



HAL
open science

Real-time infinite horizon tracking with data fusion for augmented reality in a maritime operations context.

Olivier Hugues, Jean-Marc Cieutat, Pascal Guitton

► To cite this version:

Olivier Hugues, Jean-Marc Cieutat, Pascal Guitton. Real-time infinite horizon tracking with data fusion for augmented reality in a maritime operations context.. *Virtual Reality*, 2013, 17 (3), pp.10. 10.1007/s10055-013-0234-9 . hal-00787072

HAL Id: hal-00787072

<https://hal.science/hal-00787072>

Submitted on 11 Feb 2013

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.

Merging Video and Charts for Augmented Reality by Vision and Inertial Fusion to Perform Horizon Tracking.

Olivier Hugues · Jean-Marc Cieutat · Pascal Guitton

the date of receipt and acceptance should be inserted later

Abstract In this paper we propose a robust method for real time detection of the horizon (ie. separation line between the coast and sea) in a maritime context whose ultimate goal is to combine a video stream and electronic charts. We present the combination of an image analysis algorithm with information from an inertial unit. The initial aim is to filter out environmental conditions and noise. The second aim is to detect the visible horizon using both a classic video stream and a thermal video stream taking into consideration the non-negligible presence of this type of camera on board vessels. We then present an assessment of the algorithm's performance with regard to the rate of detection of the horizon and the impact of the image's resolution on the calculational loop's execution time. The purpose of developing this method for detecting the visible horizon is to create an indirect vision functionality in augmented reality. We want to combine information from charts and the video stream in a single display. We use the information regarding the horizon's position in the image to divide it into different areas whose transparency enables either the video or charts to be given priority.

Keywords: image processing, sensor fusion, mixed reality, augmented reality, electronic charts system, geographic information system.

1 Introduction

Like in aviation, embedded computing on board vessels is increasingly present. Although computer aided navigation software (ECS¹) is still the subject of debate as opposed to traditional paper charts, the adoption in 2009 by the IMO's [1] maritime safety committee of amendments introducing the rule obliging all vessels on international voyages to have electronic chart systems confirms this development. Despite various different technological developments, there are always needs for specific developments for this type of target. These tools are however no longer limited to managing charts, but can be considered true onboard computers centralising all the data from sensors and embedded navigation tools (radar, AIS², ARPA³, sondeur, sonar, etc) [2,3]. Recent progress in the field of thermal sensors has led to manufacturers installing this type of equipment on vessels [4]. However, as with many in-situ help tools, it is essential to take into consideration the context of use [5,6] especially when both complex and dangerous situations may be encountered by people (man overboard, injuries, etc), equipment (running aground, deteriorations, loss of cargo, etc.) and the environment (oil slick, submerged polluting substances, etc). The increasing awareness of governments with regard to sea environment related issues [7] is leading maritime soft-

Olivier Hugues
ESTIA Recherche - MaxSea - LaBRI (INRIA),
E-mail: o.hugues@net.estia.fr

Jean-Marc Cieutat
ESTIA Recherche,
E-mail: j.cieutat@estia.fr

Pascal Guitton
LaBRI-IPARLA(INRIA), Bordeaux 1 University,
E-mail: guitton@labri.fr

¹ Electronic Charts System.

² Automatic Identification System : automated tracking system for identifying and locating Vessels.

³ Automatic Radar Plotting Aid : radar feature can tracking others vessels with echo radar.

ware publishers to adapt their offers in order to take into account these environmental characteristics by using, for example, emerging technologies. These tools for aiding navigation therefore have certain advantages such as improving environmental knowledge, reducing the sailor's workload and even improving performances with the aim of finding navigation solutions, responding to safety problems or proposing more economical choices [8]. However, using this type of tool is not exempt from errors and difficulties. In [8], the authors identified three error categories when using the ECS tool. The first is human errors, the second equipment errors and the third procedure or management errors. According to an IMO report [9], 90% of maritime accidents are caused by errors from the first category. In this respect, to limit problems of interpretation, we are working on reducing the "difference" between perceiving and understanding the natural environment in which the sailor develops and the tool used for aiding navigation by using an augmented reality functionality.

