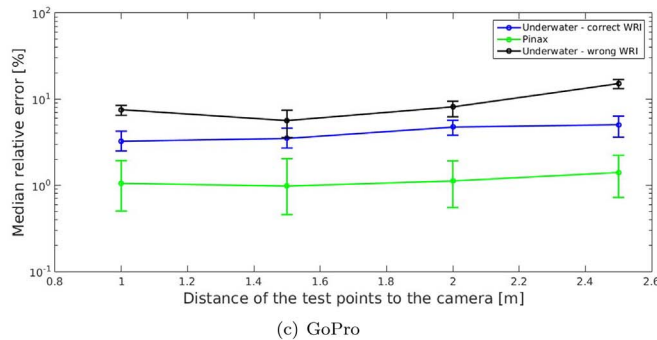
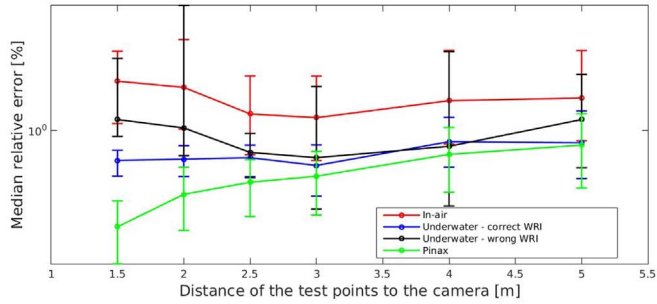
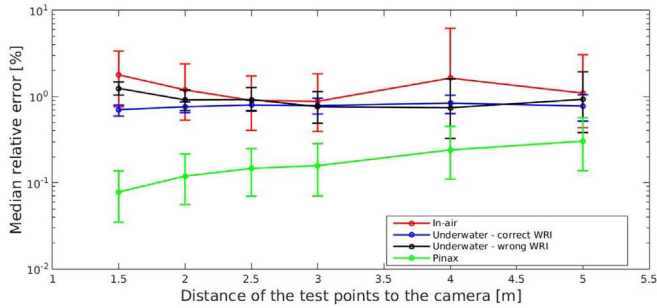


Fig. 20. The relative errors of the triangulated points in % for the four evaluated methods on the three different stereo set-ups. The error bars show the upper and lower quartile values of the error, the center dots are the medians. The error values are plotted for each nominal distance of the calibration pattern during the test.

One might expect from a purely theoretical viewpoint that in-air calibration with pinax and a state-of-the-art underwater calibration that is performed in-situ under the correct water refraction index (underwater with correct WRI) lead to similar, if not identical results. But it is important to note that underwater visibility is worse than visibility in air. This holds even in the best possible cases, i.e., over short distances in a pool. Given the same conditions, i.e., the same number of image frames for calibration, etc., the underwater calibration data is hence noisier due to blur affecting the corner detection. Pinax is hence not only more convenient to use, it also leads to more accurate calibration results in real world applications.

5. Conclusions

In this article, the pinax model for calibration and rectification of underwater cameras in flat-plane housings was introduced. Its water/window refraction correction was derived from an analysis of the axial camera model for underwater cameras, which is among others computationally hard to tackle. Therefore, realistic constraints on the distance of the camera to the window are used in the pinax model to combine aspects of a virtual pinhole model with the projection function from the axial camera model. It was shown how the pinax model allows the pre-computation of a lookup-table for very fast refraction correction of the flat-plane with high accuracy. The pinax model is very convenient to use in practice as it is based on single, in-air calibration of the complete underwater camera system. The model takes the water refraction index into account for which a rough estimate of the salinity (sweet/salty) is sufficient, and for which in-situ measurements from e.g. a CTD-sensor can also be used to derive perfectly fitting correction maps. The pinax model was analyzed in this article in three different ways. Firstly, numerical simulations were used to illustrate the influences of the different parameters on the accuracy and to motivate the model. Secondly, qualitative results from use-cases of the model in the field were presented and they illustrated that it worked fine with various real world systems designed by various third parties. Thirdly, a quantitative analysis on different stereo systems was performed where the pinax model lead a higher accuracy than state of the art underwater calibration.

