Short-Range Sensor for Underwater Robot Navigation using Line-lasers and Vision

Nicholas Hansen * **Mikkel C. Nielsen** *,**
David Johan Christensen * **Mogens Blanke** *,**

* *Technical University Denmark, Lyngby, Denmark*
(e-mail: {pnha, miconi, djchr, mb}@elektro.dtu.dk).
** *Center for Autonomous Marine Operations and Systems (AMOS), Norwegian University of Science and Technology, Trondheim, Norway.*

Abstract: This paper investigates a minimalistic laser-based range sensor, used for underwater inspection by Autonomous Underwater Vehicles (AUV). This range detection system system comprise two lasers projecting vertical lines, parallel to a camera's viewing axis, into the environment. Using both lasers for distance estimation, the sensor offers three dimensional interpretation of the environment. This is obtained by triangulation of points extracted from the image using the Hough Transform. We evaluate the system in simulation and by physical proof-of-concept experiments on an OpenROV platform.

Keywords: Autonomous Underwater Vehicle, Robot Vision, Laser Based Vision, Short-Range Sensor, Underwater Navigation

1. INTRODUCTION

Offshore installations drive an evolution in maritime technology to handle the increasing tasks of inspections and quality control. These inspection tasks has to be performed under increasingly harsh conditions, as offshore installations move to deeper and rougher waters, thus requiring even more man-hours and expensive equipment. One strategy to potentially reduce the cost of upkeep, and imporve production downtime is to use Authonomous Underwater Vehicles (AUVs) to perform the inspection tasks autonomously. Currently, a common way of inspecting the these offshore installations consist of deploying ships onsite, equipped with either a Remotely Operated Vehicle (ROV) or an AUV system to perform the inspection. Both types of system have drawbacks, e.g. inspecting confined underwater structures by ROV is often time consuming and includes risk of entanglement of the tether-cable. Most inspection tasks can be solved by an AUV without the intervention of an operator, however the current standard of AUV is typically a rather large vessel, rendering it unable to navigate smaller confined spaces.

Ideally a complete robotic system for inspection would consist of a fault-tolerant multi-robot system, capable of handling longterm offshore inspection tasks, of man-made confined structures, autonomously. For this solution to be viable, the robots must be able to handle long time submersion and exposure to the environment, thus the robots would need a base station to house and charge them. Inspection tasks would be conducted by small robots that can navigate around and/or inside the structures. The robots would upon a completed inspection task send inspection data back to the engineering team on shore.

Techniques for navigation in small unknown underwater environments can largely be categorized as either acoustic or vision based. Acoustic aided navigation such as Long Base Line (LBL) or Ultra Short Base Line (USBL) systems use the known position of acoustic transponders to navigate, whereas the Doppler Velocity Log (DVL) provide a velocity estimate with respect to the seafloor (Kinsey et al., 2006). Common for the acoustic based systems, is the complex dependency of water temperature, salinity, pressure and environmental structure, along with a relative high cost (Chandrasekhar et al., 2006).

Several authors have investigated vision-based navigation. This included (Bruno et al., 2011) who demonstrated a successful method for 3-dimensional close range reconstruction of objects using stereo vision and structured light while (Carreras et al., 2003) and (Garcia et al., 2003) showed navigation and localization using data extracted from a camera image of a known mosaic pattern located on the floor of a pool. Work describing the use of lasers for underwater range estimation have been described in several papers. In (Fournier, 1993) the authors proposed to use lasers to enhance the picture quality of a underwater video camera instead of calculating the distances to objects in the images, the goal was to increase the quality of the image allowing the operator to perform object detection at greater distance. A system using triangulation between a laser dot and a camera frame was described in (Chen and Lee, 2000) the results showed sub-millimeter accuracy between 450mm and 1200mm. The triangulation is expanded to two lasers in (Kondo and Ura, 2004), where the authors exploit the added information to do navigation with respect to different objects. In this case the lasers are mounted parallel to each other. Navigation based on Laser Vision is investigated by (Karras and Kyriakopoulos, 2007). The laser vision data is then fused with IMU data in a Kalman filter to bound the drift of the navigation estimate.

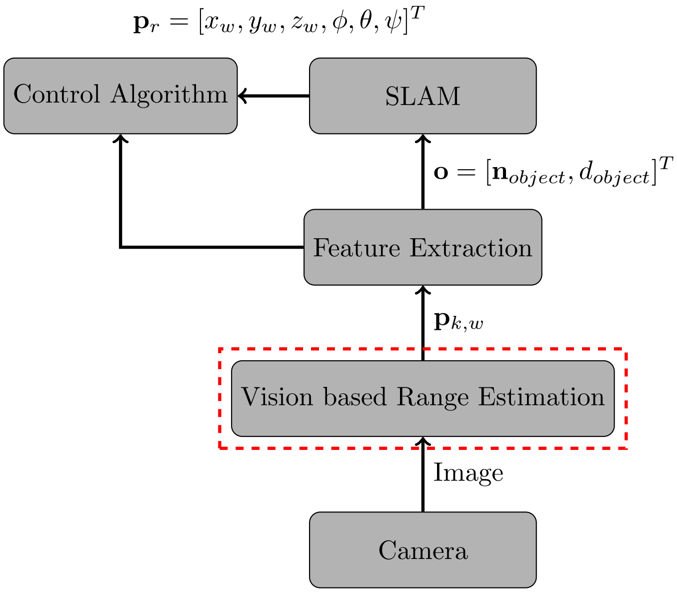

$\mathbf{p}_r = [x_w, y_w, z_w, \phi, \theta, \psi]^T$
Control Algorithm
SLAM
$\mathbf{o} = [\mathbf{n}_{object}, d_{object}]^T$
Feature Extraction
$\mathbf{p}_{k,w}$
Vision based Range Estimation
Image
Camera

Fig. 1. Overview of the robotic control architecture

A distance measurement solution utilizing imaging of two parallel laser lines with a CCD camera was presented in (Cain and Leonessa, 2012) for 2 dimensional range estimation. This proposed solution consisted of calculations using 2D triangulation of the lasers and thereby having the benefit of a rather low computational complexity. A prerequisite for this approach to function is highly accurate mounting of the parallel lasers and accurate angle alignment between both of the lasers and of the camera. The main benefit of this *active sensor* approach is the simplicity of the concept, the low cost of components and the low overall complexity of the sensor package.