To achieve this, we use embedded sensors (GPS and inertial unit) to georeference video streams from the camera in the ECS 3D virtual environment. However, the movements of vessels are just part of the disruptive elements whose frequency ranges [10] mean that inertial units cannot correctly apprehend these movements [11], hence the choice of our multi-sensor approach. To enhance the precision of registration and the "mix" between the video and charts, we therefore also propose detecting the horizon.

2 Related Work

In the literature, two approaches can be identified for carrying out this detection. The first consists of segmenting the image into regions (region-based) whilst the second involves detecting the contour (edge-based). For the same reasons as in [12], the first approach is not efficient in our context due to the large variation of the light intensity in different scenes. The authors therefore propose combining visual light, close infrared, mid-wave infrared and long-wave infrared cameras. In [13], the authors propose a vision-based approach using a multi-layer linear analysis of a non-thermal video stream to detect the horizon with a view to piloting a micro air vehicle whose statistical algorithm is based on the work of [14]. The authors of [15] propose extracting from a visible horizon based on images containing fog. The execution time is relatively long and not compatible with a real time functionality. Certain authors use machine learning [16] to segment the sky and the earth. However, this solution is dependent on the number of images

enabling machine learning and the diversity of situations and exposures complicate this learning. In [17], the authors propose an approach to extract the horizon from maritime scenes. They propose merging nine colour channels enabling the impact of variations in luminosity to be limited.

Combining an analysis of contours and colours is proposed in [18] enabling the detection of lines to be significantly improved. In a technical report [19], the author proposes a method for image analysis enabling the rolls of an airplane to be known based on a sequence of images of the horizon. The performance of this solution is however not acceptable since it is limited to one image per second. Extracting the horizon and mountain summits proposed in [20] used infrared images to determine the pixels which create the horizon (horixels). This solution is interesting but not functional for non-thermal video streams. Current knowledge concerning gyroscopes enables us to use this type of sensor to improve the robustness of vision-based methods. In this respect, augmented reality (AR) functionalities, which require a high level of precision in the registration, are excellent candidates for using the multicriteria approach (vision, gyro, etc). In [21], the authors propose for example the combined use of a video camera and a gyroscope to improve the follow up of markers for an augmented reality functionality.

3 System's Details

In the system, whose organisation is presented in Figure 1, the input image is in colour (from the visible domain to infrared) and produces a right segment output whose position is defined in screen-pixels by its two end points (x_0, y_0) et (x_1, y_1) . This right segment is then used by an AR component to define the areas in the image. These areas will enable the component to define the way in which the video will be "mixed" with electronic charts in the analysis of the sea, coast and sky.

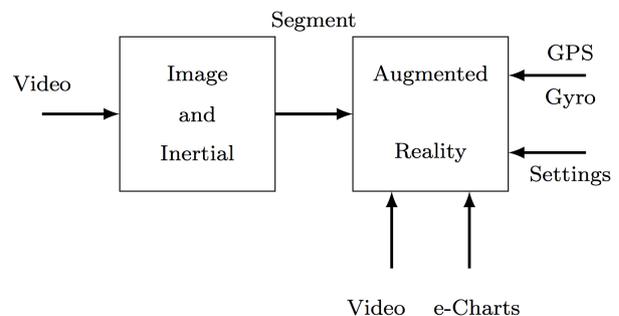


Fig. 1: System's organisation.

3.1.2 Dynamic operators

The definitions for measurement operators are not identical to “box” operators since there may be assembly issues with these boxes on the vessel for example. The sensor’s manufacturer defines the reference point for which the measurements will be expressed and there is no reason for this reference point to be identical to the reference point which enables the sensor to be positioned on the vessel:

- ${}^W T_{GPS_m}$: GPS frame measurement in the world. This operator represents a pure translation on only two components (altitude data provided by the GPS cannot be used because of its lack of precision);
- ${}^W T_{YPR_m}$: frame of the inertial unit’s measurement. This operator represents a pure rotation.

3.1.3 Camera model

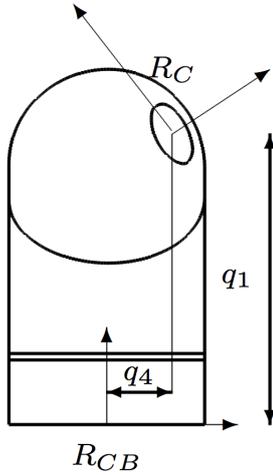


Fig. 5: Geometric camera model.

