DOI: 10.23784/HN116-10

Using a ROS-based low-cost system for bathymetric surveys

An article by LOUIS MAKIELLO

Using low-cost sensors and a software framework based on Robot Operating System (ROS) software, a low-cost multi-sensor system (MSS) was developed and tested. The system was successful in producing geo-referenced depth estimations. Potential applications include crowdsourced bathymetry. In addition, this MSS may be useful for hydrographic organisations with modest budgets, such as educational organisations and the national agencies of developing countries.

Robot Operating System | low-cost sensors | multi-sensor system | crowdsourced bathymetry
Robot Operating System | kostengünstige Sensoren | Multi-Sensor-System | Crowdsourcing-Bathymetrie

Unter Verwendung kostengünstiger Sensoren und eines Software-Frameworks, das auf der Software des Robot Operating System (ROS) basiert, wurde ein kostengünstiges Multi-Sensor-System (MSS) entwickelt und getestet. Mit dem System konnten georeferenzierte Tiefenwerte erfolgreich abgeschätzt werden. Zu den potenziellen Anwendungen gehört die Crowdsourcing-Bathymetrie. Darüber hinaus kann das MSS auch für hydrographische Organisationen mit geringen finanziellen Mitteln nützlich sein, wie z. B. Bildungseinrichtungen und Behörden in Entwicklungsländern.

Author
Louis Makiello holds a M.Sc. degree in Geodesy with specialisation in Hydrography from HafenCity University in Hamburg.

LouisMakiello@protonmail.com

Introduction

Bringing down the cost of hydrographic sensors and software has been the focus of research at the HCU's department of Geodesy and Geomatics in Hamburg in recent years.

The applications of such a low-cost system include crowdsourced bathymetry and use by hobbyists. In addition, universities and governmental agencies in developing countries would benefit from having access to hydrographic survey equipment at a fraction of the cost of current commercially available systems.

With low-cost surveying systems, the frequency and extent of hydrographic surveys could be increased. This is of obvious benefit in modelling and understanding water circulation in coastal areas, estuaries and inland waterways.

The low-cost hydrographic system presented in this paper makes use of open-source software and low-cost sensors. It has been designed to allow for additional sensors to be easily integrated. The ROS software framework used is popular in the field of robotics. It could be adapted for use with robotic vehicles. This fits the current trend of automation and artificial intelligence.

Educating future generations of hydrographers requires providing students with practical experience. This is best done using low-cost sensors whose loss or damage by inexperienced operators would not be a major financial blow to educational establishments. Open-source software that can be accessed and modified by students is also advantageous compared to proprietary software.

Only a small fraction of the earth's oceans and seas have been sampled for depth, greatly limiting our understanding and knowledge of the ocean. Mapping the ocean using hydrographic vessels is a costly and time-consuming endeavor. Specialised hydrographic vessels only represent a tiny fraction of the total number of seagoing vessels. A much larger number of vessels are equipped with single beam echo sounders (SBES) for navigational and safety purposes. If these SBES could be used to gather depth data, our knowledge of the seas and oceans could be much increased. Crowdsourced bathymetry has the potential to multiply worldwide depth sampling.

The Seabed 2030 Project of the Nippon Foundation and GEBCO aims to chart 100 % of the oceans by 2030 and has cited crowdsourced bathymetry as an option to help achieve this (Mayer et al. 2018). The IHO (International Hydrographic Organization) Inter-Regional Coordinating Committee has established a Crowdsourced Bathymetry Working Group (CSBWG). OpenSeaMap aims to make crowdsourced bathymetry available to the public (OpenSeaMap 2019).

Components

In selecting components, importance was placed on keeping down component purchasing costs. Also, components were to be assembled without the use of expensive workshop equipment. Finally, components were to function as much as possible with the use of open-source software.

Fig. 1 shows the main components and Table 1 shows component costs. Total cost is 985 Euros. For testing, the system was initially mounted on a rented »Kanadier« type canoe and subsequently on an inflatable canoe which cost 80 Euros.

Sensors

The sensors used include a single beam echo sounder, a GNSS module and a MARG (magnetic angular rate and gravity) array.

The echo sounder was an Airmar Echorange SS510 which operates on a frequency of 200 kHz. With a cost of 616 Euros, it is the most expensive single component. It outputs standard NMEA messages and could thus be substituted by any other NMEA echo sounder. The echo sounder outputs time of return and water temperature at the transducer head.

The GNSS module used was a NEO-6M which uses SBAS (satellite based augmentation system). It was set-up to output position at a rate of 5 Hz with a navigation engine optimised for use on a vessel.