This paper will build on the idea of utilizing imaging of two laser lines to obtain under water range measurements. The solution that will be presented extends previous results by extending the line-laser based vision solution such that it will provide 3-dimensional range estimation. The results of this work is a 3-dimensional point cloud of the surroundings. A benefit of the 3-dimensional point cloud is the ability to extract a multitude of features, e.g. planes, circular objects etc. for e.g. Simultaneous Localization and Mapping (SLAM) filter.
The solution presented will be implemented and tested on the OpenROV platform. OpenROV is a small, open source ROV platform, with a small form factor and low cost build materials[1] The OpenROV comes equipped with an on-board computer running Linux, thrusters and sensors, including a camera.

Fig. 1 show the part of the system (encased by a dashed box) being treated in this paper.

2. METHOD

The following section will describe the method used for range estimation.

2.1 Reference frames

When working with ocean vessels, including AUVs and ROVs, several reference frames are used for describing the motions and environments of said vessels. The *North-East-Down* (NED) reference frame $\{n\} = [x_n, y_n, z_n]^T$ is an Earth-Fixed tangent plane on the surface of the earth, with the x-axis always pointing north, y-axis pointing east and z-axis pointing down towards the earth center (Fossen, 2011).
The body-fixed reference frame $\{b\} = [x_b, y_b, z_b]^T$ is a moving coordinate frame that has the origin and axes fixed to the vessel, such that the x-axis is going from aft to fore of the vessel, the y-axis going from port to starboard, and the z-axis is pointing down (Fossen, 2011).
The camera base reference frame $\{c\} = [x_c, y_c, z_c]^T$ is a camera-fixed frame with origin $\mathcal{O}_c$ located at the aperture of the camera. The z_c-axis is aligned with the plane normal of the image plane, thereby describing the distance away form the camera, while x_c and y_c-axis are parallel to image frame such that y_c is directed downwards in the image plane and the x_c-axis point from the center towards the right in the image frame hence completing the right hand rule. Notational a point $\mathbf{p}_k$ in NED-frame will be denoted $\mathbf{p}_{k,n}$, while in the body-frame it is denoted as $\mathbf{p}_{k,b}$.

2.2 Pinhole Model

A pinhole model is used for the projection of a single point $\mathbf{p}_k$ in NED coordinates, to a point $\mathbf{q}_k$ on the image plane.

A point $\mathbf{p}_k$, can be projected onto an image plane as the point $\mathbf{q}_k$. By introducing a homogeneous coordinate system, $\mathbf{p}_k$ can be represented as:

$$\mathbf{p}_k = [x, y, z, 1]^T \tag{1}$$

$\mathbf{p}_k$ can be projected through the aperture $\mathcal{O}$ of a camera located at the origin, and represented by a point $\mathbf{q}_k$

$$\mathbf{q}_k = [x_f, y_f, z_f, 1] \tag{2}$$

on the image plane where x_f, y_f and f are the x- and y-coordinates in the image plane and the focal length. By placing the image plane in front of the aperture as illustrated in Fig. 2, the relations of similar triangles can be used to express the projection (Kuipers, 2002)

$$\mathbf{q}_k = [\frac{f}{z}x, \frac{f}{z}y, f, 1]^T \tag{3}$$

(3) can be rewritten in matrix notation

$$\mathbf{q}_k = \mathbf{Q}\mathbf{p}_k \tag{4}$$

where the perspective transformation $\mathbf{Q}$ is given by

$$\mathbf{Q} = \frac{f}{z}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{f} & 0 \end{bmatrix} \tag{5}$$

Given a point in space $\mathbf{p}_{k,n} = [x_n, y_n, z_n, 1]^T$, the position $\mathbf{p}_{robot} = [x_r, y_r, z_r, 1]^T$ and orientation $\mathbf{\Theta}_{robot} = [\theta_r, \phi_r, \psi_r, 1]^T$ of the robot carrying the camera in NED coordinates, a rotation from NED-frame to body-frame is given by the rotation matrix

$$\mathbf{R}_{bn} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6}$$

[1] (Stackpole and Lang, 2012)

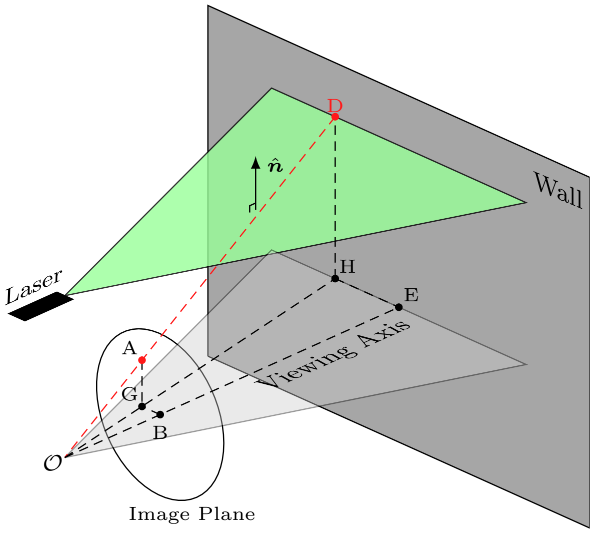


Fig. 2. Modified Pinhole model

where $\mathbf{R}_{bn}$ is the rotation matrix from NED-frame to body, defined as

$$\mathbf{R}_{bn} = \mathbf{R}_{nb}^T \tag{7}$$

and a translation, given by the translation matrix $\mathbf{T}_{bn}$.