We have chosen to define the geometric model of the camera (Figure 5) by five transformations enabling switching from the camera base’s frame (R_{CB}) to the camera head frame (R_H). Since the operator enables the camera head’s position in its base to be known it is therefore split into five transformations as follows:

1. A q_1 translation (camera height);
2. A q_2 azimuth rotation;
3. A q_3 elevation rotation;
4. A q_4 translation (lens shift);

5. A q_5 roll rotation (the camera does not physically have this degree of freedom. We have nevertheless modelled the camera so as to use this degree of freedom later).

So, the camera model:

$${}^{CB} T_C = Tr_{q_1} \cdot R_{q_2} \cdot R_{q_3} \cdot Tr_{q_4} \cdot R_{q_5}$$

3.2 Calculations

We must now calculate the camera head’s frame in GC, and therefore, the camera base’s frame for the same reference point.

$${}^{GC} T_{CB} = ({}^{VF} T_{GC})^{-1} \cdot {}^{VF} T_{CB}$$

$${}^{GC} T_C = {}^{CB} T_C \cdot {}^{GC} T_{CB}$$

We must now know the GC frame in the world. The position of this reference point therefore depends on the four following operators:

1. GPS measurement (${}^W T_{GPS_m}$);
2. Inertial Unit measurement (${}^W T_{YPR_m}$);
3. IU correction (in the case of the unit’s box is not perfectly positioned);
4. GPS correction (in the case of the GPS box is not perfectly positioned).

$${}^W T_{GC} = {}^W T_{GPS_m} \cdot {}^W T_{YPR_m} \cdot {}^{GC} T_{YPR} \cdot {}^{GC} T_{GPS} \quad (2)$$

We now have all the operators required to know the camera’s position in the world.

$${}^W T_C = {}^{GC} T_C \cdot {}^W T_{GC}$$

We shall extract from this operator the values required to carry out the algorithm presented in the following section.

3.3 Inertial Tracking

Knowledge of the camera’s calibration matrix and its position in the previously calculated world reference point enable the inertial unit to be used [22] so as to limit the exploration area of our image analysis as illustrated in Figure 6. We consider, in its initial state, that the camera is positioned horizontally with regard to the water. The inertial unit’s angular information enables us to assess the height in pixels of the horizon in the image. We therefore define an area around this height encompassing the horizon with a red rectangle as illustrated in Figure 6 whose width is the width of the image. The height of this strip is defined by $\alpha + \epsilon$, with $\epsilon = \pm 5^\circ$. For example, for an image resolution of 640x480, the height of the strip is 24 pixels (only 5% of the total image’s pixels).

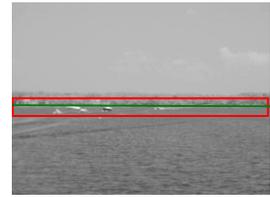


Fig. 6: Limiting the image analysis area by inertial unit.

Using the inertial unit to limit the area of exploration in our image analysis (Section 3.4) has two advantages. Firstly, reducing the number of pixels to be processed by our analysis phase enables performances compatible with real time execution and embedded equipment whose hardware resources are limited to be obtained. In fact, since there is a small number of pixels to be processed, the algorithm’s performances authorise a minimum frequency of 32 images per second for the highest resolution tested (Section 4). Secondly, we limit the risks of detecting horizontal lines at the surface of the water and in the sky. An example is given in Figure 7 showing the advantage of reducing the detection area.



(a) Image processing on frame 47 without gyro tracking. A wave is detected as the horizon.



(b) Image processing on frame 47 with gyro tracking. The wave is not detected.

Fig. 7: Illustration of the advantage of reducing the detection area. Example for a thermal image.

3.4 Image Processing

The aim of this image analysis is to detect the horizon from an image which comes indifferently from a video camera or a thermal camera. The right segment thus detected is represented in the source image by a line.

