



**HAL**  
open science

## Mobile Augmented Reality System for marine navigation assistance

Jean-Christophe Morgère, Jean-Philippe Diguët, Johann Laurent

► **To cite this version:**

Jean-Christophe Morgère, Jean-Philippe Diguët, Johann Laurent. Mobile Augmented Reality System for marine navigation assistance. 12th IEEE International Conference on Embedded and Ubiquitous Computing, Aug 2014, Milan, Italy. 10.1109/EUC.2014.49 . hal-01083265

**HAL Id: hal-01083265**

**<https://hal.science/hal-01083265v1>**

Submitted on 23 Jul 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Mobile Augmented Reality System For Marine Navigation Assistance

Jean-Christophe Morgère  
Universite de Bretagne Sud  
Lab-STICC  
France

Email: jean-christophe.morgere@univ-ubs.fr

Jean-Philippe Diguët  
CNRS, Lab-STICC  
France

Email: jean-philippe.diguët@univ-ubs.fr

Joahnn Laurent  
Universite de Bretagne Sud  
Lab-STICC  
France

Email: johann.laurent@univ-ubs.fr

**Abstract**—Augmented Reality devices are about to reach mainstream markets but applications have to meet user expectations in terms of usage and ergonomics. In this paper, we present a real-life outdoor AR application for marine navigation assistance that alleviates cognitive load issues (orientation between electronic navigational devices and bridge view) for vessels and recreational boats. First, we describe the current application and explain the requirements to draw relevant and meaningful objects. Secondly we present the software architecture of our pervasive system, which is compliant with different contexts and applications cases. Then, we detail our Marine Mobile Augmented Reality embedded System (MMARS). Finally, we present implementations on both Embedded system and smartphone.

## I. INTRODUCTION

Augmented Reality opens new perspectives to applications, more interactions between world and users become possible, environments can be enriched with virtual objects in many domains like geolocation, education, maintenance and medical. In the maritime domain, current augmented reality systems are made to limit accidents on vessels [1], [2], more precisely collisions and groundings. Recreational boats groundings and collisions with fixed objects are 33% of accidents in USA in 2012 (Table I) [3] like vessels accidents [4]. Our objective is the design of a mobile augmented reality system that runs an application for navigation assistance for both, recreational boats and vessels. It leads three kinds of constraints. First, the system must be low footprint with a lightweight, energy self-sufficient and hands-free solution, these points are addressed in Section II and Section III. Secondly, the system must be autonomous, MMARS is capable to fuse data and compute algorithms for application needs, this is explained in Section IV. Finally it must be pervasive, namely it must acquire available data from the AR device (e.g. GPS and Orientation) and from the maritime environment (boat devices), this is presented in the Section V. The implementation of our system is presented in Section VI, then we conclude and introduce our future works on power consumption and user feedbacks.

## II. MARINE APPLICATION

### A. Application context

Figure 1 is a picture from a vessel bridge view. There are three kinds of information visible on it. The first one is the seamarks, there is a channel represented by the red and green

TABLE I  
TOP FIVE PRIMARY ACCIDENT TYPES IN 2012 IN USA

Accident Rank	Accident Type	Number of Accidents	Number of deaths	Number of Injuries
1	Collision with recreational vessel	1010	47	711
2	Flooding/swamping	509	68	193
3	Collision with fixed object	475	50	340
4	Grounding	422	10	244
5	Skier mishap	387	20	388



Fig. 1. Sailor bridge view in Lorient harbour, France

buoys and a danger signal represented by a yellow and black buoy. All danger area are not visible from the bridge view but they are on nautical chart in North-up orientation. The second one is daymarks which are bell tower, water tower and lighthouse (also written on nautical chart), two of them are perceptible on the picture. The last information is the dynamic data represented by the passenger ship, data on vessels can be received by AIS or radar system represented on displays in head-up orientation. AIS system sends heading, boat speed, position and other data to prevent from collisions. To get a safe navigation the user must see buoys and know their meanings, look at the chart to see under water reefs (danger area) and he can also have systems to detect boats.

