

# CALIBRATION AND VALIDATION OF THE INTEL T265 FOR VISUAL LOCALISATION AND TRACKING UNDERWATER

Tim Appelt, Joschka van der Lucht, Michael Bleier, Andreas Nüchter

Informatics VII – Robotics and Telematics, Julius Maximilian University of Würzburg, Germany  
{joschka.lucht, michael.bleier, andreas.nuechter}@uni-wuerzburg.de

Commission II, WG II/9

**KEY WORDS:** visual odometry, underwater, robotics, Intel T265, stereo camera, calibration, stereo calibration

## ABSTRACT:

Localization and navigation for autonomous underwater vehicle (AUV) has always been a major challenge and many situations complex solutions had to be devised. One of the main approaches is visual odometry using a stereo camera. In this study, the Intel T265 fisheye stereo camera has been calibrated and tested to determine its usability for localisation and navigation under water as an alternative to more complex systems. Firstly the Intel T265 fisheye stereo camera was appropriately calibrated inside a water filled container. This calibration consisting of camera and distortion parameters got programmed onto the T265 fisheye stereo camera to take the differences between land and underwater usage into account. Successive the calibration, the accuracy and the precision of the T265 fisheye stereo camera were tested using a linear, a circular and finally a chaotic motion. This includes a review of the localisation and tracking of the cameras visual odometry compared to a ground truth provided by an OptiTrack V120:Trio to account for scaling, accuracy and precision. Also experiments to determine the usability with fast chaotic motions were performed and analysed. Finally, a conclusion concerning the applicability of the Intel T265 fisheye stereo camera, the challenges using this model, the possibilities for low cost operations and the main challenges for future work is conducted.

## 1. INTRODUCTION

Performing reliable localization and navigation is well documented and investigated. But in highly unstructured underwater environments it remains very difficult. While localisation and navigation in a structured environment can be easily accomplished by using a vision based approach, a laser based approach or relying on the Global Positioning System (GPS), these methods create significant challenges underwater. Localization and navigation of an Autonomous Underwater Vehicle (AUV) has to meet a lot of requirements to ensure the correct movement of the robot in underwater environments. It is very important to get reliable measurement in critical missions like the detection of unexploded naval mines or mapping task in a reef environment. Most techniques use an acoustic (Leonard et al., 1998) or vision based approach (Dunbabin et al., 2006).

The Intel T265 is designed to capture the trajectory on land through visual odometry. However, the goal is to be able to use this stereo camera underwater as well. The main question is to what extent this is possible and therefore also useful. In this paper we look into the question of whether the Intel T265 can be adjusted by calibrating its fisheye camera so that it can also be used for tracking underwater. And to what extent the Intel T265 could be limited in its usability under water due to problems that it also shows on land.

This paper evaluates the capability of the Intel T265 camera for localisation and tracking in an underwater near range environment. It describes the experimental platforms, procedures and then presents results for the reliability of the vision system. Finally, it evaluates the results for tracking and localisation with the Intel T265 and examines the usability for underwater applications.

## 2. STATE OF THE ART

There are different approaches to localisation in an underwater environment. One of the difficult aspects of this endeavour is that approaches that work on land not necessarily translate that well into underwater environments. For example Global Positioning System (GPS) will only work above water (Taraldsen et al., 2011), because of the disturbances under water, laser based approaches tend to have difficulties as a result of the light refraction.

One of the possibilities in this field lies with vision-based approaches of underwater localisation. But there are some requirements to the successful use. Firstly the AUV has to travel relative close to trackable features. Concluding that the visual approach needs to be performed close to the bottom of the ocean. Performing reliable localization and navigation is well documented and investigated. But in highly unstructured underwater environments it remains very difficult.

For the typical visual approach relatively clear shallow waters and sufficient natural lighting is needed. Also the robot has to be in a feature rich environment. It is most suited for about 1 m altitude above the ocean floor or near coral reefs. To evaluate the pictures of the cameras the detection of features is needed. Features are a local, meaningful, detectable parts of the image. The visual approach uses the Harris feature detection to automatically detect the features. This means the image has to be filtered, the gradients of each pixel have to be computed, a window around each pixel has to be constructed and the determinants of the windows have to be computed. The Results are reliable and temporal stable features. After the features are computed the algorithm searches for similar features using the normalized cross correlation similarity measure (ZNCC). Approximate epipolar constraints are used to prune the search



Figure 1. Custom underwater camera housing for the Intel T256.

space and only the strongest corners are evaluated. If similar features have been detected in both images of the stereo camera and again one time step ahead in the next pair of pictures the features create a 3 way match. Now these features get undistorted and their location is used to calculate the image motion. The result of the differential image motion is now used to estimate a feature's three-dimensional position. This also calculates the altitude above sea level. Now the motion of the vehicle is predicted using an iterative least median squares method. This results in a translation vector of the robot. The translation vector is now used to calculate the new position of the robot.