Our image analysis is carried out in several stages:

1. Histogram normalisation
2. Smooth Gaussian
3. Dilate/Erode
4. Smooth Laplacian
5. Threshold
6. Hough Lines Binary
7. Group Lines
8. Lines Filter

We used the OpenCV⁵ library to implement this functionality. From the source frame, (Figure 8a), histogram normalisation (1) (Figure 8b) is used to limit the undesirable effects caused by variations in brightness. This requires two parameters (minimum and maximum) so as to determine the proportionality rule between the input pixel value and the output pixel value. We obtain the best results for values between 150 and 180. The Gaussian convolution filter (2) is used to reduce noise and smooth the image (Figure 8c). It requires a single parameter, the core, whose value is between 7 and 9 to obtain the best results. We then apply a morphological closure operator (3) enabling noise to be reduced (Figure 8e). Erosion and dilatation are executed with the structuring element (core) by default from the OpenCV library with 6 to 8 iterations. We then apply a Laplacian convolution filter (4) in order to extract the initial contours whose iteration number is also between 6 and

⁵ Intel Open Source Computer Vision.

8 (Figure 8d). Finally, after having binarised (Figure 8f) the image (5), we use the Hough transform (6) to detect the image's lines. The next phase consists of statistically analysing lines detected by the previous method. We calculate the average and the standard deviation for the position of each end of the segments to group together close segments or those which are juxtaposed. Finally, we filter the segments detected according to their slopes in the image. This is the second way information from the inertial unit is used in our algorithm. We use the roll value to filter the segments detected. All the segments whose slopes in the image have a difference of over 5% of the unit's value are eliminated. Some results executed on others sources frames are available in Figure 9.

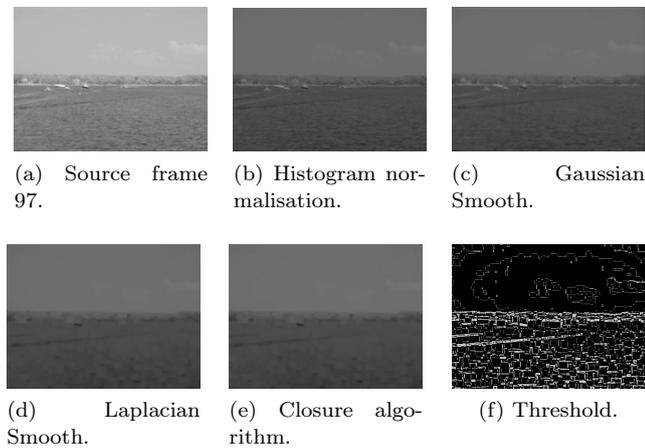


Fig. 8: Presentation of the different stages of our image analysis on a thermal image extracted from an image sequence.

4 Evaluating the Algorithm's Performance

We created this functionality in C# (managed language). Evaluations were carried out on a PC equipped with an Intel^(TM) Core 2 Duo processor at 2.66GHz. Tests were carried out on several image sequences in different situations and exposures. We present the results for three image sequences including a thermal image. We tested approximately 13062 images from sequences. Our detection algorithm correctly identifies the horizon is around 98.45% of cases. The graph in Figure 10 presents for each sequence of images the number of images tested (left) and the number of images for which

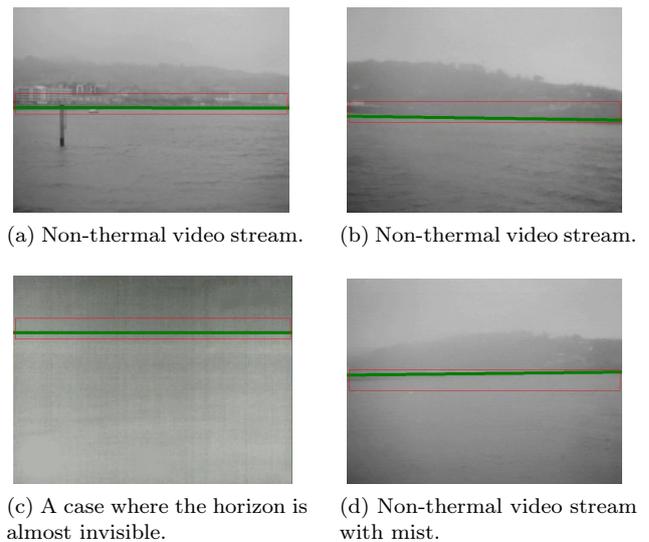


Fig. 9: Some results.

the horizon has been detected (right). The calculation time for the horizon extraction routine (excluding rendering) achieves an average of 6.5 milliseconds for a resolution of 640x480.