So on a boat, understanding of the position is vital when approaching sensitive environments such as coasts, open sea reefs or dangerous navigational channels. The current practices consist of going back and forth between map analysis and visual observations. Matching map indications with a real en-

vironment can be tricky and error-prone, and it also represents a loss of time that is precious in case of emergency. In case of fog, night or bad weather matching is very difficult or impossible and stress is the most causes of error decision. It will be safer if the most important information were displayed in the bridge view [5] and more precisely in front of the user eye. But the boat is a uncomfortable case environment since a ship is a very unstable system. All the continuing motions have various parameters depending on boat speed, user moves and ocean oscillations. For example swell periods can vary between 0.05Hz and 0.1Hz. So the challenge is to get an application that displays the most important information in accordance to the maritime rules in a bridge view in the user's field of view.

### B. Ergonomic requirements

The first constraint is the choice of objects to display; in the case of navigation assistance, one of the most important ones is the seamarks. Two parameters can change application needs and objects to draw: user types (depending on vessel types or recreational boat types) and ergonomics. Like Porathe's application [2] or ARVCOP Technology system [1], information for vessels are AIS data and radar that help to prevent collision risks, vessels channel to show the correct way and NOGO area depending on the water height. But rapid passenger vessels have other constraints such as water height under hull, speed (speed is about 30 or 35 knots) and navigate in coastal waters, whereas some of recreational boats don't have any AIS receiver or GPS on-board. Services offered by the application have to take into account of maritime rules and user habits to prevent with alerts the grounding and collisions risks. Let's consider the compliance with the standard representation of the International Association of Lighthouse Authorities (IALA) Maritime Buoyage System [6] that defines one system of marks. A maritime buoy (or beacon) for instance has three elements: shape, colour, a top mark and depending on the buoy an emitted light that dissociates the type of danger and shows the right way. Due to the user habits, we have kept the real shapes and colours (except for black colour which is the transparency for augmented reality) of buoys which are visible in Figure 2. In addition to those objects some arrows are displayed for instance to prevent the user from danger in case of bad visibility. The mobile system must provide a variable luminosity that is efficient enough to handle extreme luminosity variations from a shining sea to a dark night. The device must be also able to work offline, be lightweight, self-content and easy to wear (low footprint). Ergonomics is also very important for acceptability and efficiency, it would be counterproductive to display all possible objects on the display because it's very painful to decode and interpret a lot of information. Some feedbacks from users could allow us to offer a representation for objects and superimpose, in addition of buoys, lighthouses, squares or arrows like in Figure 3. The number of visible objects is limited to a given frustum by a near and far plane as depicted in Figure 4. This is the field of view that must be tuned according to user choices,

moreover the maximum number or group of objects needs to be fixed. Some arrows are also added to indicate the presence of surrounding boats equipped with AIS and will be added for objects out of the field of view.

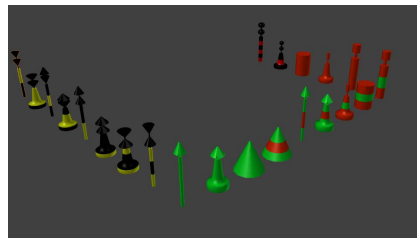


Fig. 2. 3D buoys for sailor assistance (black background for transparency on LASTER see-through glasses)

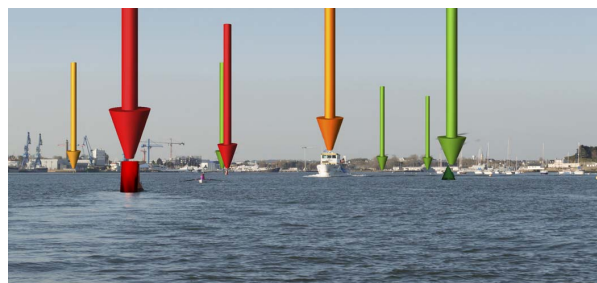


Fig. 3. Sailor Assistance AR case study. E.g. Buoy (GPS+Digital Chart) and Cruising Boat (AIS+GPS) positioning