Acknowledgements

The research leading to the presented results was supported in part

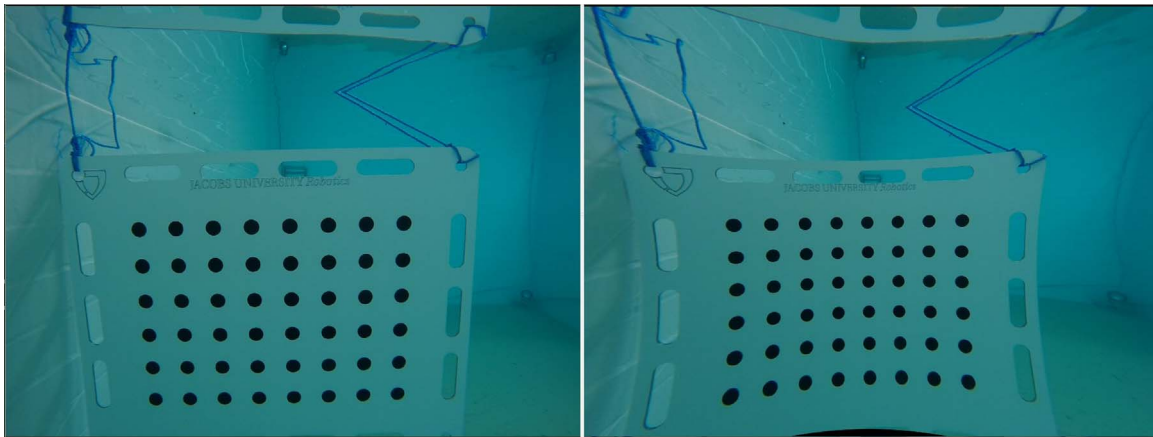


Fig. 21. For the GoPro data, the in-air calibration leads to such severe distortions that the stereo processing is completely failing in this case. On the left, an example GoPro image from a test sequence is shown; on the right, the “corrected” image based on in-air calibration is shown.

by the European Community's Seventh Framework Programme under grant agreements: n. 288704 “Marine robotic system of self-organizing, logically linked physical nodes (MORPH)” and n. 611373 “Cognitive Autonomous Diving Buddy (CADDY)” as well as under grant agreement n. 635491 “Effective Dexterous ROV Operations in Presence of Communications Latencies (DexROV)” in the European Community's Horizon 2020 Framework Programme.

The authors thank Nikola Miskovic and his team from the University of Zagreb for providing the underwater camera housing used in the experiments with a BumblebeeXB3 as part of the quantitative analysis of our method as well as in qualitative trials of the method on several vehicles in different application scenarios under varying environment conditions. We also thank in this context the University of Zagreb for tests of the system on their AUV “Buddy” and the Consiglio Nazionale delle Ricerche (CNR) for trials on the ROV “Artu”. We also thank ATLAS Elektronik for providing the underwater housing for a Bumblebee2 camera and the use of the robot Seacat for recording test data. Furthermore, we thank the University of Girona and the Instituto Superior Tecnico for data from their vehicles Girona500, respectively Medusa with different version of the BumblebeeXB3 in terms of factory settings and housing designs. Finally, we thank the Institut français de recherche pour l'exploitation de la mer (Ifremer) for test data from a custom made camera and housing on the ROV Vortex.