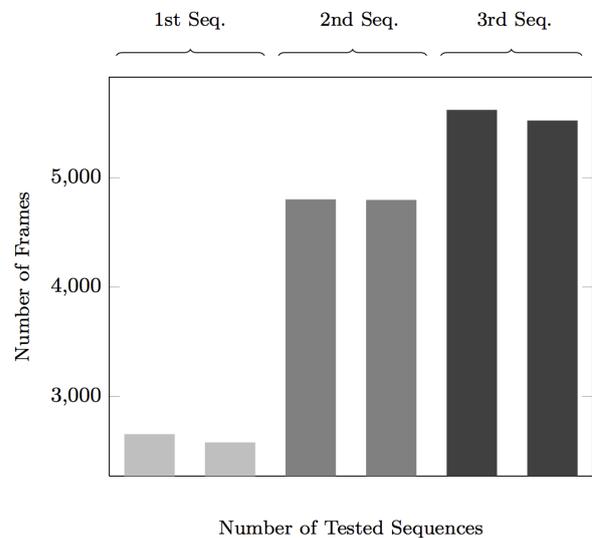


Fig. 10: Algorithm evaluation.

Concerning performances, we wanted to ensure that the different potentially usable resolutions would not deteriorate the video's visual quality. For the highest resolution tested, the algorithm enables approximately 32 fps to be obtained. Details of the impact of the im-

age's resolution on our algorithm are available in Figure 11.

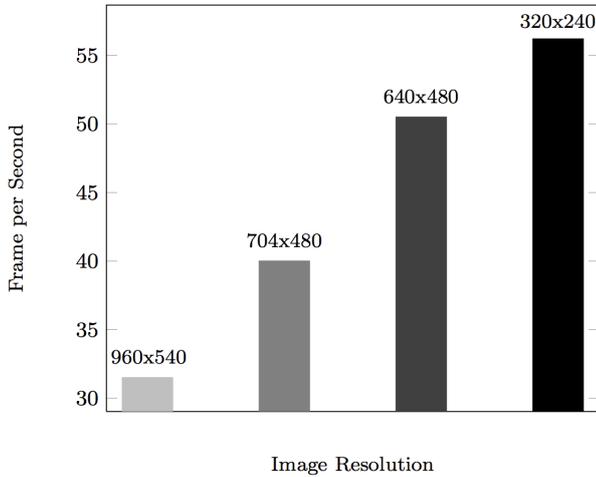


Fig. 11: FPS variation depending on frame resolution.

5 Merging video and charts

In addition to control the video camera by clicking on the e-chart as described here [4,24], the two objectives of this horizon detection presented in the previous sections are mixing video with digital geographical data embedded on the vessel and registering this video. We have onboard 3D mapping software. This represents a modelling of the environment in which the vessel evolves and whose modelled entities are for example the coast, seabed, beacons, etc. We therefore use the vessel's GPS position as well as the on-board inertial unit (roll, pitch, yaw) to georeference the camera's video stream. This video stream is thus positioned in the computer assisted navigation tool's virtual 3D world [4,8]. In Figure 12, we can see the video stream (black & white image) referenced in the 3D environment. The distance between the plane of projection and the vessel is arbitrary. This point of view is not provided for the user, but is illustrated here to understand the video's georeferencing. Only the "first person" point of view is given, ie. the 3D environment virtual camera is in the same position as the real camera on the vessel. This enables the point of view illustrated in Figure 14 to be obtained.

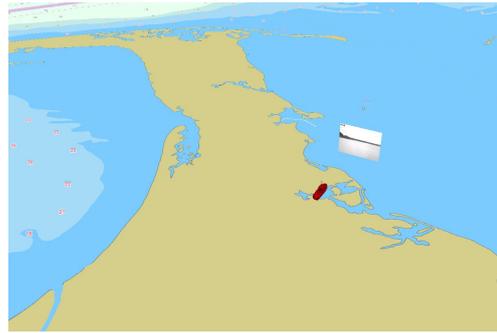


Fig. 12: Example of a georeferenced image in ECS 3D virtual world.