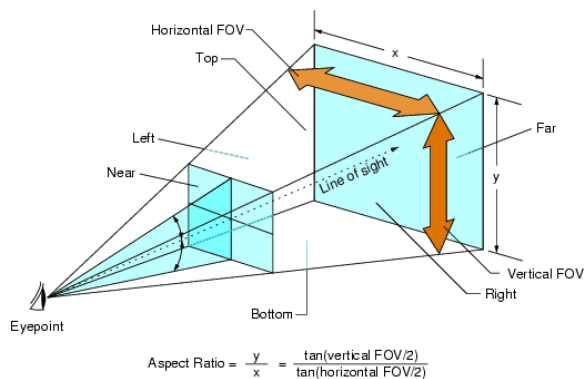


Fig. 4. Field Of View

### III. MMARS

In case of emerging products in the mainstream market of augmented reality<sup>123</sup>, wireless links are used to send data, such

<sup>1</sup><http://www.google.com/glass/start/>

<sup>2</sup><http://optinvent.com/see-through-glasses-ORA>

<sup>3</sup><http://laster.fr/products/seethru/>

as positions data from a smartphone (with GPS chip) to the AR glasses. The AR glasses might not be used alone in geolocation applications, only orientation can be used according to the MEMs or the camera. This is why, according to our objective, which is a low power, low footprint autonomous wearable solution, we have designed our own embedded system as shown in Figure 5. Our prototype is based on a Gumstix Duovero Zephyr[7] plugged (as a motherboard) to an ad hoc extension board (daughterboard), that we have specifically designed. This motherboard implements a System on Chip (SoC) that allows to compute 3D rendering on the GPU using the vertex and fragment shaders and to execute concurrently separate tasks, such as acquisition and data fusion, on the dual core processor. Moreover the CPU frequency can be set between 300MHz and 1GHz according to the power management policy. The resolution and the frame rate have to be in accordance with the display that is used but the frequency can be downgraded from 60Hz to 30Hz to limit the power consumption. An important memory storage is required to store OS, drivers, 3D objects, charts and application files, so we use the Secure Digital card slot on the board. For example the buoys shown in Figure 2 require 1MB, four 3D boats take 500kB in very low quality and no textures and a NOAA chart from San Francisco to Cape Flattery requires 36Mo. Finally the board implements wireless connectivities: Bluetooth 3.0 and WiFi 802.11(b/g/n) to communicate with the boat environment. Our system doesn't need power greedy wireless connection for display update and to get geolocation and orientation estimation since all computing tasks are embedded. For instance if we consider the case of recreational boats without any devices on board, MMARS can be fully autonomous if the on-board GPS is activated and if nautical charts are pre-loaded. In case of boats equipped with GPS and AIS devices a low data-rate Bluetooth connexion can be established with MMARS.

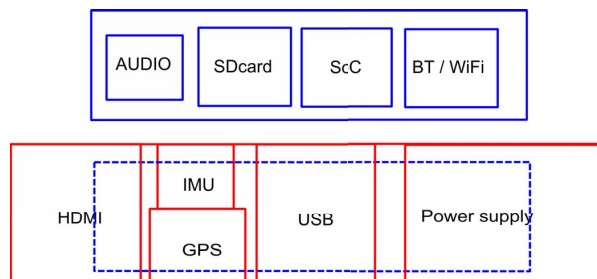


Fig. 5. Layout of MMARS motherboard and its daughterboard

We have implemented on the daughterboard an InvenSense IMU (MPU-9150<sup>4</sup>) to acquire data and compute the head orientation with a 9 DOF (Degree of Freedom) device that combines a 3D magnetometer, a 3D accelerometer and 3D gyroscope MEMs (MicroElectroMechanical systems). The MEMs, in addition with a filter, is one of the best solution