$$\mathbf{T}_{bn} = \begin{bmatrix} 1 & 0 & 0 & \Delta x \\ 0 & 1 & 0 & \Delta y \\ 0 & 0 & 1 & \Delta z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8}$$

is needed to express a point $\mathbf{p}_{k,n} = [x_n, y_n, z_n, 1]^T$ in NED-frame, in body-frame coordinates $\mathbf{p}_{k,b} = [x_b, y_b, z_b, 1]^T$. Rewriting (4) with the translations and rotation,

$$\mathbf{q}_{k,b} = \mathbf{Q}\mathbf{R}_{bn}\mathbf{T}_{bn}\mathbf{p}_{k,n} \tag{9}$$

where $\mathbf{p}_{k,n}$ and $\mathbf{q}_{k,b}$ are the k-th point in NED and body-frame respectively.
Further, the image plane can differ in orientation and origin compared to the body-frame. In order to achieve a complete general form of the projection from NED-frame to camera-frame, another rotation and translation is needed

$$\mathbf{q}_{k,c} = \mathbf{Q}\mathbf{R}_{cb}\mathbf{T}_{cb}\mathbf{R}_{bn}\mathbf{T}_{bn}\mathbf{p}_{k,n} \tag{10}$$

Thus, any given mounting position and orientation of the camera can be accounted for.

2.3 Triangulation

By placing two lasers, above and below the camera, each projecting a laser line that is parallel to the viewing axis of the camera, the projected lasers can be used for triangulation. Fig. 3 illustrates the triangle $\mathcal{O}DH$ in Fig. 2 projected onto the viewing axis, for both the top and bottom laser, thus illustrating the two-dimensional setup. Note that the laser points for the remainder of the section will be treated in the body-frame $\mathbf{p}_{k,b} = [x_b, y_b, z_b, 1]^T$, as the rotation into NED-frame is a simple matter of rotating and translating given by an AUV pose.

The lasers can be seen as producing a single point, D and F respectively, which are projected through the origin $\mathcal{O}$ onto the image plane as the points A and C. The baseline distance b between the lasers, will reflect the distance z_b, as the lasers are parallel (Cain and Leonessa, 2012)

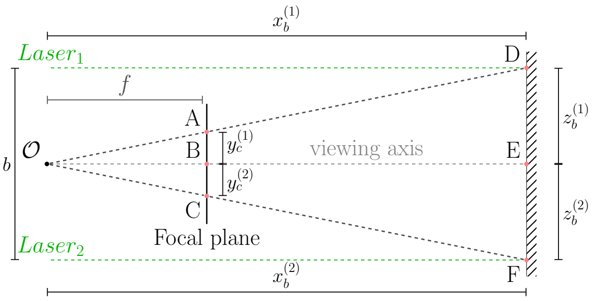


Fig. 3. Side view of the laser sensor triangulation

$$z_b = b \tag{11}$$

2.4 Extended Triangulation

In (Cain and Leonessa, 2012) the total baseline distance b between the lasers is used for triangulating a single distance to an unknown object.
In this work we present a solution that utilizes each laser independently, thus a range estimate is achieved for each laser, see Fig. 3. This increase in information yields an increase in dimensionality of the results, thus we are able to reconstruct a 3-dimensional point cloud of the surroundings instead of a 2-dimensional point cloud.

Consider the setup in Fig. 3: the lasers and the aperture of the camera (points $\mathcal{O}$, D & F) create a triangle that is similar to the triangle created by the projected points on the image plane and the aperture (points $\mathcal{O}$, A & C). Using the relation between these two triangles would result in a single distance (from origin $\mathcal{O}$ to point E). However, the projected points of the lasers, as shown in Fig. 3, creates four, pairwise similar triangles - by using the cameras viewing axis as the base of the triangle - instead of only the two used by (Cain and Leonessa, 2012). The triangles $\mathcal{O}DE$ and $\mathcal{O}AB$, $\mathcal{O}EF$ and $\mathcal{O}BC$ are pairwise similar, as the angles $\angle\mathcal{O}AB = \angle\mathcal{O}DE$, and $\angle\mathcal{O}CB = \angle\mathcal{O}FE$.
By defining the distances from the points A, C, D and F to the cameras viewing axis (points B and E), as

$$y_c^{(1)} \triangleq |AB|, \quad z_b^{(1)} \triangleq |DE| \tag{12}$$

$$y_c^{(2)} \triangleq |BC|, \quad z_b^{(2)} \triangleq |EF| \tag{13}$$

with the opposite sides of the triangles defined, the relations in (4) can be used to find the distances $x_b^{(i)}$. Inserting the definitions into the relations of similar triangles in (3) and (4) yields

$$[x_c^{(i)}, y_c^{(i)}, z_c^{(i)}, 1]^T = \mathbf{R}_{cb}\mathbf{T}_{cb}[x_b^{(i)}, y_b^{(i)}, z_b^{(i)}, 1]^T \tag{14}$$

$$[x_c^{(i)}, y_c^{(i)}, f, 1]^T = \mathbf{Q}[x_c^{(i)}, y_c^{(i)}, z_c^{(i)}, 1]^T \tag{15}$$

Without loss of generality, the origin of the camera frame is placed in the origin of the body-frame, this ensures that only a rotation is needed when going from body-frame to camera frame, and vice versa.