References

- Agrawal, A., Ramalingam, S., Taguchi, Y., Chari, V., 2012. A theory of multi-layer flat refractive geometry. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3346–3353. <http://dx.doi.org/10.1109/CVPR.2012.6248073>.
- Asakawa, K., Kojima, J., Kato, Y., Matsumoto, S., Kato, N., 2000. Autonomous underwater vehicle aqua explorer 2 for inspection of underwater cables. In: Underwater Technology, 2000. UT 00. Proceedings of the 2000 International Symposium on. pp. 242–247. <http://dx.doi.org/10.1109/UT.2000.852551>.
- Beall, C., Dellaert, F., Mahon, I., Williams, S.B., 2011. Bundle adjustment in large-scale 3d reconstructions based on underwater robotic surveys. In: OCEANS, IEEE - Spain, pp. 1–6.
- Bingham, B., Foley, B., Singh, H., Camilli, R., Delaporta, K., Eustice, R., Mallios, A., Mindell, D., Roman, C., Sakellariou, D., 2010. Robotic tools for deep water archaeology: surveying an ancient shipwreck with an autonomous underwater vehicle. *J. Field Robot.* 27, 702–717. (TY - JOUR).
- Bodenmann, A., Thornton, B., Nakajima, R., Yamamoto, H., Ura, T., 2013. Wide area 3d seafloor reconstruction and its application to sea fauna density mapping. In: Oceans - San Diego, pp. 1–5.
- Brandou, V., Allais, A., Perrier, M., Malis, E., Rives, P., Sarrazin, J., Sarrazin, P., 2007. 3d reconstruction of natural underwater scenes using the stereovision system iris. In: OCEANS 2007 - Europe. pp. 1–6. <http://dx.doi.org/10.1109/OCEANSE.2007.4302315>.
- Brignon, L., Munaro, M., Allais, A., Opderbecke, J., 2011. First sea trials of a laser aided three dimensional underwater image mosaicing technique. In: OCEANS. IEEE, pp. 1–7.
- Chapman, P., Bale, K., Drap, P., 2010. We all live in a virtual submarine. *Comput. Graph. Appl.* IEEE 30, 85–89.
- Chari, V., Sturm, P.F., 2009. Multi-view geometry of the refractive plane. In: BMVC. British Machine Vision Association.
- Chen, X., Yang, Y.-H., 2014. Two-view camera housing parameters calibration for multi-layer flat refractive interface. In: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on. pp. 524–531. <http://dx.doi.org/10.1109/CVPR.2014.74>.
- Davie, A., Hartmann, K., Timms, G., de Groot, M., McCulloch, J., 2008. Benthic habitat mapping with autonomous underwater vehicles. In: OCEANS 2008. pp. 1–9. <http://dx.doi.org/10.1109/OCEANS.2008.5151927>.
- Enchev, I., Pfingsthorn, M., Luczynski, T., Sokolovski, I., Birk, A., Tietjen, D., 2015. Underwater place recognition in noisy stereo data using fab-map with a multimodal vocabulary from 2d texture and 3d surface descriptors. In: IEEE Oceans.
- Foresti, G., 2001. Visual inspection of sea bottom structures by an autonomous underwater vehicle. *IEEE Trans. Syst., Man, Cybern. Part B Cybern.* 31, 691–705. <http://dx.doi.org/10.1109/3477.956031>.
- Galceran, E., Campos, R., Palomeras, N., Carreras, M., Ridao, P., 2014. Coverage path planning with realtime replanning for inspection of 3d underwater structures. In: Robotics and Automation (ICRA), 2014 IEEE International Conference on. pp. 6586–6591. <http://dx.doi.org/10.1109/ICRA.2014.6907831>.
- Gedge, J., Gong, M., Yang, Y.-H., 2011. Refractive epipolar geometry for underwater stereo matching. In: Computer and Robot Vision (CRV), 2011 Canadian Conference on. pp. 146–152. <http://dx.doi.org/10.1109/CRV.2011.26>.
- Gracias, N., Santos-Victor, J., 2000. Underwater video mosaics as visual navigation maps. *Comput. Vis. Image Underst.* 79, 66–91.
- Hartley, R., Zisserman, A., 2003. Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge.
- Heng, L., Burki, M., Lee, G.H., Furgale, P., Siegart, R., Pollefeys, M., 2014. Infrastructure-based calibration of a multi-camera rig. In: Robotics and Automation (ICRA), 2014 IEEE International Conference on. pp. 4912–4919. <http://dx.doi.org/10.1109/ICRA.2014.6907579>.
- Heng, L., Furgale, P.T., Pollefeys, M., 2015. Leveraging image-based localization for infrastructure-based calibration of a multi-camera rig. *J. Field Robot.* 32, 775–802. (URL (<http://dblp.uni-trier.de/db/journals/jfr/jfr32.html#HengFP15>)).
- Heng, L., Li, B., Pollefeys, M., 2013. Camodocal: Automatic intrinsic and extrinsic calibration of a rig with multiple generic cameras and odometry. In: Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on. pp. 1793–1800. <http://dx.doi.org/10.1109/IROS.2013.6696592>.
- Hollinger, G., Englot, B., Hover, F., Mitra, U., Sukhatme, G., 2012. Uncertainty-driven view planning for underwater inspection. In: Robotics and Automation (ICRA), 2012 IEEE International Conference on. pp. 4884–4891. <http://dx.doi.org/10.1109/ICRA.2012.6224726>.
- Horgan, J., Toal, D., 2006. Review of machine vision applications in unmanned underwater vehicles. In: Control, Automation, Robotics and Vision, 2006. ICARCV '06. In: Proceedings of the 9th International Conference on. pp. 1–6.
- Hue, J., Serayet, M., Drap, P., Papini, O., Wuerbel, E., 2011. Underwater archaeological 3d surveys validation within the removed sets framework symbolic and quantitative approaches to reasoning with uncertainty. *Lecture Notes in Computer Science*, vol. 6717. Springer, Berlin/Heidelberg, pp. 663–674.
- Johnson-Roberson, M., Pizarro, O., Williams, S.B., Mahon, I., 2010. Generation and visualization of large-scale three-dimensional reconstructions from underwater robotic surveys. *J. Field Robot.* 27, 21–51.
- Jordt, A., Köser, K., Koch, R., 2016. Refractive 3d reconstruction on underwater images. *Methods Oceanogr.* 15–16, 90–113. <http://dx.doi.org/10.1016/j.mio.2016.03.001>, (URL (<http://www.sciencedirect.com/science/article/pii/S2211122015300086>)). Computer Vision in Oceanography).
- Jordt-Sedlazeck, A., Jung, D., Koch, R., 2013. Refractive plane sweep for underwater images. In: Weickert, J., Hein, M., Schiele, B. (Eds.), Pattern Recognition 35th German Conference, GCPR 2013, Saarbrücken, Germany, September 3–6, 2013. Proceedings. Springer Berlin Heidelberg, Berlin, Heidelberg, 333–342. <http://>