<sup>4</sup><http://www.invensense.com/mems/gyro/mpu9150.html>

for pose estimation in marine navigation because of the boat environment: the weather conditions, the user, the boat movements and the depth field of the scene. We didn't choose a camera-based approach, which is not adapted to our context. First, in case of direct or reflected sunlight on the lens, the picture cannot be directly exploitable because of the glare. In addition to the glare, lens flare is caused by a reflection on the lens due to a little angle between the light source and the lens. Some rings or circles may appear on the picture and induce image processing to correct it. Second, in night condition we could use an infrared camera but unfortunately we would still require a normal camera (like a smartphone camera) as well that increase the price of the system and introduce more image processing to decode frames extracted from the infrared camera. Third, in case of fog, some additional image processing has to be performed to eliminate noise [8], but it requires large computations capacities and far objects might not be perceived anyway. Finally, in normal condition (with no disturbance), a scale computation must be done on frames because of the camera FoV which is different from a real size scene. The use of the camera induce an image processing rate at 30Hz which is the minimum frequency required to capture user's head movement, and the data flow vary with the camera resolution. For instance, a 720p resolution is about the double of bytes to be treated in 33ms than a SVGA resolution. Maritime navigation is a wide open space, the camera must acquire small elements in far distance so a HD camera is needed to keep those information. Finally the considered objects are far by definition and the accuracy of the positioning is not critical. So, the incremental cost of image processing compared to the results and our application context don't justify the use of camera in MMARS.



Fig. 6. Laster see-through glasses and MMARS board

In our experiment setup, the prototype is directly connected to see-through glasses from Laster Technologies shown in Figure 6. The model we use is a modified MG1 mask[9], which provides a 800\*600 pixels display of 50 degrees of FOV in diagonal and a maximum brightness of 5000cd/m<sup>2</sup>. This lightness is enough for outdoor applications in shiny environments, whereas mainstream AR glasses are about 3000cd/m<sup>2</sup>, which could be not enough in such conditions. The maximum power consumption of the see-through glasses is 180mA at 7.2Vdc (1.3W) whereas DuoVero Zephyr running a Linux console (kernel 3.6.0) without any applications and no specific power

optimization requires 250mA at 5Vdc (1.25W) see Table II. This consumption increases up to 430mA with a graphical application and orientation computing (2.25W).

TABLE II  
POWER CONSUMPTION

System	Parameter	Power Consumption (mA)
Prototype v1	Graphical Linux, no application	275
Prototype v2	Graphical Linux, no application	260
Prototype v1	Linux Console, no application	251
Prototype v2	Linux Console, no application	240
Application	No BT, no WiFi	180
Laster MG1	85Hz, 5000cd/m <sup>2</sup>	180

#### IV. SOFTWARE ARCHITECTURE

##### A. Computation requirements

Multiple software tasks are required to design and position the graphical objects. The first one is in charge of the loading and removing of objects when they appear or disappear from the user's field of view, it means that the software must have access to objects, to their GPS positions and to 3D specifications. The second one is a set of tasks that includes the computation of object orientations based on the head position. It first requires current GPS data that can be provided by the boat (or the system) and secondly an IMU that must be embedded in the glasses to obtain the user's field of view with a fusion data algorithm. The third must be used to decode AIS or radar data (when it is possible) and detect potential collisions or NOGO area approaches. Finally the last one is in charge of the graphic pipeline: 3D/2D projection, Z-buffering, lighting and frame buffer updates.

##### B. Algorithms

In our case, the mobile Operating System is an embedded Linux that offers a large amount of APIs and drivers. In augmented reality applications, the main functionality is the graphic task, since the system can draw 2D/3D objects with colours or textures. In embedded systems with mobile OS, such as Android or Linux, the OpenGL ES2.0 API allows the programmers to access a lot of 3D functions such as rotate, scale, translate, texturing, lighting and material rendering, according to the vertex and fragment shaders. The next task is the attitude computation or head tracking. This step is necessary to place the right objects in the right orientation on the display.

