

Horizon Line Detection in Marine Images: Which Method to Choose?

Evgeny Gershikov, Tzvika Libe, and Samuel Kosolapov
 Department of Electrical Engineering
 Braude Academic College of Engineering
 Karmiel 21982, Israel

e-mail: eugenyl1@braude.ac.il, tzvika_libe1@walla.com, and ksamuel@braude.ac.il

Abstract— Five algorithms designed to find a horizontal line separating sea and sky in marine images in real-life conditions were implemented in this work and compared by their performance: accuracy and relative speed. One of the selected algorithms was based on regional covariances in luminance images, the second one was based on edge detection and the Hough transform, the third one used maximal local edge detection and the least-squares method, the fourth one employed median filtering in small neighborhoods and linear regression and the fifth one was based on regional edge magnitudes and the least-squares method. Real-life images were used for comparison. The most accurate line with respect to the angular error was obtained by using the edge detection and Hough transform based algorithm. The highest accuracy with respect to the position of the line was achieved by the regional covariance method. However, the highest speed was achieved by using the regional edge magnitude algorithm.

Keywords—horizon detection; marine images; edge detection; median filtering; image analysis; local edge magnitudes; regional edge magnitudes; regional covariances

I. INTRODUCTION

The horizon line is used for different purposes, such as navigation in airborne and marine vehicles and military surveillance. In an airborne vehicle, the horizon line can be used to determine, for example, its roll, pitch and yaw angles. In the case of military surveillance the horizon line is helpful in detecting the distance to targets.

Many horizon line detection methods are known today, for example, the methods in [1-8]. Some of these algorithms are based on edge detection [9], while others employ statistical methods [4]. Due to the variety of techniques, a comparison of the detection performance that they can achieve can be very helpful [1]. The goal of this research is to implement and compare the accuracy of a number of well-known as well as new or modified horizon-line detection approaches. Considering that in the later stages of this research the selected algorithm is to be implemented on a stand-alone hardware unit, algorithms complexity and their relative speed is also evaluated.

The structure of this paper is as follows. In the next section we present the algorithms discussed in this work for horizon line detection in marine images. Then, in Section III we describe possible improvements of the detection techniques. Section IV discusses the methods and criteria used in the comparison of the algorithms and Section V presents horizon detection results: quantitative and visual.

Finally, Section VI provides a summary of this work and our conclusions.

II. ALGORITHMS COMPARED

Five algorithms are compared in this work, as described below. The motivation for their choice is comparison of local feature based algorithms, such as those based on edges (for example, H-HC or H-LSC), with global feature based methods, such as H-COV-LUM or H-REM, that use regional covariances and regional edge magnitudes, respectively. The H-MED algorithm extends the meaning of a local feature (edge) at a pixel to its small neighborhood and then looks for the maximal edge in the vertical direction. Thus, it introduces a compromise between local and global features. No algorithms that require a training stage, such as neural networks and support vector machines were chosen for the comparison, but only simple low complexity methods were taken. Considering future DSP implementations training was found to be impractical.

The compared methods are:

“H-COV-LUM” – an algorithm that uses regional covariances, as introduced in [4], but modified to calculate these covariances using luminance images. Although there are cases where the color information is important [10], in this case, using achromatic image data only improves the algorithm speed significantly with minimal loss in accuracy. COV-LUM stands here for Covariance of Luminance.

“H-HC” – uses pre-processing, Canny edge detector [11] and Hough transform [12]. HC stands for Hough and Canny.

“H-LSC” – uses pre-processing, maximal local edge detection and calibration by the least-squares method approach. LSC stands here for Least Squares Calibration.

“H-MED” – searches for the maximal edge in the vertical direction based on an extended neighborhood of a pixel, followed by median filtration in order to reject outlying points and linear regression. MED stands here for median.

“H-REM” – divides the image into vertical stripes and searches for the maximal regional edge magnitude in each stripe, followed by the least-squares technique to estimate the best line passing through the maximal edge coordinates. REM stands here for Regional Edge Magnitudes.

The algorithms are described in more detail in the next subsections.