$$[x_c^{(i)}, y_c^{(i)}, f, 1]^T = \mathbf{Q}\mathbf{R}_{cb}\mathbf{T}_{cb}[x_b^{(i)}, y_b^{(i)}, z_b^{(i)}, 1]^T \tag{16}$$

where the rotation matrix $\mathbf{R}_{cb}$ and translation matrix $\mathbf{T}_{cb}$ are

$$\mathbf{R}_{cb} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{T}_{cb} = I_4 \tag{17}$$

thus, the camera and body-frame origin are identical, yet the frames are rotated. Due to the definition of the camera frame, the distance z_c (distance from camera to object), expressed in body coordinates, can be reduced to $x_b^{(i)}$. Solving (16) for $x_b^{(i)}$

$$x_b^{(i)} = z_b^{(i)} \frac{f}{y_c^{(i)}} \tag{18}$$

(18) is clearly nonlinear, which stems from the nonlinearities of the *perspective projection*[2] in (5), i.e. angle information and distance information are lost in the transformation. The nonlinearities in (18) can however be neglected, as the baseline distance b is known, information about $z_b^{(i)}$, is known

$$b = z_b^{(1)} + z_b^{(2)} \tag{19}$$

where $z_b^{(i)}$ is a known distance between laser i and the camera

$$|z_b^{(i)}| = c^{(i)}, \quad c^{(i)} \in \mathbb{R} \tag{20}$$

With $x_b^{(i)}$ and $z_b^{(i)}$ known, the last coordinate, $y_b^{(i)}$ can be calculated. Consider again that the problem can be expressed in two dimensions by the projection of triangles. Fig. 4 shows a top view of the setup with the modified pinhole camera model. The point A is the same point from the top laser, as illustrated in Fig. 3 By projecting the point H onto the image plane as point G, two triangles are generated. The triangles $\mathcal{O}GB$ and $\mathcal{O}HE$, are similar, meaning that $\angle\mathcal{O}GB = \angle\mathcal{O}HE$. Now defining the distances from G and H to the points B and E respectively, as

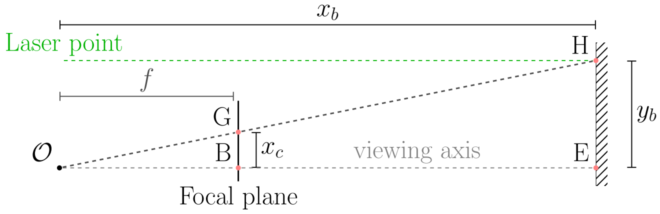


Fig. 4. Top view of the laser sensor triangulation

$$x_c \triangleq |GB|, \quad y_b \triangleq |HE| \tag{21}$$

with the opposites defined, and inserting into (16) and solving for y_b gives

$$y_b = x_b \frac{x_c}{f} \tag{22}$$

inserting (18) into (22) yields a reduced expression

$$y_b = z_b \frac{x_c}{y_c} \tag{23}$$

Any point $\mathbf{p}_{k,n}$ in NED-frame, can be reconstructed by a point $\mathbf{q}_{k,b}$ in body-frame by (18), (19) and (23), repeated here for convenience:

$$x_b = z_b \frac{f}{y_c}, \quad y_b = z_b \frac{x_c}{y_c}, \quad z_b = c \tag{24}$$

2.5 Generalized Triangulation

The equations summarized in (24) are valid only under the assumption that the lasers that are parallel to the viewing axis of the camera on the AUV, such that the distance $z_{b,i}$ is constant, and therefore known (19). Thus the right-angled triangulation is only possible in the special case, where the lasers and camera axes are parallel, something that can be difficult to achieve. Thus, any alignment or tilt/angle errors between the lasers or the camera, will invalidate the assumptions behind (24). The implications of operating under invalid conditions for the above equations, depends entirely on the requirements set for the sensor system, e.g. low tolerance in distance errors will reflect on the misalignment tolerances.

The triangulation must be generalized to avoid the hard constraints on the alignment imposed by the right-angled triangulation scheme. To generalize a solution based on plane-line intersection is pursued. Consider the laser projecting a plane into the environment. The laser plane is defined by a normal vector $\mathbf{n}_{laser,b} = [x_b, y_b, z_b, 1]^T$ and a single point on the plane $\mathbf{p}_{laser,b} = [x_{0,b}, y_{0,b}, z_{0,b}, 1]^T$, the position of the laser in body-frame, as illustrated in Fig. 2. Any point $\mathbf{p}_{k,b}$ can be said to belong to the plane defined by $\mathbf{n}_{laser,b}$ and $\mathbf{p}_{laser,b}$ if the points satisfies (25).

$$(\mathbf{p}_{k,b} - \mathbf{p}_{laser,b}) \cdot \mathbf{n}_{laser,b} = 0 \tag{25}$$

The point $\mathbf{q}_{k,b}$ on the image plane can be considered to belong to the projection line created by projecting the point $\mathbf{p}_{k,b}$ through the aperture $\mathcal{O}$. Thus, the relation between the points $\mathbf{p}_{k,b}$ and $\mathbf{q}_{k,b}$ are described by the line equation:

$$\mathbf{p}_{k,b} = d\mathbf{n}_{line,b} + \mathbf{q}_{k,b} \tag{26}$$

where d is a scalar distance along the vector $\mathbf{n}_{line,b}$, which is given by:

$$\mathbf{n}_{line,b} = \mathbf{q}_{k,b} - \mathcal{O} \tag{27}$$

Inserting (26) in (25) and solving for the d:

$$d = \frac{(\mathbf{p}_{laser,b} - \mathbf{q}_{k,b}) \cdot \mathbf{n}_{laser,b}}{\mathbf{n}_{line,b} \cdot \mathbf{n}_{laser,b}} \tag{28}$$

Finally, inserting (28) into the line equation in (26).

$$\mathbf{p}_{k,b} = \frac{(\mathbf{p}_{laser,b} - \mathbf{q}_{k,b}) \cdot \mathbf{n}_{laser,b}}{\mathbf{n}_{line,b} \cdot \mathbf{n}_{laser,b}} \mathbf{n}_{line,b} + \mathbf{q}_{k,b} \tag{29}$$

The expression in (29) is valid for any orientation of the laser plane, however a estimate of the plane normal $\mathbf{n}_{laser,b}$ is necessary.