- dx.doi.org/10.1007/978-3-642-40602-7_36, (URL (http://dx.doi.org/10.1007/978-3-642-40602-7_36)).
- Jordt-Sedlazeck, A., Koch, R., 2012. Refractive calibration of underwater cameras. In: Proceedings of the 12th European Conference on Computer Vision - Volume Part V ECCV'12, pp. 846–859.
- Jordt-Sedlazeck, A., Koch, R., 2013. Refractive structure-from-motion on underwater images. In: Computer Vision (ICCV), 2013 IEEE International Conference on. pp. 57–64. <http://dx.doi.org/10.1109/ICCV.2013.14>.
- Kang, L., Wu, L., Yang, Y.-H., 2012. Experimental study of the influence of refraction on underwater three-dimensional reconstruction using the svp camera model. *Appl. Opt.* 51, 7591–7603.
- Kawahara, R., Nobuhara, S., Matsuyama, T., 2013. A pixel-wise varifocal camera model for efficient forward projection and linear extrinsic calibration of underwater cameras with flat housings. In: Proceedings of ICCV2013 Workshop on Underwater Vision. pp. 819–824.
- Kim, A., Eustice, R., 2013. Real-time visual slam for autonomous underwater hull inspection using visual saliency. *IEEE Trans. Robot.* 29, 719–733. <http://dx.doi.org/10.1109/TRO.2012.2235699>.
- Kunz, C., Singh, H., 2008. Hemispherical refraction and camera calibration in underwater vision. In: OCEANS 2008. pp. 1–7. <http://dx.doi.org/10.1109/OCEANS.2008.5151967>.
- Kunz, C., Singh, H., 2010. Stereo self-calibration for seafloor mapping using auvs. In: Autonomous Underwater Vehicles (AUV), 2010 IEEE/OES. pp. 1–7. <http://dx.doi.org/10.1109/AUV.2010.5779655>.
- Lavest, J.-M., Rives, G., Lapresté, J.-T., 2003. Dry camera calibration for underwater applications. *Mach. Vis. Appl.* 13, 245–253.
- Li, R., Li, H., Zou, W., Smith, R., Curran, T., 1997. Quantitative photogrammetric analysis of digital underwater video imagery. *IEEE J. Ocean. Eng.* 22, 364–375. <http://dx.doi.org/10.1109/48.585955>.
- Lots, J.F., Lane, D.M., Trucco, E., 2000. Application of 2 1/2 d visual servoing to underwater vehicle station-keeping. In: OCEANS 2000 MTS/IEEE Conference and Exhibition, vol. 2. pp. 1257–1264.
- Mallios, A., Rida, P., Carreras, M., Hernandez, E., 2011. Navigating and mapping with the sparus auv in a natural and unstructured underwater environment. In: OCEANS 2011. pp. 1–7.
- Marks, R.L., Wang, H.H., Lee, M.J., Rock, S.M., 1994. Automatic visual station keeping of an underwater robot. In: OCEANS, vol. 2. pp. II/137–II/142. IEEE.
- McLeod, D., Jacobson, J., Hardy, M., Embry, C., 2013. Autonomous inspection using an underwater 3d lidar. In: Oceans - San Diego, 2013. pp. 1–8.
- Millard, R.C., Seaver, G., 1990. An index of refraction algorithm for seawater over temperature, pressure, salinity, density, and wavelength. *Deep Sea Res. Part A. Oceanogr. Res. Pap.* 37, 1909–1926.
- Negahdaripour, S., Barufaldi, C., Khamene, A., 2006. Integrated system for robust 6-dof positioning utilizing new closed-form visual motion estimation methods in planar terrains. *IEEE J. Ocean. Eng.* 31, 533–550.