Another possibility is to use a SLAM Algorithm for the visual approach. The main difference here is the fact that an IMU is used in addition to the localization via visual odometry. In the case that the visual odometry is not precise enough, this should enable a correction to prevent errors. However, an IMU that is too inaccurate can also have the opposite effect and lead to inaccuracies and drifts due to the fusing of the data. A typical attempt to counteract this is loop closing in the SLAM algorithm.

A fisheye camera is particularly suitable for underwater as well as above water odometry. The main advantage is the large field of view, which allows the camera to extract more features. This is particularly advantageous in an underwater environment, as the amount of features can be difficult. A disadvantage, however, is the great distortion of the camera, as this is required for a good camera calibration. Otherwise, the distortion would lead to great uncertainty and simply lead to error.

There are many different ways to calibrate a camera. In order to decide which method is best, you first have to know the type of camera and the distortion of the camera in order to select a suitable one. In our case it is a stereo fisheye camera and the calibration of the stereo fisheye camera in OpenCV is particularly suitable. The algorithm implements not only the type of camera but also the type of distortion required.

### 3. EXPERIMENT SETUP

The Experiments have been performed in an 7m by 2.4 by 2.4 water filled container (van der Lucht et al., 2019) The camera used is an Intel T256 with an specially designed housing to protect the camera from water. The housing, seen in Fig. 1, consists of a PET plastic shell with an acrylic glass flat port interface. The T265 is a stereo camera from Intel. It has a size of 108 x 25 x 13mm and weighs about 55g. It was specially developed

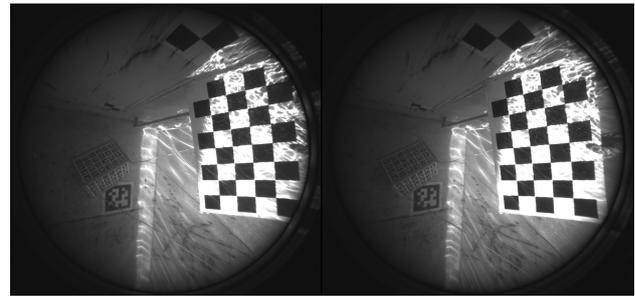


Figure 2. Stereo camera view of the calibration process.

for visual tracking in space. For this, the T265 has two fisheye cameras that can take pictures. It is also equipped with an Intel® Movidius™ Myriad™ 2 VPU, which already calculates a highly optimized visual SLAM on the camera. This leads to a very short delay and very efficient energy consumption. The user can only access the results with software developed by INTEL. These results include a relative position of the camera to its starting position. The so-called visual odometry. This enables the camera to be precisely positioned in the room as soon as there are enough clues for the VISUAL-SLAM. The camera also outputs a stream of both of its camera images. In addition, a gyroscope and an accelerometer are installed, the data of which can also be read out.

The experiments are partitioned in two phases. Because the camera is actually intended for use in air, it's necessary to calibrate the camera first. Once the successful calibration the second part of the experiments is the validation of the resulting accuracy underwater and the comparison between the calibrated and non-calibrated camera.

#### 3.1 Calibration Phase

The occurring refraction on the air-glass interface of the underwater housing requires a new calibration of the camera parameters. A correct and high quality calibration of the camera with special attention to the refraction parameters For this a calibration structure is placed into the water. We use two different types of calibration targets for this type of applications. First one is an planar chessboard pattern and the second one is a corner shaped AprilTag pattern. To ensure the maximum precision, both patterns are float glass plates who are direct printed with the associated pattern.

The images during the camera movement in front of the calibration patterns, seen in Fig. 3, are captured by an notebook under usage of the ROS integration from the provided Intel Software.