2.6 Calibration

In order to estimate $\mathbf{n}_{laser,b}$ the robot is positioned with a known distance and angle towards a planar surface e.g. a wall. Using the knowledge that the projection of the points in space $\mathbf{p}_{k,n}$ onto the image plane is the intersection between the wall and the laser plane, the estimate of the laser plane can be conditioned on the wall parameters. Hence (30) mirrors (25) with all laser parameters replaced by wall parameters.

$$(\mathbf{p}_{k,b} - \mathbf{p}_{wall,b}) \cdot \mathbf{n}_{wall,b} = 0 \tag{30}$$

In the same spirit as previous section (26) into (30) and solving for d yields (31)

$$d = \frac{(\mathbf{p}_{wall,b} - \mathbf{q}_{k,b}) \cdot \mathbf{n}_{wall,b}}{\mathbf{n}_{line,b} \cdot \mathbf{n}_{wall,b}} \tag{31}$$

Having found the points of intersection between the wall and laser plane, the problem of finding a laser plane $\mathbf{n}_{laser,b}$ such that (25) is satisfied. This is done using Least Squares approximation.

[2] (Siegwart et al., 2011, Equations 4.36-37 p. 153)

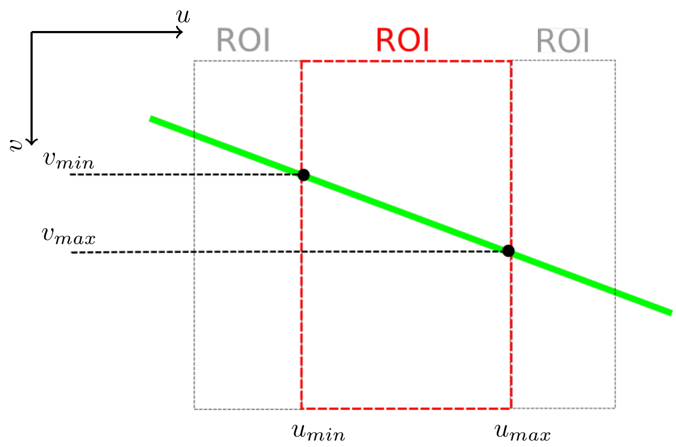


Fig. 5. Overview of a ROI

3. IMAGE ANALYSIS

A captured image is divided into a number of Regions Of Interest (ROI), and data is extracted from each ROI. To estimate the laser lines in the image, the linear probabilistic Hough transform is used (see (Kiryati et al., 1991) and (Matas et al., 2000)).

A color filter is used on the original image to create a binary image, on which, for each ROI, the Hough Transform is applied. A single center point on each line is estimated in each ROI. This center the line is computed as

$$u_{avg} = \frac{u_{\max} - u_{\min}}{2}, \quad v_{avg} = \frac{v_{\max} - v_{\min}}{2} \tag{32}$$

where $u_{\min}$ and $u_{\max}$ are the horizontal borders of the ROI, this ensures that the points are always in the center of the ROI, resulting in a constant angle difference between the points, as illustrated in Fig. 5. $v_{\min}$ and $v_{\max}$ are the top and bottom points in vertical axis in each ROI.

This method can create some offset from the real lines, due to the centering of the points in the horizontal axis, this has however not been prominent in the simulations and results. A brief outline of the entire vision algorithm is given in Algorithm 1

Algorithm 1 Pseudo Code of Complete Algorithm

```
for all Image do
    Convert image to HSV colors
    Filter greens form image
    Skeletonize image to reduce line width
    for all Region of Interest do
        Extract lines by Hough Transform
        Calculate center $(u, v)$ of lines                ▷ (32)
        Transform $(u, v)$ to $(x_{c,i}, y_{c,i}, z_{c,i})$
        Calculate $(x_{b,i}, y_{b,i}, z_{b,i})$          ▷ Eq. (24)
        add $(x_{b,i}, y_{b,i}, z_{b,i})$ to pointcloud
return pointcloud
```

4. EXPERIMENTAL PLATFORM

The OpenROV platform is chosen as the basis for the experimental setup. Computationally the OpenROV is powered by the BeagleBone Black ARM platform, and comes with a variety of sensors and three thrusters, yielding three degrees of freedom (surge, yaw and heave). The sensors included are: Inertial Measurement Unit (IMU), pressure sensor and a web-camera.

An ARM distribution of Robot Operating System (ROS) is used as an overlay to operate the vehicle. ROS allows for easy communication to and from the OpenROV, has several logging tools readily available, and prebuilt packages for the OpenCV computer vision libraries. ROS enables multiple robots to be connected in a ROS network, a necessity of the future vision of this system, see (Quigley et al., 2009).

5. EXPERIMENTS AND RESULTS

This section will cover the simulation and testing of the methods and algorithms outlined in the above sections.

5.1 Simulation

To test the developed algorithm, a simulated image is constructed using the 3D modeling software Blender. A representation of the AUV with camera and lasers, is placed inside a $2m$-by-$2m$ box, $3m$ away from the front facing wall, and centered between the side walls. The simulated lasers and camera both have a fan-angle of $\alpha = 120°$. Fig. 6 shows an illustration of the simulation environment. The top image in Fig. 7 shows an image captured from the AUV in the simulation environment from Fig. 6, with the estimated centers of the lines displayed onto the image is shown in the bottom image. Fig. 7 exemplifies that the algorithm outlined in the previous section is able to identify the lines in the image and estimate the position of both lasers in the image. The estimated points are projected into a body-frame coordinate system, to verify the algorithm, Fig. 8 shows this projection plotted in body-frame. Fig. 8 shows the algorithm successfully projects the points back into body-frame.