The MARG array used was the MPU-9250, which has tri-axial accelerometers, gyroscopes and magnetometers. Its performance is enhanced by the use of temperature compensation for bias and scale factor. Commercially available higher-end MARG arrays are calibrated with the use of a turntable and thermal chamber. Such equipment may not be available to users of a low-cost system. Therefore, a simplified calibration procedure was carried out with the use of bubble levels and a hair dryer to manipulate temperature. Accelerometers and magnetometers are calibrated for bias and scale factor whereas gyroscopes are only calibrated for bias so as to negate the need for a turntable. Magnetometer scale factor calibration is done by sampling the earth's magnetic field with all three magnetometer and taking the average of the three as the true amplitude of the earth's magnetic field.

Timestamping

Timestamps for the MARG array are based on a PPS (pulse per second) signal from the EZ-0048 GNSS module, which uses atomic clocks on satellites as a reference. In case of loss of GNSS signal, a DS3231 real time clock is used to timestamp MARG array data. This real time clock is synchronised with GNSS time. The timestamps for the NEO-6M position are also based on atomic clocks. Echo sounder data is timestamped using the DS3231 real time clock. A single DS3231 communicates with both the Teensy microcontroller and the Raspberry Pi 3 single board computer via its I2C connection.

Processing

For processing, a Teensy 3.6 microcontroller and a Raspberry Pi 3 single board computer are used. The microcontroller runs a Madgwick algorithm to compute attitude at a rate of around 900 iterations per second while also performing timestamping

Fig. 1: Main components

Components	Costs [€]
Teensy 3.6	38.00
Rasperry Pi 3	34.45
Waveshare 5 inch resistive touch screen LCD(B)	39.99
GY-NEO6MV2 GNSS module	10.99
EZ 0048 GNSS module	30.99
MPU-9250 MARG array	5.99
DS3231 real time clock	4.87
RS-422 to USB converter	26.55
Serial Adapter DB9 female to terminal bloc	13.48
Airmar SS510 echo sounder	616.47
Fan and case for Rasperry Pi	8.99
Wireless keyboard and mouse	18.99
UPSPack power module	18.99
RAVPower 20,000 mAh power bank	26.99
LC-R127R2PG 12 V batteries	15.94
USB to TTL serial cable	3.54
Electronics supplies	20.00
Cadiz storage box	3.99
Cadiz storage box lid	1.99
Vacuum storage bag	5.99
Construction supplies	35.00
Total	**985.19**