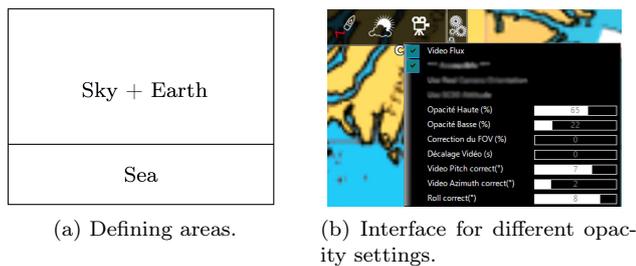
We divide the screen into two distinct areas as illustrated in Figure 13a. We use shader programming⁶ to vary the image's transparency differently according to each area. By doing this, the height of the Sky+Earth Area varies according to the horizon's position detected by the previously described algorithm.

Sky + Earth area: The video displayed represents 80% and digital data 20%. Charts must not be completely abandoned because in certain situations, for example close to the coast, mountains can be used as reference points for sailors. Since charts also model mountains, this information must not be removed. We use a function executed by the graphic chart defined by the DirectX (HLSL) library enabling linear interpolation of the transparency between this area and the lower area;

Sea area: The video displayed represents 20% of the display and digital data 80%. The video must not be completely removed because in certain situations, close to ports, for example, other vessels may be close by and visually "below" the visible horizon. The video must also not be made completely transparent due to a risk of making other surrounding vessels disappear.

The transparency of each area is adjusted by default with the previously presented values. However, users are still able to modify these values if they deem necessary by using linear cursors in the ECS graphic interface as illustrated in Figure 13b. An example of different cases is available in Figure 14.

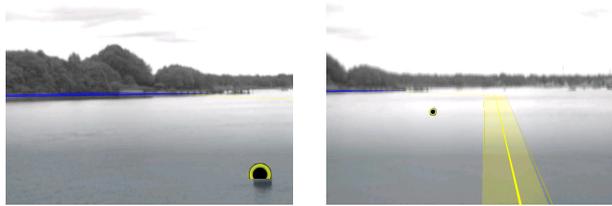
⁶ Instructions directly executed by the graphic card.



(a) Defining areas.

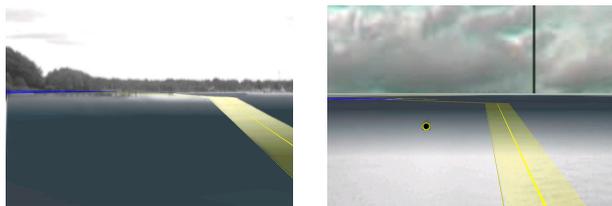
(b) Interface for different opacity settings.

Fig. 13: Defining the image's different areas.



(a) Thermal video in both areas. Visual augmentations by vectorial objects for the coast and a buoy.

(b) Thermal video and strictly virtual information (route to follow) without real representation.



(c) Thermal video in the upper area, e-chart color in the lower area and vectorial augmentations.

(d) Thermal video in the lower areas and modelled data in the upper areas and vectorial augmentations. (clouds are virtual !)

Fig. 14: Some examples of mixing the video stream, e-chart and vectorial augmentations.

6 Future Work

Our image analysis could benefit from several improvements. Robustness could still be improved by using the presence of two types of camera. Currently, detection is carried out in the video stream chosen by the user (classical or thermal), but we could use the thermal video stream to improve the detection of the horizon in the classic video stream in bad weather, for example. Another possible improvement would be automatically adapting the surface of the image analysis area. We could pre-process enabling the "height" of this area to be evaluated, which would enable a greater number of cases to be taken into account. We could also improve, if necessary, calculation loop execution time by using

GPU based programming given the parallel nature of much of the processing carried out.

7 Conclusion

We have presented an image analysis algorithm enabling the horizon to be detected in maritime scenes. The algorithm is able to detect the horizon by indifferently using a video stream from a classic camera or a thermal camera. We propose assessing our image analysis according to the quality of detection of the visible horizon and the processing execution speed for several image resolutions. Performance is satisfactory as we use values provided by the inertial unit to define the horizon's detection area in the image. Finally, we use the horizon detected by our image analysis to divide the display screen into two distinct areas. For each area, the transparency of the video image is defined showing, or not, information from the 3D environment for a maritime computer-assisted navigation tool (ECS).