#### 3.2 Validation Phase

For the validation of the accuracy using the new calibrated Intel T256 underwater, we placed some markers on top of the housing and used an external tracking system to capture a ground truth trajectory. The tracking system we used is the OptiTrack V120:Trio. This is mounted on an tripod and is placed next to the container. In Fig.4 the intel housing with the mounted markers is shown. The markers are above the water surface during the whole experiments. For the experiments we used three different moving patterns, linear, circular and chaotic. All this movements have been performed with both, the calibrated and non-calibrated camera. This allows us to compare the the calibrated vs. non-calibrated camera to show the improvement

through the calibration and validate both against the OptiTrack ground truth.

#### 4. METHOD



Figure 3. Intel T265 moved in the water above the calibration pattern during the calibration process.

First of all the Intel T265 has to be integrated into the system. The experiment uses a Raspberry PI 4 Model B, that runs a linux distribution that interfaces the camera both with the program provided by Intel and a ROS environment. The Intel T265 creates odometry data using an proprietary Visual-Inertial-SLAM computed on two fisheye cameras and an IMU. The Intel T265 is placed in a specially manufactured housing for water protection, seen in Fig. 1, that later will be mounted to a Bluerobotics BlueRov2.

First it is necessary to recalibrate the Intel T265 for the usage underwater to compensate the refraction at the air-glass and glass-water interfaces of the T265 housing, seen in Fig. 1. To achieve this, we take pictures, seen in Fig. 2 of a planar chessboard and a 3d AprilTag (Olson, 2011) calibration target in a water filled container (van der Lucht et al., 2019). The calibration is done by the OpenCV fisheye stereo implementation also used Abraham and Förstner (Abraham and Förstner, 2005) method. Ensuring we validate the quality of the resulting camera calibration.

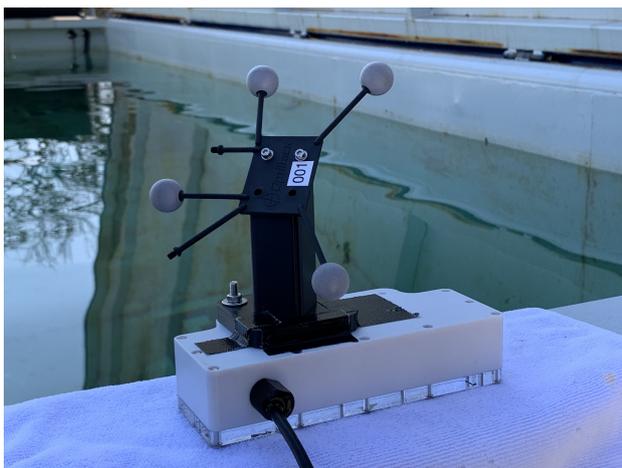


Figure 4. OptiTrack V120: Trio markers mounted on top of the housing to capture ground truth data during the experiments.

This part explains the fisheye stereo calibration method of the Abraham and Förstner (Abraham and Förstner, 2005) which is also implemented by OpenCV. The Method uses "traditional algorithms from the field of photogrammetry and computer vision for calibration of fish-eye stereo". (Abraham and Förstner, 2005) First of all in the first step pictures have to be taken and automatic measurements have to be calculated. This means you have to define a plane with target points that are known. Typical example are a chessboard pattern and AprilTags. In our case both are printed on a glass plate. Then several images of these planes have to be taken using the T265 camera. It must be ensured that all images are taken using the raw images from the Intel T265. Because the normal pictures are already calibrated through the calibration an Intel T265 get at the end of its production process. If this images would be used only the differences between the calibrations will be calculated. We ensure that the planes in the pictures cover nearly the whole field of view and with different distances. The basic Math fo the fisheye model will be described in the following part.

This basic equation describes the coordinate vector  $x(c)$  of a point  $x$  in a reference frame that equals the a Point  $X$  in the world coordinate frame after the application of an Rotation  $R$  and a Translation  $T$

$$x(c) = RX + T , \quad (1)$$

with the coordinates

$$x = Xc(1), \quad y = Xc(2), \quad z = Xc(3) . \quad (2)$$

The pinhole projection coordinates of the point  $x$  that is re-

strained though the pinhole model by

$$\begin{aligned} a &= \frac{x}{z} \\ b &= \frac{y}{z} \\ r^2 &= a^2 + b^2 \\ \theta &= \operatorname{atan}(r) . \end{aligned} \quad (3)$$