- Negahdaripour, S., Firoozfard, P., 2006. An rov stereovision system for ship-hull inspection. *IEEE J. Ocean. Eng.* 31, 551–564. <http://dx.doi.org/10.1109/JOE.2005.851391>.
- Negahdaripour, S., Fox, J., 1991. Undersea optical stationkeeping: improved methods. *J. Robot. Syst.* 8, 319–338.
- Negahdaripour, S., Sekkati, H., Pirsiavash, H., 2007. Opti-acoustic stereo imaging, system calibration and 3-d reconstruction. In: Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on. pp. 1–8.
- Pessel, N., Opderbecke, J., Aldon, M.-J., 2003. Camera self-calibration in underwater environment. In: Proceedings of the 11th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision.
- Pizarro, O., Eustice, R., Singh, H., 2003. Relative pose estimation for instrumented, calibrated imaging platforms. In: Proceedings of Digital Image Computing Techniques and Applications. pp. 601–612.
- Purcell, M., Gallo, D., Packard, G., Dennett, M., Rothenbeck, M., Sherrell, A., Pascaud, S., 2011. Use of remus 6000 auvs in the search for the air france flight 447. In: OCEANS 2011. pp. 1–7.
- Quan, X., Fry, E.S., 1995. Empirical equation for the index of refraction of seawater. *Appl. Opt.* 34, 3477–3480.
- Roswell, W., Halikas, A., Halikas, G., 1976. The index of refraction of seawater - technical report. University of California, San Diego Visibility Laboratory.
- Sedlazeck, A., Koch, R., 2011. Calibration of housing parameters for underwater stereo-camera rigs. In: Hoey, J. (Ed.), Proceedings of the British Machine Vision Conference. BMVA Press. <http://dx.doi.org/10.5244/C.25.118>.
- Sedlazeck, A., Köser, K., Koch, R., 2009. 3d reconstruction based on underwater video from rov kiel 6000 considering underwater imaging conditions. In: OCEANS 2009 - EUROPE. pp. 1–10. <http://dx.doi.org/10.1109/OCEANSE.2009.5278305>.
- Servos, J., Smart, M., Waslander, S., 2013. Underwater stereo slam with refraction correction. In: Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on. pp. 3350–3355. <http://dx.doi.org/10.1109/IROS.2013.6696833>.
- Shortis, M.R., Harvey, E.S., 1998. Design and calibration of an underwater stereo-video system for the monitoring of marine fauna populations. *Int. Arch. Photogramm. Remote Sens.* 32.
- Treibitz, T., Schechner, Y., Kunz, C., Singh, H., 2012. Flat refractive geometry. *IEEE Trans. Pattern Anal. Mach. Intell.* 34, 51–65. <http://dx.doi.org/10.1109/TPAMI.2011.105>.
- Treibitz, T., Schechner, Y. Y., Singh, H., 2008. Flat refractive geometry. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. pp. 1–8.
- Yamashita, A., Kawanishi, R., Koketsu, T., Kaneko, T., Asama, H., 2011. Underwater sensing with omni-directional stereo camera. In: Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on. pp. 304–311. <http://dx.doi.org/10.1109/ICCVW.2011.6130257>.
- Yau, T., Gong, M., Yang, Y.-H., 2013. Underwater camera calibration using wavelength triangulation. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. pp. 2499–2506.
- Zhang, Z., 2000. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* 22, 1330–1334.