Table 1: Component costs

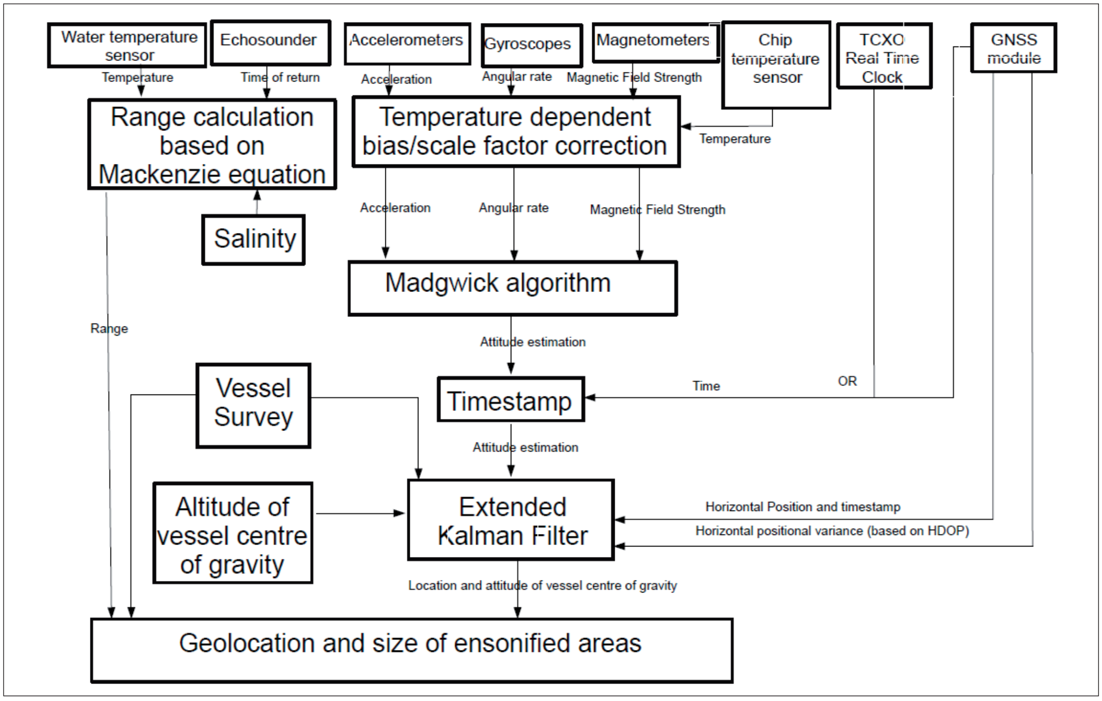

Fig. 2: System overview

and temperature compensation. The Raspberry Pi 3 runs ROS Kinetic on an Ubuntu Xenial operating system. Using a Raspberry Pi 3 eliminates the need for a Linux laptop or desktop computer as bathymetric pointclouds can be produced directly on the Raspberry although acquired data may be transferred to such a computer for post-processing.

System design and operation

To fuse data from multiple sensors, the Madgwick (Madgwick 2010) algorithm and the Extended Kalman Filter contained in the Robot Localisation ROS package (Moore and Stouch 2016) were used. Fig. 2 shows an overview of the system. GNSS data is converted to UTM and the surface of the water is modelled as a flat plane.

Filter tuning

The Madgwick algorithm's two filter gains of beta and zeta were adjusted. Zeta was set to zero as no gyroscope drift was detected for the MPU-9250. Beta was set to 0.6, a value larger than the 0.041 used in the original Madgwick study (Makiello 2019). This decreases the time needed for the filter to initially converge.

The Extended Kalman Filter adjustable parameters were tuned as follows: For the measurement noise covariance matrix R, sensor variances for attitude were set according to error estimations from static testing. The variance for horizontal position from the GNSS module was set dynamically according to the HDOP (horizontal dilution of precision) messages contained in the NMEA strings from the sensor. The process noise covariance matrix Q was tuned with the help of a test performed on land, with the system carried around a course measured on the ground that served as »ground truth«.

Water depth calculation

Depth is determined based on water temperature, salinity and the time taken for an acoustic pulse to travel from the vessel transducer to the sea floor and back. The Mackenzie equation (1981) is used for computation. Water temperature is measured at the echo sounder head throughout the survey.

Testing and results

Two tests were carried out: A first test in the Jaffe-Davids canal, an urban canal in Hamburg, and a second test in Eichbaumsee lake. Fig. 3 and Fig. 4 show the canal test. Fig. 3 shows the view in Rviz (Gossow et al. 2011), a robotics visualisation pack-

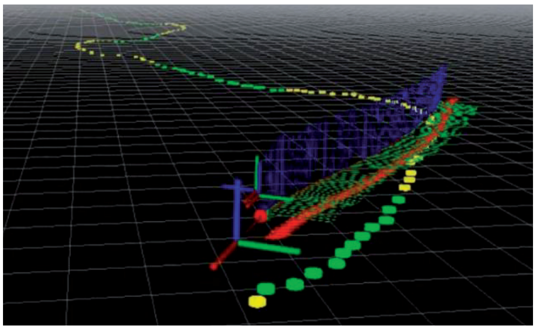

Fig. 3: Canal survey, Rviz view. Grid: 1 metre squares on the water surface. Yellow and green circles: ensonified areas. Red translucent arrows: GNSS horizontal position. Thick axes: positions of the echo sounder head, centre of gravity, MARG array and GNSS antenna (from bottom to top). Thin long axes: position and attitude estimates of vessel centre of gravity produced by EKF

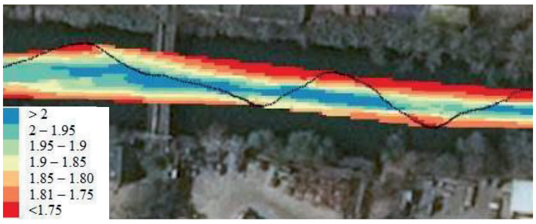

Fig. 4: Canal survey. Triangular Interpolated Network (TIN), 1-metre raster. Black dots: centres of ensonified areas

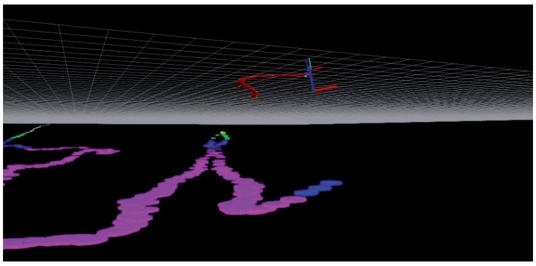

Fig. 5: Rviz view of Eichbaumsee survey. Grid: 1 metre squares on the water surface. The survey is viewed from underwater. The circular markers show the ensonified areas