The fisheye distortion is expressed by

$$r(\theta) = k_1\theta + k_2\theta^3 + k_3\theta^5 + k_3\theta^7 + \dots \quad (4)$$

and the distorted point  $[x'; y']$  is defined by

$$\begin{aligned} x' &= (\theta(d)/r)a \\ y' &= (\theta(d)/r)b . \end{aligned} \quad (5)$$

For the last step the coordinates get transformed into their respective pixel coordinates. These are described by  $[u; v]$  with

$$\begin{aligned} u &= f(x)(x' + \alpha y') + c(x) \\ v &= f(x)y' + c(y) . \end{aligned} \quad (6)$$

The next step is to approximate the values through direct solutions. For this the software will find the points of interests in the image that vary from method to method, but have to be on a plane. By processing a modified Direct Linear Transformation algorithm for each image a approximate The image processing software performs the point numbering and measurement automatically. Therefore, groups of four points are built twice on each plane allowing an exterior and interior orientation are estimated, considering the equidistant projection model.

Now the camera can be calibrated. Using the approximated points from the bundle adjustment non-linear iterative self-calibrating bundle adjustment the following can be calculated at once:

- intrinsic parameter of the both cameras
- the extrinsic parameters
- the relative stereo parameters
- estimated point coordinates

This algorithm minimizes the reprojection error taking into account the real measurements as well as the projection model. With the help of the intrinsic and extrinsic parameters can be calibrated There are two Calibrations done in this paper:

#### 4.1 Chessboard Calibration

One way to calibrate the cameras is to use chessboard patterns. The method used here already calibrates the cameras as a connected stereo pair. For this purpose, several pairs of images are first recorded, each showing the chessboard pattern in the largest possible version. It is important to record the chessboard pattern in as many positions as possible and to avoid excessive changes in exposure in order to ensure the visibility of the pattern. With the help of these image pairs, the algorithm now calculates a calibration that is as precise as possible. This Method uses the edges of the chessboard pattern as the measured points that than are used in the above described algorithm.

#### 4.2 AprilTag Calibration

The other way to calibrate the cameras is to use AprilTags. The method used here already calibrates the cameras as a connected stereo pair. For this purpose, several pairs of images are first recorded, each showing the AprilTag pattern in the largest possible version. It is important to record the AprilTag pattern as clearly as possible and to avoid excessive changes in exposure in order to ensure the visibility of the pattern. With the help of these image pairs, the algorithm now calculates a calibration that is as precise as possible. These method uses the exact position of the AprilTag for the real measurements in the above described algorithm.

#### 4.3 Distortion

The Distortion used in the calculations is the one proposed by Kannala and Brandt. (Kannala and Brandt, 2004) It uses a combination of radial distortion and tangential distortion. With the overall distortion

$$r(\theta) = k_1\theta + k_2\theta^3 + k_3\theta^5 + k_3\theta^7 + \dots , \quad (7)$$

the radial direction

$$\begin{aligned} \delta(\theta, \varphi) &= (l_1\theta + l_2\theta^3 + l_3\theta^5) \\ & (i_1\cos\varphi + i_2\sin\varphi + i_3\cos 2\varphi + i_4\sin 2\varphi) \end{aligned} \quad (8)$$

and tangential direction

$$\begin{aligned} \delta(\theta, \varphi) &= (m_1\theta + m_2\theta^3 + m_3\theta^5) \\ & (j_1\cos\varphi + j_2\sin\varphi + j_3\cos 2\varphi + j_4\sin 2\varphi) . \end{aligned} \quad (9)$$

This distortion is needed for the Intel T265 and can be generated by the OpenCV Software. This distortion also gets calculated in the calibration and contains like shown a radial and tangential coefficient taking into account that fisheye cameras are never have a perfect radial distortion.

In the next step, the calibrated camera is placed underwater and the tracking and localisation is activated. The trajectory and all odometry data is logged. Ground truth data to validate the accuracy of tracking and localisation of the Intel T256 under water is captured using an external OptiTrack V120:Trio system. The needed markers for the external tracking system are placed on the outside of the housing and are always above the surface of the water. This allows to record the trajectory of the Intel T265 with high precision. The calibration between the coordinate systems of the camera and the OptiTrack System is done by an hand-eye calibration using an 3d AprilTag calibration target. Thus it is possible to compare the trajectories, created by the Intel 265 and the OptiTrack system and validate the accuracy of the Intel 265 for the usage under water. For better comparability we also record data in air to compare the accuracy of the camera under normal conditions in air with the accuracy after new calibration in water.