References

1. Organisation Maritime Internationale (OMI), Proposed Modular Structure for RADAR Performance Standards. Design and Installation. CCRP and off-set compensation. (2007)
2. S. Grant, J. Goodyear, Online : www.mar.dfo-mpo.gc.ca (2010)
3. B. Pillich, G. Buttgenbach, in *Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE* (2001), pp. 1116–1119. DOI 10.1109/ITSC.2001.948818
4. O. Hugues, J.M. Cieutat, P. Guitton, in *12th Virtual Reality International Conference (VRIC)* (IEEE Computer Society France, 2010), ISBN: 2-9515730-9-X, pp. 149–154
5. G. Calvary, J. Coutaz, D. Thevenin, Q. Limbourg, L. Bouillon, J. V., *Interacting with Computers* **15**, 289 (2003)
6. M. Petit, C. Claramunt, C. Ray, G. Calvary, in *Proceedings of the 8th International Symposium on Web and Wireless Geographical Information Systems*, ed. by M. Bertolotto, X. Li, C. Ray (Springer Verlag, 2008), 5373, pp. 100–111
7. French Government. Oceans round table (grenelle de la mer). http://www.legrenelle-environnement.fr/IMG/pdf/Livre_bleu_anglais_web.pdf (2009). URL http://www.legrenelle-environnement.fr/IMG/pdf/Livre_bleu_anglais_web.pdf
8. W. Jie, H. Xian-Zhong, in *Systems, Man and Cybernetics, 2008. SMC 2008. IEEE International Conference on* (2008), pp. 1895–1899. DOI 10.1109/ICSMC.2008.4811566
9. Organisation Maritime Internationale (OMI). <http://www.imo.org/>
10. O.M. Phillips, *The dynamics of the upper ocean* (Cambridge monographs on mechanics and applied mathematics (Cambridge University Press), 1966)
11. M. Maldi, Hybrid tracking with occultation for augmented reality. (suivi hybride en présence d'occultations pour la réalité augmentée). Ph.D. thesis, Evry-Val d'Essonne University, Evry (2007)

12. H. Bouma, D.J. J. de Lange, S. P. van den Broek, In Proceedings of SPIE Electronic Imaging 2008 System Analysis for Digital Photography. **7114**, 1 (2008)
13. S. Todorovic, M.C. Nechyba, in *Florida Conference on Recent Advances in Robotics* (2004)
14. S. Ettinger, M. Nechyba, P. Ifju, M. Waszak, in *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, vol. 3 (2002), vol. 3, pp. 2134–2140. DOI 10.1109/IRDS.2002.1041582
15. H.Z. Yuan, X.Q. Zhang, Z.L. Feng, in *Image Analysis and Signal Processing (IASP), 2010 International Conference on* (2010), pp. 191–194. DOI 10.1109/IASP.2010.5476135
16. S. Fefilatyev, V. Smarodzinava, L.O. Hall, D.B. Goldgof, in *Machine Learning and Applications, 2006. ICMLA '06. 5th International Conference on* (2006), pp. 17–21. DOI 10.1109/ICMLA.2006.25
17. Y. Wang, Z. Liao, H. Guo, T. Liu, Y. Yang, in *Image and Signal Processing, 2009. CISP '09. 2nd International Congress on* (2009), pp. 1–5. DOI 10.1109/CISP.2009.5305740
18. B. Zafarifar, H. Weda, P.H.N. de With, *Visual Communications and Image Processing 2008* **6822**(1), 682220 (2008). DOI 10.1117/12.766689. URL <http://link.aip.org/link/?PSI/6822/682220/1>
19. T. Cornall, Technical Report MECSE-4-2004 (2004)
20. J.H. Woo, G.S. Kim, Conference on Machine Vision Application (MVA2005) pp. 526–529 (2005)
21. S. You, U. Neumann, IEEE Virtual Reality Conference pp. 71–78 (2001)
22. MTI XSens. Mti <http://www.xsens.com/en/general/mti>
23. F.A. Zabala, E-Book (2006)
24. O. Hugues, J.M. Cieutat, P. Guitton, in *AH '10: Proceedings of the 1st Augmented Human International Conference* (ACM, New York, NY, USA, 2010), pp. 1–6. DOI <http://doi.acm.org/10.1145/1785455.1785467>. URL <http://doi.acm.org/10.1145/1785455.1785467>