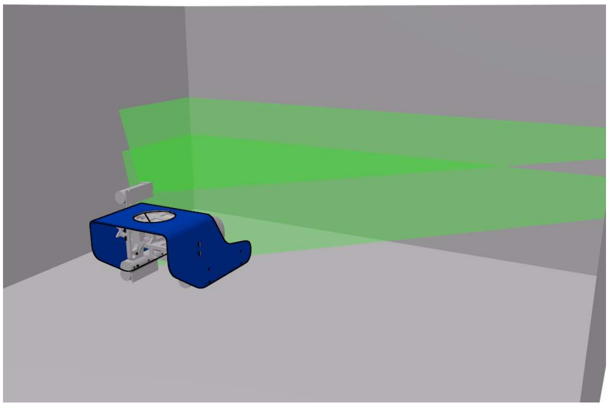
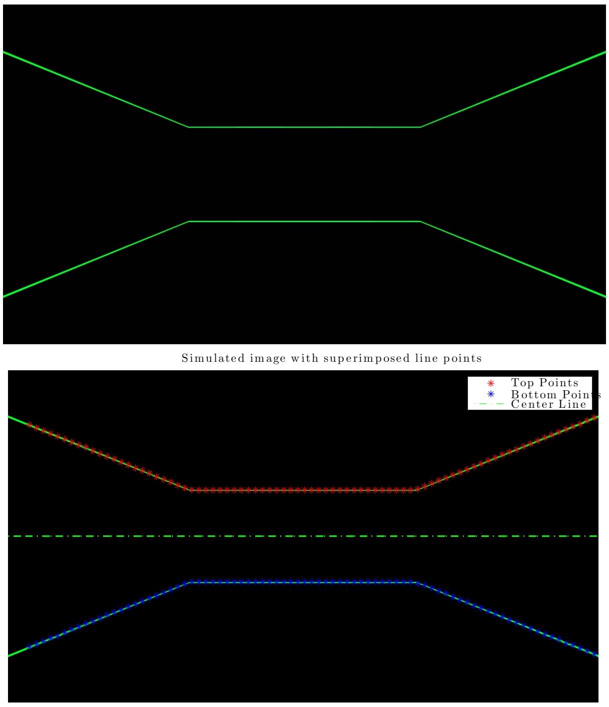


Fig. 6. Simulation Setup, with AUV placed in cubicle

The distance error of each wall, both for the top and bottom laser, is calculated for each point individually, the results are shown in the error plot in Fig. 8. To calculate the distance error, each of the projected points are assigned to a specific wall and the distance from the plane formed by each wall is calculated. The distance error for each of the three walls is $\approx 1 - 2\%$, except for two points on left wall, bottom laser. These higher errors are due to these two points being wrongfully included as the left wall, and not the center wall.



Simulated image with superimposed line points

Fig. 7. Simulated Image, produced by the setup

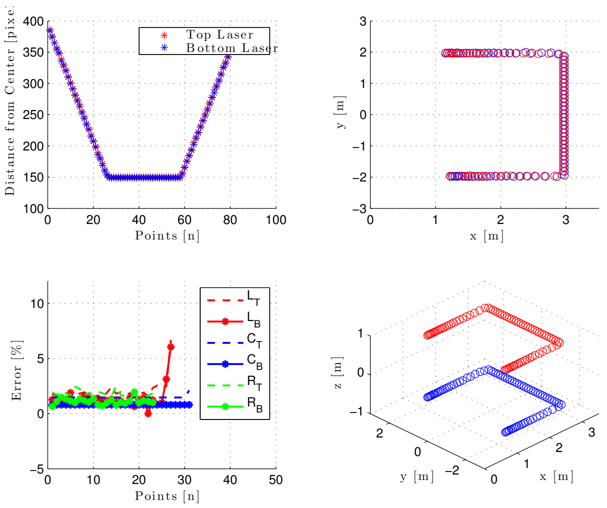


Fig. 8. Line points transformed into 3D space

The theoretical maximum range of the sensor can be calculated by (18)

$$x_b = b\frac{f}{y_c} \tag{33}$$

where the point in the camera frame $\mathbf{y}_c$ is calculated in (34)

$$y_c = v\frac{h_s}{h_p} \tag{34}$$

where h_s is the height of the CCD sensor, h_p is the vertical pixel resolution and v is the pixel value.
By setting

$$\begin{aligned} h_s &= 0.018\text{m} \\ h_p &= 1080\text{pixels} \\ v &= [1, \ldots, 300] \\ f &= 0.009238\text{m} \\ b &= 0.1\text{m} \end{aligned}$$

Further, a series of simulated distance images are created in the simulation environment, with the above given specifications to verify the theoretical distance. The AUV is placed directly facing a wall at an 90° angle, at an increasingly greater distance. The simulated and theoretical results are shown in Fig. 9 The simulated results follow the theoretical results closely, thus the algorithm is working on the simulated images. The pixel value is < 50 for distances < 2m, which is more than sufficient for the small confined spaces that the system is intended for. The accuracy of the sensor is degrading as the distance increase, as the second plot in Fig. 9 illustrates. The plot shows the pixel difference over distance and the 1 pixel difference is highlighted (green line). From the plot it is seen that the pixel difference for distances > 2.5m becomes < 1pixel. This results in quantization errors becoming too great, for any distances beyond this, as such the sensor is no longer a viable at range above this threshold value.