## 5. RESULTS

The results are structured in three parts according to the three different movement patterns. For each movement pattern there is a calibrated and a non-calibrated data record, turquoise coloured in the figures. Moreover ground truth data, captured by the OptiTrack tracking system and coloured purple in the figures, exists and will be compared to the computed movement by the Intel T256 camera.

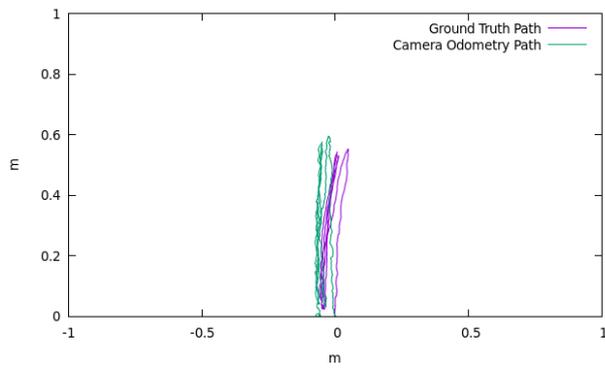


Figure 5. Tracking path for linear movement with calibrated camera.

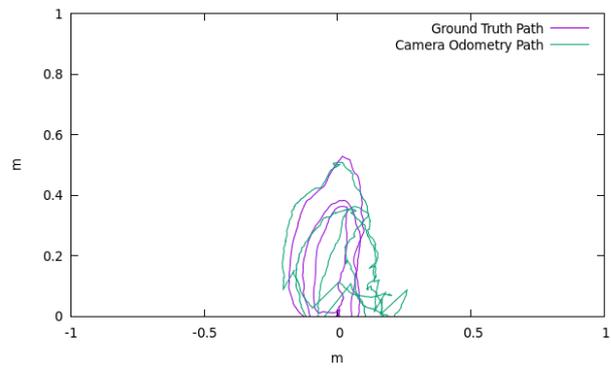


Figure 7. Tracking path for circular movement with calibrated camera.

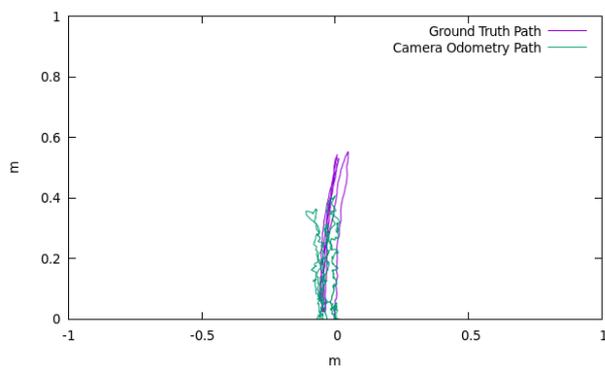


Figure 6. Tracking path for linear movement with non-calibrated camera.

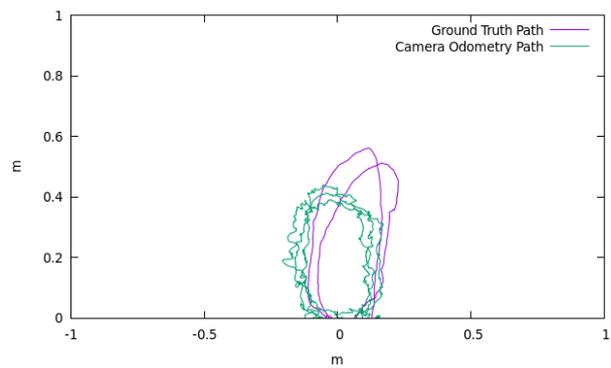


Figure 8. Tracking path for circular movement with non-calibrated camera.

### 5.1 Linear Motion

The first experiment uses a linear motion and shows that both the calibrated and the non-calibrated camera are capable of tracking this simple motion. But two major differences are shown in between Fig. 5 and Fig. 6. While in Fig. 5 the path is very straight and linear, one can see a lot of noise in Fig. 6. This is mainly due to the calibration of the camera, since the de-scaling of the image enables a clearer image and thus better tracking behaviour for the camera. Furthermore you can see a difference in the scaling of both paths. Covering the camera with a glass plate increases the need for proper calibration, while the water increases the focal length. This leads to a non-calibrated camera calculating a shorter distance than the real distance.