A. Regional covariance based algorithm (H-COV-LUM)

An algorithm for horizon detection for remotely piloted Micro Air Vehicles was introduced in [4]. The algorithm

receives an image taken from the air as input and searches for an optimal partition of the image into two regions: sky and ground (or in our work sky and sea) using a line, which is the detected horizon. The optimization criterion is based on the determinants and traces of the covariance matrices of the two regions. More specifically, if we denote a sky pixel by $\mathbf{x}_{i,j}^s = [R_{i,j}^s \ G_{i,j}^s \ B_{i,j}^s]^T$, where $R_{i,j}^s, G_{i,j}^s, B_{i,j}^s$ are the primary red, green and blue values at the pixel (i,j) , and we denote a ground pixel by $\mathbf{x}_{i,j}^g = [R_{i,j}^g \ G_{i,j}^g \ B_{i,j}^g]^T$, then the covariance matrices of the two regions are given by $\Lambda^s = E\left(\left(\mathbf{x}_{i,j}^s - \mu^s\right)\left(\mathbf{x}_{i,j}^s - \mu^s\right)^T\right)$, $\Lambda^g = E\left(\left(\mathbf{x}_{i,j}^g - \mu^g\right)\left(\mathbf{x}_{i,j}^g - \mu^g\right)^T\right)$, where $\mu^s = E\left(\mathbf{x}_{i,j}^s\right)$ and $\mu^g = E\left(\mathbf{x}_{i,j}^g\right)$. $E()$ denotes here statistical mean. The optimization criterion, considered for the possible horizon line orientations and positions and maximized is given by [4]:

$$J = \frac{1}{\det(\Lambda^s) + \det(\Lambda^g) + \text{trace}^2(\Lambda^s) + \text{trace}^2(\Lambda^g)}, \quad (1)$$

where $\det()$ denotes the determinant and $\text{trace}()$ denotes the trace of the covariance matrices Λ^s and Λ^g .

We consider a similar criterion to the one in (1) for the luminance image, thus the optimization term J becomes

$$J = \frac{1}{\text{var}(Y^s) + \text{var}(Y^g) + \text{var}^2(Y^s) + \text{var}^2(Y^g)}, \quad (2)$$

where $\text{var}()$ stands for variance and Y^s, Y^g are the luminance values of the sky and ground regions, respectively. A simplified optimization criteria

$$J = \frac{1}{\text{var}(Y^s) + \text{var}(Y^g)} \quad (3)$$

can be used instead of the one in (2) with similar results. Thus, we search for the line maximizing (3) among all considered horizon line orientations and positions. Also, defining a region of interest (ROI) in the image and searching the horizon line only in this area speeds up the algorithm significantly. Alternatively, the input image can be down-sampled prior to the application of the algorithm to reduce its runtime, but this will decrease the accuracy as well.

B. Edge detection and Hough transform based algorithm (H-HC)

The stages of this method can be summarized as follows.

1. Pre-process the image using morphological erosion to reduce the probability of the detection of weak edges in the later stages. A small circular structuring element can be used here. Alternatively, the image can be smoothed using a low pass filter, but we found morphological operations to provide better performance in terms of preserving the edges [13].
2. Apply Canny [11] edge detector to the pre-processed image.

3. Apply the Hough transform [12] to the edges map.
4. Choose the horizon line to be the longest line found in the previous step.

C. Edge detection and least-squares calibration based algorithm (H-LSC)

This algorithm is based on edge detection as well, but uses a simple algorithm to detect the maximal local edge in the vertical direction in each column of the image. Its stages are described below.