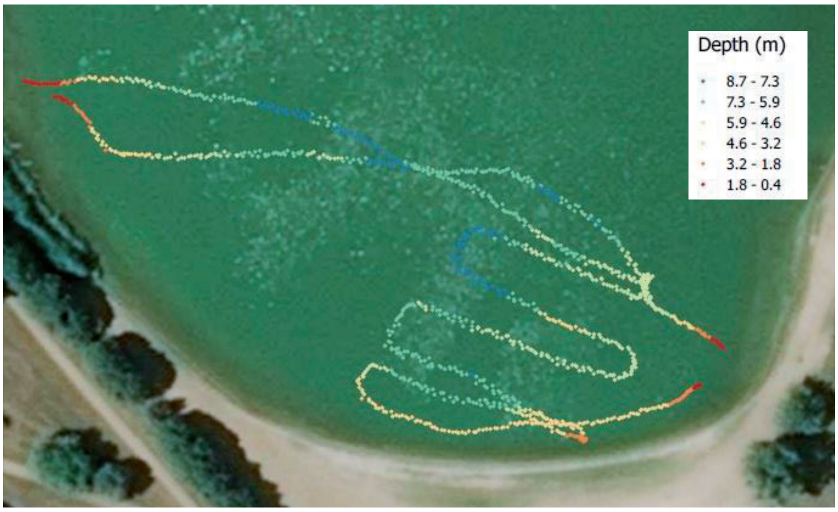

Fig. 6: Eichbaumsee survey

age, which produces a 3D view of the survey. Positions of sensors and sensor data as well as the motion of the vessel can be observed both during a survey and in post-processing. Fig. 4 shows a triangular interpolated network (TIN) overlayed onto satellite imagery of the canal.

To evaluate accuracy, cross line tests were performed. Examining the cross line tests, we found ten pairs of points with horizontal distances within 30 cm. The difference in depth between the two points in a pair was always less than 10 cm and usually only a few centimetres.

Fig. 5 and Fig. 6 show the survey conducted on the Eichbaumsee lake near Hamburg. Fig. 5 shows the Rviz view. A custom ROS node allows bathymetry to be displayed using markers of different colours depending on depth. The diameter of the markers is three times the ensonified area, which varies depending on depth.

The Raspberry Pi 3 used to run Rviz is susceptible to freezing if 3D visualisation is done during a survey due to insufficient processing power. Therefore, as an alternative, it is possible to record sensor data during a survey with processing and visualisation done after concluding the survey. Data acquisition can be monitored during a survey by viewing sensor data streams using the »rostopic« package. It was found that survey data could be collected non-stop for 34 days before an SD card (128 Gigabyte) swap was required.

Outlook

Integrating tide gauge data would allow for surveys to be carried out in tidal zones. The same feature could be used to convert depth relative to the water surface to absolute depths relative to a global datum.

Use of a sound velocity profile could, if available, increase accuracy of bathymetric data.

Visual odometry or visual-inertial odometry may be performed with low-cost sensors and would improve vessel positioning in GNSS-compromised areas. Initial tests using the ROVIO algorithm (Bloesch et al. 2015) have shown promise. //

References

Bloesch, Michael; Sammy Omari et al. (2015): Robust Visual Inertial Odometry Using a Direct EKF-Based Approach. IROS, DOI: 10.3929/ethz-a-010566547

Gossow, David; Adam Leeper et al. (2011): Interactive Markers: 3-D User Interfaces for ROS Applications. IEEE Robotics & Automation Magazine, DOI: 10.1109/MRA.2011.943230

Mackenzie, Kenneth V. (1981): Nine-term equation for sound speed in the oceans. The Journal of the Acoustical Society of America, DOI: 10.1121/1.386920

Madgwick (2010): An efficient orientation filter for inertial and inertial/-magnetic sensor arrays. http://x-io.co.uk/res/doc/madgwick_internal_report.pdf

Makiello, Louis (2019): Development of a ROS-based low-cost localization system for hydrographic surveys. Master thesis, HafenCity University Hamburg

Mayer, Larry et al. (2018): The Nippon Foundation-GEBCO Seabed 2030 Project: The Quest to See theWorld's Oceans Completely Mapped by 2030. Geosciences, DOI: 10.3390/geosciences8020063

Moore, Thomas; Daniel Stouch (2016): A Generalized Extended Kalman Filter Implementation for the Robot Operating System. Intelligent Autonomous Systems, DOI: 10.1007/978-3-319-08338-4_25

OpenSeaMap (2019): https://map.openseamap.org