A special task with MEMs is implemented, we want to two kinds of profiles, a low consumption profile and a high accuracy profile to get the choice depending on the needs. So we have implemented the following filters: 7 states Kalman Filter with prelinearization [10], 7 states EKF (Extended Kalman Filter) [11] and a Gradient Descent algorithm [12]. Currently we have to realize more tests to improve the filter for accuracy profile. In addition, MEMs calibrations are required for the magnetometer and the accelerometer due to their offsets and perturbations; it is possible to calibrate those sensors

with Kalman filters [13], [14]. A wireless task has also been developed to acquire data from the boat network (GPS, AIS, etc.) according to the Bluetooth or WiFi connections. NMEA (National Marine and Electronics Association)<sup>5</sup> is the most commonly used data format in marine electronics, so GPS, AIS and radar have this data format. After detecting the header and decode an AIS sentence, we prevent the user from a collision risk [15] by displaying a warning message.

To detect collision risk and manipulate objects positions, the world geodetic system WGS84 [16] in degree minute second format is not the best. To introduce them easily in a virtual world in the augmented reality application, we make an UTM (Universal Transverse Mercator) projection [17] on the coordinates to place objects in metric units on a plane. UTM coordinates system is a conformal projection.

#### V. PERVASIVE SYSTEM

We propose a new approach for the application assistance, we have designed it as a services server. This server currently offers five kinds of services:

- S1: GPS functions
- S2: AIS functions
- S3: Radar functions
- S4: Electronic Navigational Charts (ENC) functions
- S5: Performance data functions

S1, S2 and S3 services are related to positioning devices, S4 gives access to nautical charts and the last one S5 is related to on-board sensors to evaluate the boat performance. Three information are extracting from S1: position, COG (Course Over Ground) and SOG (Speed Over Ground), this is the main service, no applications could be run without boat position. S1 is able to offer sufficient information to run the application that displays in alphanumeric format the data related to position, speed and course. S2 is required for safe navigation in busy areas, it is used to prevent collisions with other boats that emit AIS data according to the availability of S1. Based on S2 data, the application is able to display information about surrounding boats as presented in Figure 7 and can compute collision routes, time to impact and can raise alerts. On this figure some boats are plotted thanks to AIS data and an alert is sent to the user in case of collision route, also arrows are displayed above boats positions to give more visibility to far boats. S3 provides specific ARPA (Automatic Radar Plotting Aid) services based on radar data, it includes the closer target (boats or other objects like emerged reefs) position (CPA) and the time to target (TCPA) as well as the course and speed of the detected objects. S4 is a service able to load and extract data from a 3D ENC (written before the application runs) and gives all graphical elements to the task in charge of displaying 2D/3D information in the navigation assistance; an example of the first tests is visible in Figure 7. The Buoys and arrows represent a channel from Lorient harbour, there are lateral marks in red (port marks) and green (starboard marks) according to the IALA representation in Region A and an

<sup>5</sup><http://www.nmea.org/>

isolate danger mark with black and red bands (only its arrow is visible due to the distance). S5 has a list of sub-services that are activated or not according to the sensors plugged to the navigation processor. A navigation processor on a well-equipped sailing boat typically provides data such as apparent and true winds, the boat speed over water, heading, roll and pitch. Those information can be display like in the Figure 8 as it was presented in [18].

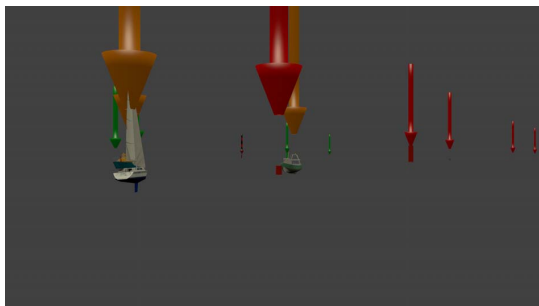


Fig. 7. Navigation application (C) - Services: S1 + S2 + S4 (black background for transparency on LASTER see-through glasses)

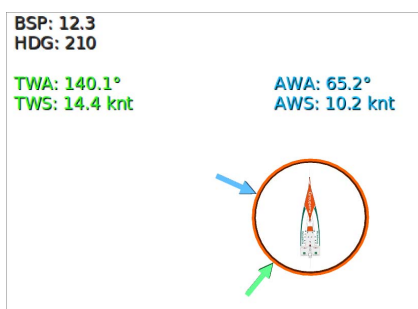


Fig. 8. Regatta application (B) - Service: S5 (smartphone version)