### 5.2 Circular Motion

The second experiment features a circular motion to simulate a more complex movement. As seen in 7 and Fig. 8 both are rather successful in tracking the circular motion. But two major differences are shown in between Fig. 7 and Fig. 8. For one Fig. 8 shows again a lot more noise in the path data, but Fig. 7 also shows noise. The noise in Fig. 7 is a result of the properties of the T265. The IMU inside the T265 is not very accurate. If you move the camera to abruptly a SLAM-Error inside the camera is produced. This result in a big uncertainty that has to be counteracted by a good calibration or will be ultimately resolved by the loop-closing feature of the camera. This effect is shown more detailed in the next subchapter. While the calibrated path is straight one can see a lot of the noise in the

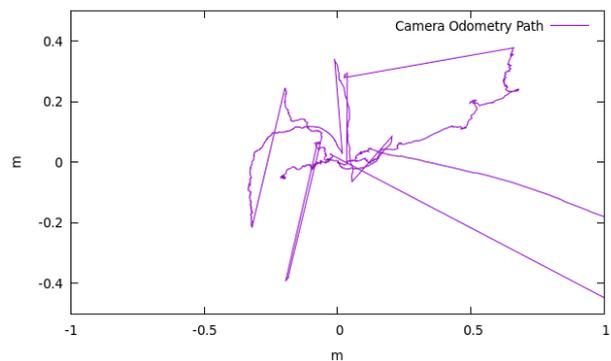


Figure 9. Tracking path for free movement with calibrated camera.

non-calibrated path. This is again mainly due to the calibration of the camera, since the de-scaling of the image enables a clearer image and thus better tracking behaviour for the camera. Another indicator again is the scaling of the path. While both trajectories continued for a similar length. The non-calibrated path remains shorter than the calibrated one.

### 5.3 Chaotic Motion

The last experiment combines the linear and circular movement with some quick rotations. It shows that the stability of the camera is the key for a successful tracking. Both experiments create several SLAM-Error for both the calibrated and the non-calibrated camera. As shown in Fig. 9 and Fig. 11 the calibrated

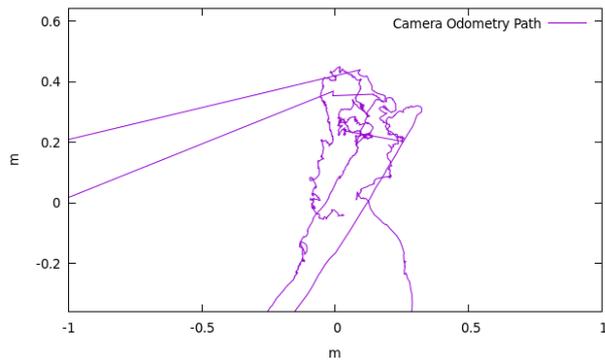


Figure 10. Tracking path for free movement with non-calibrated camera.

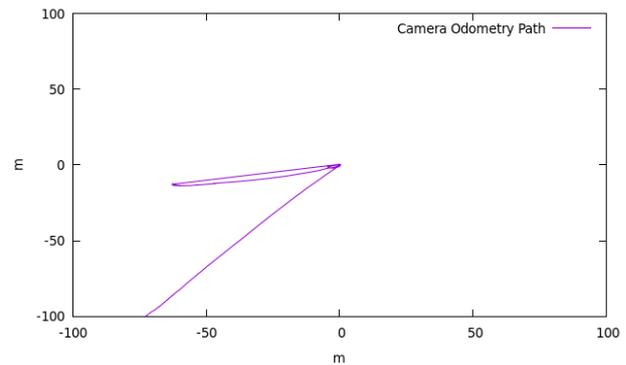


Figure 12. Zoomed out tracking path for free movement with non-calibrated camera.

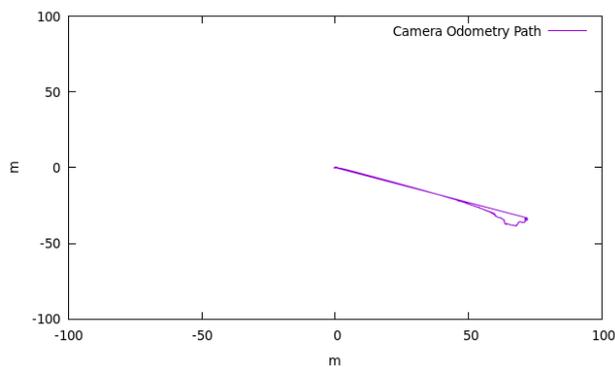


Figure 11. Zoomed out tracking path for free movement with calibrated camera.