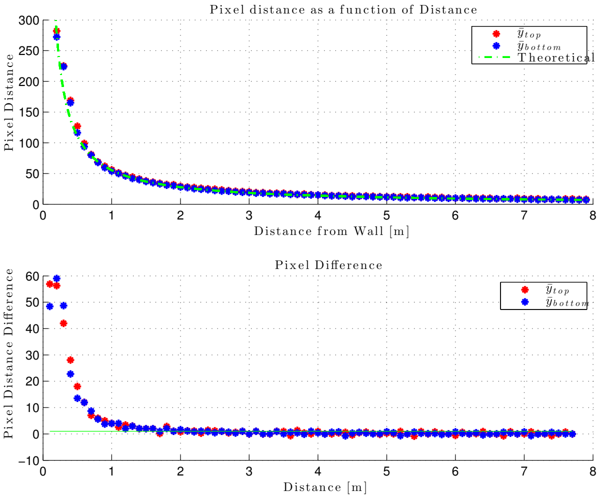


Fig. 9. Pixel values as function of distance

In order to extend the theoretical maximal range of the sensor, two given options are to:

- Increase the camera resolution
- Increase the baseline distance between the lasers and the camera

By increasing the baseline distance the theoretical maximum distance can be found from (33). Fig. 10 shows the maximum theoretical distance calculated for baseline distances $b \in \{0.1\text{m}, 0.2\text{m}, 0.3\text{m}, 0.4\text{m}, 0.5\text{m}\}$.
From Fig. 10 it is seen that an increase in the baseline distance $b = 0.5$m results in an increase the effective range of the sensor to $5 - 8$m.

5.2 Real Test

To test the developed algorithm the OpenROV is fitted with two line lasers, and placed in a testing environment.

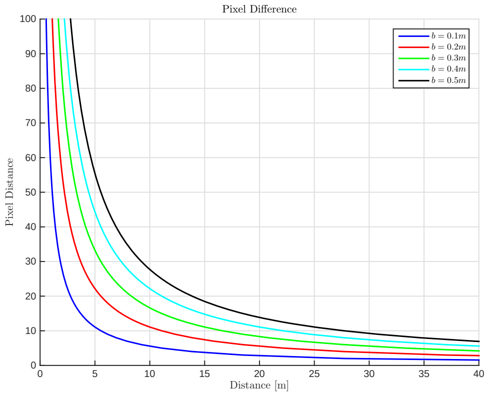


Fig. 10. Theoretical Pixel values as function of distance

The testing environment simulates the small and confined spaces that could make up the insides of a submerged man-made offshore structure. A series of images are taken throughout the experimental setup, by manually placing the OpenROV in different positions and orientations, simulating an inspection mission. Fig. 11 shows an image taken from the OpenROV, and the found lines from the image algorithm. The image shown in Fig. 11 is taken by placing the OpenROV facing into a corner, parallel to the nearest wall. The identified laser line segments found by the Hough Transform are highlighted in red in Fig. 11, with the green dots running along the center of the image, are areas in which the Hough transform did not find any lines.

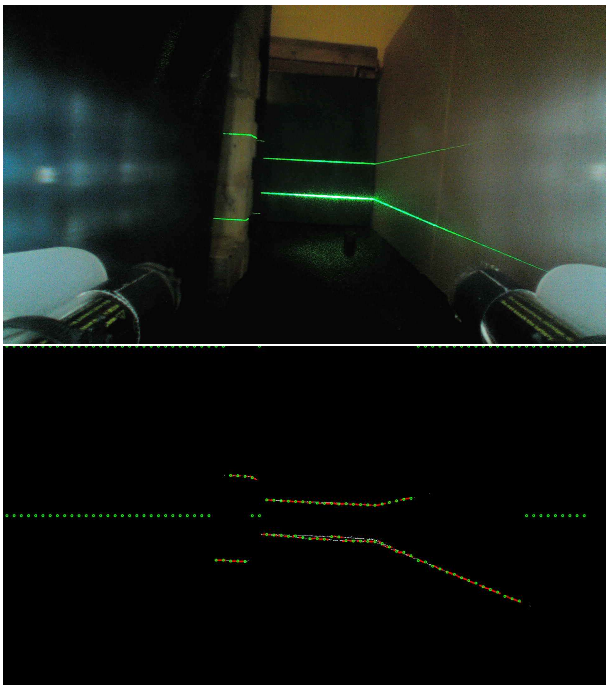


Fig. 11. Original and processed image from the OpenROV

When comparing the two images in Fig. 11, it is seen that the image analysis algorithm successfully finds and extracts the visible laser lines. In order to estimate the distances in the image, a calibration is conducted on a series of images with known x_b distances. It is assumed that the normal plane of both the wall and the camera is aligned with the x-axis of the body-frame, thus $\mathbf{n}_{camera,b} = \mathbf{n}_{wall,b} = [1, 0, 0]^T$. Fig. 12 provides an overview of the error in the estimated distances between the camera and the wall in both calibrated and uncalibrated cases for both top laser and bottom laser. In the uncalibrated case the laser planes are assumed to be parallel with the viewing axis of the camera and hence fulfilling the right-angled constraint. For the bottom mounted laser the actual alignment of the laser is almost identical to the assumed alignment, while the top mounted laser is significantly off. Fig. 13 shows the calibrated and uncalibrated point projections in body-frame. The calibration is performed based on the image at $0.5m$ and two subsequent images at $0.45m$ and $0.55m$ are projected based on this calibration. Compared to the uncalibrated projections, all the calibrated projections align with the wall constraint. However the assumption that $\mathbf{n}_{camera,b} = \mathbf{n}_{wall,b} = [1, 0, 0]^T$ might not be valid in each image due to uncertainties in the test setup. To avoid this $\mathbf{n}_{wall,b}$ must be estimated for each image. This could possibly be done by placing a known checkerboard pattern on the wall and use common camera calibration methods as shown in (Massot-Campos and Oliver-Codina, 2014) to estimate the wall normal plane.