All services depend on the environment boat, indeed the services for recreational boats without any devices are not the same than fully-equipped ferrys for instance. So the system must be adaptive, at start time, it scans the environment to detect which services are available on the boat. Then a single or multiple applications are proposed according to available services and user profile as depicted in Figure 9. If no services are detected on the boat and the ENC service is not available, the system use its GPS device and run the simplest orientation application. The typical example for recreational boats we consider includes GPS, AIS devices and ENC data, so our system detects the S1, S2 and S4 services, collects data from ENC, build a 3D map and starts a navigational application with seamarks and information about boats in the vicinity.

## VI. IMPLEMENTATION

MMARS has two wireless connectivities to get services on the boat: Bluetooth and WiFi. Our system is able to scan Bluetooth devices and Wifi server on the boat and connects to

the server that runs on the host. Based on these connexions MMARS acquires data in a NMEA format except form ENC data, which are updated though requests when necessary. After decoding the NMEA header sentences the system is able to list which services are available. The first service to start is the GPS (S1), because we need to know the boat position in order to scan according to the ENC service, if there are ENC data corresponding to the navigational area on the system memory. If there are no data, ENC service uses the Wifi or Bluetooth on board (if it is available) and requests a download to the ENC server (if a server is available) and updates the system. A web connection can be used, so S4 is able to request a chart download from a web server. An easiest solution is the use of a SD card, the user just have to copy electronic charts on the memory card and plugs it to the MMARS.

We have tested two user profiles, the first one is related to a race sailing boat and the second one is a recreational boat environment. In the first case, the connexion is based on the Bluetooth RFCOMM protocol between MMARS and an open navigation processor as proposed in [18], NMEA sentences are decoded and displayed by the system according to the sailor expectations (Figure 8). The second test is based on a laptop with a RFCOMM server and a Wifi that broadcast GPS and AIS data to the system. MMARS is able to find devices on both connections and lists services to the user. After a check and a validation by the user, the system offers choices about applications that can be started.

We decide to compare our MMARS system to a mainstream AR device. Our application is currently ported to Android with the goal to compare first the ergonomics and use of the glasses. Second, this stage will allow us to validate the solution of an all embedded system for navigation assistance against the mainstream solution, which consists of using both glasses and smartphone jointly. The main advantages of having only one device would be that latency is kept to a minimum, and that energy consumption is optimized. Today the system alone (without glasses), with a graphical animation, 9 DoF running and a Bluetooth communication for GPS and AIS data has a power consumption about 2.25W (Table II). In the current version the power management policy is simple but are working on some more aggressive strategies with the objective to decrease the MMARS power consumption down to 1.25W. The porting of the application on Android is made easier by the fact that both OS share a Linux Kernel. Drivers for the i2C, the GPU or the Wireless connexions are identical and both have OpenGL ES support. However, Android has higher level of abstraction, with managers which allows an easier use of the components but it is less optimised in computations, so it increases the power consumption. A native development kit enable us to program in C and C++ in order to write low-level applications (which have access to the drivers). The graphical part is managed with a free open source 3D engine written in C++. All tasks not related to the hardware such as NMEA decoder, collision detection, MEMs filters are exactly the same on the two systems. It allows the possibility to port the application in a reasonable amount of time.












Services	Platforms	Interfaces	MARS	Application cases
 GPS  Navigation Processor  AIS  RADAR  Charts	Pleasure boating  Ocean racing  Professionals 	No connexion  		Orientation Application (A) Regatta application (B) Navigation application (C)

Fig. 9. Pervasive environment: depending on services available and according to the interfaces on platforms MMARS offers multiple application cases

## VII. CONCLUSION AND FUTURE WORK

In this paper we presented a new solution for a navigation assistance which is implemented on a Marine Mobile Augmented Reality embedded System. This solution is pervasive since it can adapt to available services scanned on the boat allowing the user to get the most appropriate application depending on the available services. Three kinds of applications are currently proposed to the user but some ergonomic improvements are still under studied with a panel of professionals and boaters. We are also going to implement those applications on a smartphone connected to a mainstream system to compare see through AR glasses technology.