camera shows a lot of SLAM-Errors but most of them get contained within a reasonable radius and resolve back to the trajectory. Only one time the SLAM-Error is so big that the trajectory drifts about 60 meters and has to be caught by the loop closing. The non-calibrated camera in Fig. 10 and Fig. 12 has even bigger problems dealing with the SLAM-Errors. Every SLAM-Error dooms the trajectory. There are two SLAM-Errors that get resolved by the loop-closing but the 3rd SLAM-Error is fatal. Without the proper calibration the error can not be stopped and the trajectory shifts about 100 meters away and remains far of the actual trajectory.

#### 5.4 Error Sources

The Main source of error are the SLAM-Errors created by the cheap IMU built into the T265. Earlier Version of T265 Software where even more unreliable, most of the bigger SLAM-Errors would result in a diverge into NAN values. Intel fixed some of the problems in newer software version but the problem still exists in rare cases. In the version used in this experiments most of the SLAM-Errors can be caught by a good camera calibration.

In total the calibration absorbs the refraction effects by changing primarily the focal length and distortion parameters. This makes for a significant change of tracking behaviour for the T265 camera. First data suggests that the camera calibration increases accuracy of the measurements. Using these measurements it is possible to track the trajectory reliable for short distances. This will enable the system to operate as a valuable

component for a future structured light mobile mapping system. But the approach also requires enough trackable features to work properly.

But there are additional error sources that have to be considered. The T265 camera's internal V-SLAM algorithm may fail resulting in wrong trajectory data. Unfortunately, this error cannot be fixed by the calibration as it is a software error in the Intel software that runs on the T265. This is checked using the IMU and mitigated by resetting the camera tracking. Consequentially the results show that the T265 is capable to operate as the localisation and tracking system for underwater applications.

#### 5.5 Conclusion

The Experiments show that the proper calibration of the stereo camera of the T265 has a stabilizing effect on the trajectories the camera can track. The fisheye camera model implemented by Intel for this camera is able to absorb the refraction at the air-glass and glass-water interfaces. Thereby the calibration can reduce the noise for the trajectories in a significant way and makes the T265 usable for underwater applications. There are just some small misalignments between the ground truth and the computed path left. But the limiting factor of the T265 is the inaccuracy with the IMU. This error leads to many error inside the tracking algorithm. The calibration can stabilize a lot, but not all of this SLAM-Errors. The main achievement is the consistency in scale that can be reached with the calibration.

#### 6. FUTURE WORK

Till this point the experiments using the T265 have been conducted in a very small scale. So the next logical step is to validate the usability of the setup in a larger scale and over a longer period of time. Most interesting is how the camera can scale over a longer path, what the optimal height over ground is and how fast the AUV can move without losing the trajectory. The functionality will be tested with help of a BlueRobotics BlueROV2. The laptop is then replaced by a small mobile processor unit, which will also be under water. A Raspberry Pi 4.0 was selected for this task.

#### REFERENCES

Abraham, S., Förstner, W., 2005. Fish-eye-stereo calibration and epipolar rectification. *ISPRS Journal of photogrammetry and remote sensing*, 59(5), 278–288.

Dunbabin, M., Usher, K., Corke, P., 2006. Visual motion estimation for an autonomous underwater reef monitoring robot. *Field and Service Robotics*, Springer, 31–42.

Kannala, J., Brandt, S., 2004. A generic camera calibration method for fish-eye lenses. *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, 1, 10–13 Vol.1.

Leonard, J. J., Bennett, A. A., Smith, C. M., Jacob, H., Feder, S., 1998. Autonomous underwater vehicle navigation. *MIT Marine Robotics Laboratory Technical Memorandum*, Citeseer.

Olson, E., 2011. Apriltag: A robust and flexible visual fiducial system. *2011 IEEE International Conference on Robotics and Automation*, IEEE, 3400–3407.

Taraldsen, G., Reinen, T. A., Berg, T., 2011. The underwater gps problem. *OCEANS 2011 IEEE - Spain*, 1–8.

van der Lucht, J., Bleier, M., Nüchter, A., 2019. A LOW COST UNDERWATER TEST ENVIRONMENT. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLII-2/W17*, 399–404. <https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-2-W17/399/2019/>.