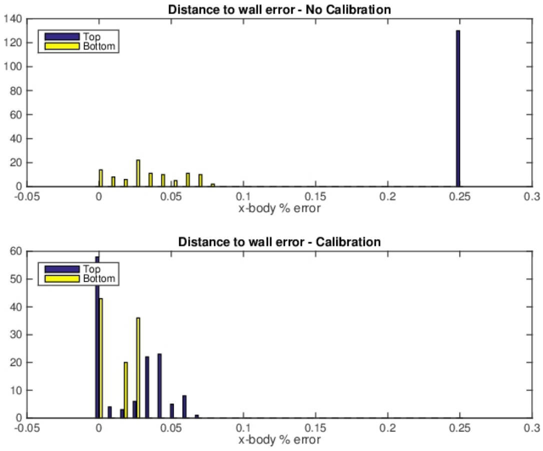


Fig. 12. Chart showing the difference between the actual distance to the wall and the calculated distance

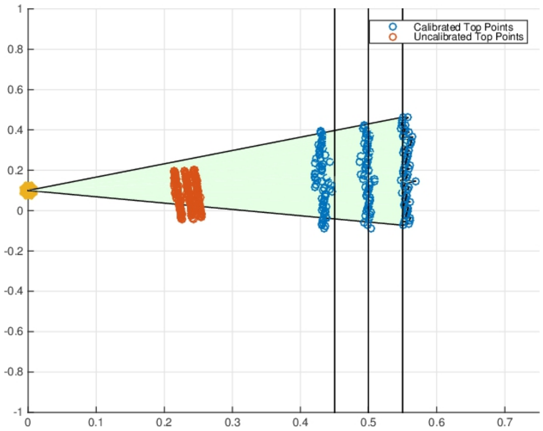


Fig. 13. Calibrated and uncalibrated point project into body-frame at different distances.

6. CONCLUSIONS

This paper has presented a low complexity, low cost three-dimensional underwater range sensor, which extends the work of (Cain and Leonessa, 2012). The sensor was tested in simulation and experimentally. Further, a calibration method was proposed, tested and showed a post-calibration error of approximately 5% and below. The system consists of a combination of lasers and a camera, both of which can be water proofed and are available in relatively small form factors, making this sensor suitable for smaller AUV vehicles designed for navigating in confined spaces. The sensor was shown to provide a three-dimensional point cloud of the surroundings, which could be used for e.g. feature extraction in navigation filters.

7. FUTURE WORK

Plans for further work on this sensor consists of conducting underwater experiments, especially focusing on different turbidity conditions. Turbid water is the main challenge to optical based systems underwater and hence real underwater tests at different levels of turbidity is required to fully evaluate the robustness of this approach.

The results of the paper shows that the required assumption on the alignment between camera and wall normal planes might not hold, hence the wall normal should be estimated as previously discussed. Furthemore work towards utilizing the sensor in closed loop control and ultimately in basic environmental mapping is still necessary. The system is not constrained to work with two line lasers, laser projectors could be used to project patterns into the environment. However such a system would require special attention to identifying and isolating specific parts of the pattern.

REFERENCES

Bruno, F., Bianco, G., Muzzupappa, M., Barone, S., and Razionale, A. (2011). Experimentation of structured light and stereo vision for underwater 3D reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(4), 508–518.

Cain, C. and Leonessa, A. (2012). Laser Based Rangefinder for Underwater Applications. *American Control Conference (ACC)*, 6190–6195.

Carreras, M., Ridao, P., Garcia, R., and Nicosevici, T. (2003). Vision-based localization of an underwater robot in a structured environment. *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, 1, 971–976.

Chandrasekhar, V., Seah, W.K., Choo, Y.S., and Ee, H.V. (2006). Localization in underwater sensor networks: survey and challenges. *Proceedings of the 1st ACM international workshop on Underwater networks - WUWNet '06*, 33.

Chen, H.H.C.H.H. and Lee, C.J.L.C.J. (2000). A simple underwater video system for laser tracking. *OCEANS 2000 MTS/IEEE Conference and Exhibition. Conference Proceedings (Cat. No.00CH37158)*, 3, 1543–1548.

Fossen, T.I. (2011). *Handbook of marine craft hydrodynamics and motion control.* John Wiley & Sons, Ltd.

Fournier, G.R. (1993). Range-gated underwater laser imaging system. *Optical Engineering*, 32(9), 2185.

Garcia, R., Nicosevici, T., Ridao, P., and Ribas, D. (2003). Towards a real-time vision-based navigation system for a small-class UUV. *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, 1(October).

Karras, G.C. and Kyriakopoulos, K.J. (2007). Localization of an underwater vehicle using an IMU and a laser-based vision system. *2007 Mediterranean Conference on Control and Automation, MED.*

Kinsey, J.C., Eustice, R.M., and Whitcomb, L.L. (2006). A Survey of Underwater Vehicle Navigation: Recent Advances and New Challenges. *{IFAC} Conference of Manoeuvering and Control of Marine Craft.*

Kiryati, N., Eldar, Y., and Bruckstein, A. (1991). A probabilistic Hough transform. *Pattern Recognition*, 24(4), 303–316.

Kondo, H. and Ura, T. (2004). Navigation of an AUV for investigation of underwater structures. *Control Engineering Practice*, 12(12), 1551–1559.

Kuipers, J. (2002). *Quaternions and Rotation Sequences: A Primer with Applications to Orbits, Aerospace, and Virtual Reality.* Princeton paperbacks. Princeton University Press.

Massot-Campos, M. and Oliver-Codina, G. (2014). One-shot underwater 3D reconstruction. In *Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA)*, 1–4. IEEE, Barcelona.

Matas, J., Galambos, C., and Kittler, J. (2000). Robust Detection of Lines Using the Progressive Probabilistic Hough Transform. *Computer Vision and Image Understanding*, 78(1), 119–137.

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Berger, E., Wheeler, R., and Mg, A. (2009). ROS: an open-source Robot Operating System. *ICRA*, 3(Figure 1), 5.

Siegwart, R., Nourbakhsh, I., and Scaramuzza, D. (2011). *Introduction to Autonomous Mobile Robots.* Intelligent robotics and autonomous agents. MIT Press.

Stackpole, E. and Lang, D. (2012). Openrov. URL `http://openrov.com`.