Our future work is organised as two directions. First, the next step is a real life test to be held in the Lorient harbour with both (MMARS and smartphones) systems simultaneously used on a boat. The aim is the validation of the objects to draw during real conditions with a user panel (recreational boats and professional) and then, get user feedbacks. These tests will be used to refine ergonomics, to adapt the relevant number of objects and to detect the missing data for a usefull application. Secondly, we will work on a very aggressive power management policy since the objective is to reach 8 hours of activity with a standard smartphone battery.

## REFERENCES

- [1] ARVCOP, <http://www.arvcop.com/>.
- [2] T. Porathe, *3-D Nautical Charts and Safe Navigation*, ser. Mälardalen University Press dissertations. Mälardalen University, 2006, <http://books.google.fr/books?id=NqCcNQAACAAJ>.
- [3] US Coast Guard, Department of Homeland Security (US). 2012 Recreational Boating Statistics 2013.
- [4] J. Law, "Marine casualty analysis: We keep history from repeating itself," *Coast Guard Journal of Safety & Security at Sea, Proceedings of the Marine Safety & Security Council*, vol. 69, no. 3, 2012.
- [5] J. Prison and T. Porathe, "Navigation with 2-d and 3-d maps: A comparative study with maritime personnel," *Proceedings of the 39th Nordic Ergonomics Society Conference*, pp. 1–3, 2007.
- [6] F. A. Pielou, "The introduction of iala maritime buoyage system a," *Journal of Navigation*, vol. 31, pp. 422–425, 9 1978.
- [7] GUMSTIX, <https://store.gumstix.com/index.php/products/355/>.
- [8] LYYN, [www.lyyn.com](http://www.lyyn.com).
- [9] Laster Technologies, <http://www.laster.fr/produits/MG1/>.
- [10] D. Comotti, "Orientation estimation based on gauss-newton method and implementation of a quaternion complementary filter," Tech. Rep., 2011.
- [11] J. Marins, X. Yun, E. Bachmann, R. Mcghee, and M. Zyda, "An extended kalman filter for quaternion-based orientation estimation using marg sensors," in *Intelligent Robots and Systems. IEEE/RSJ International Conference on*, vol. 4, 2001, pp. 2003–2011 vol.4.
- [12] R. V. Sebastian O.H. Madgwick, Andrew J.L. Harrison, "Estimation of IMU and MARG orientation using a gradient descent algorithm," *IEEE International Conference on Rehabilitation Robotics*, 2011.
- [13] P.-F. Guo, H. Qiu, Y. Yang, and Z. Ren, "The soft iron and hard iron calibration method using extended kalman filter for attitude and heading reference system," in *Position Location and Navigation Symposium (PLANS)*, 2008.
- [14] I. Frosio, F. Pedersini, and N. Alberto Borghese, "Autocalibration of MEMS accelerometers," *Instrumentation and Measurement, IEEE Transactions on*, vol. 58, no. 6, pp. 2034–2041, June 2009.
- [15] L. P. Perera and C. G. Soares, *Detections of potential collision situations by relative motions of vessels under parameter uncertainties*. Taylor & Francis Group, London, UK, 2012.
- [16] D. T. Sandwell, *Reference Earth Model - WGS84*, Scripps Institution of Oceanography, 2002.
- [17] D. Agency, "The universal grids: Universal transverse mercator (UTM) and universal polar stereographic (UPS)," Defense Mapping Agency, Hydrographic/Topographic Center, Fairfax, VA, USA, Tech. Rep. TM8358.2, 1989.
- [18] R. Douguet, J.-C. Morgère, J.-P. Diguet, and J. Laurent, "Coupled open navigation and augmented reality systems for skippers," in *International conference on innovation in high performance sailing yachts (Innov'Sail)*, Lorient, France, 2013.