1. Pre-process the image using morphological erosion.
2. Find the maximal vertical local edge in each column of the image. The simplest way to measure the local edge magnitude is using an approximation of the vertical derivative, e.g., $|Y_{i+1,j} - Y_{i,j}|$. Store the (i,j) coordinates of the maximal edges.
3. Use the least-squares method to find the optimal line passing through the maximal edges' coordinates. Edges with very small values as well as very big ones can be discarded here since they are most likely caused by noise. This increases the algorithm robustness in the presence of varying lighting effects.
4. An optional step of median filtering can be added to remove outliers. This step can be applied to the vertical coordinates of the maximal edges (prior to Step 3) or to the regression errors (following Step 3). We define the regression error as the error at coordinates (i,j) of a maximal edge, i.e.,

$$\text{err}_{i,j} = |i - a_1j - a_0|, \quad (5)$$

where a_0, a_1 are the optimal line coefficients found in Step 3. We define the median filtered error $\text{err}_{i,j}^{\text{med}}$ as $\text{err}_{i,j}$ after applying a median filter. Now the outliers are (i,j) , where $|\text{err}_{i,j} - \text{err}_{i,j}^{\text{med}}| > Th$, and can be removed. Th here is the threshold (e.g., a value of 1).

D. Median filtering and linear regression based algorithm (H-MED)

This algorithm employs median filters in several stages providing high performance in the presence of noise. The stages of the algorithm are:

1. Pre-process the image using morphological erosion.
2. Find the maximal vertical edge in each column of the image using the extended neighborhood of each pixel. Here the edge at pixel (i,j) is measured as the absolute difference between two median values of the 5 pixels above and including pixel (i,j) and the 5 pixels below it, i.e., $\text{edge}_{i,j} = |\text{med}_{i,j}^1 - \text{med}_{i,j}^2|$, where

$$\begin{aligned} \text{med}_{i,j}^1 &= \text{median}\{Y_{k,j}\}_{k=i-4}^i, \\ \text{med}_{i,j}^2 &= \text{median}\{Y_{k,j}\}_{k=i+1}^{i+5}. \end{aligned} \quad (6)$$

The (i,j) coordinates of the maximal edges are stored.

3. Use linear regression to find the optimal line passing through the maximal edge coordinates.
4. An optional step of median filtering can be added to remove outliers. This step can be the same as Step 3 in H-LSC.

E. Regional edge magnitudes and least-squares calibration based algorithm (H-REM)

This algorithm extends the idea of local edge magnitudes used in the H-HC and H-LSC methods, for example, beyond the concept of extended neighborhoods used by H-MED to regional edge magnitudes. The steps of this method are:

1. Pre-process the image using morphological erosion.
2. Divide the image into vertical stripes each consisting of a number of columns. Run on each stripe separately and calculate the edge magnitude at each pixel in it in the vertical direction using a simple, but robust equation for the edge:

$$edge_{i,j} = \frac{1}{L} \left| \sum_{k=i-L}^{i-1} Y_{k,j} - \sum_{k=i+1}^{i+L} Y_{k,j} \right|, \quad (7)$$
 where L is the number of pixels used in the calculation above and below the current pixel.
3. Sum the edge magnitudes of each row in the current stripe to get regional edge magnitudes.
4. Find the row of the maximal regional edge in each stripe. Associate it with the column number corresponding to the center of the stripe to get the maximal edge coordinates.
5. Use the least-squares method to find the optimal line passing through the edge coordinates. Edges with very small values as well as very big ones can be discarded here as in the H-LSC algorithm. This is optional since the algorithm employs a greater neighborhood for edge magnitude calculation making it more robust.
6. An optional step of median filtering can be added as in the H-LSC algorithm.

Next we discuss several possible improvements of the algorithms.

III. PROPOSED IMPROVEMENTS TO THE ALGORITHMS

Most of the proposed algorithms rely on a certain measurement of edge magnitudes (H-COV-LUM is an exception). To make these measurements more reliable we propose the following ideas.

A. Filtering out very dark or very bright pixels

The idea is to filter out the dark areas of the image as well as areas with sun light effects by calculating the image color energy everywhere and discarding the pixels where this energy is too high or too low [14]. A simple measure of the image color energy at pixel (i,j) is

$$energy_{i,j} = R_{i,j}^2 + G_{i,j}^2 + B_{i,j}^2 \quad (8)$$

or, alternatively,

$$energy_{i,j} = R_{i,j} + G_{i,j} + B_{i,j}. \quad (9)$$

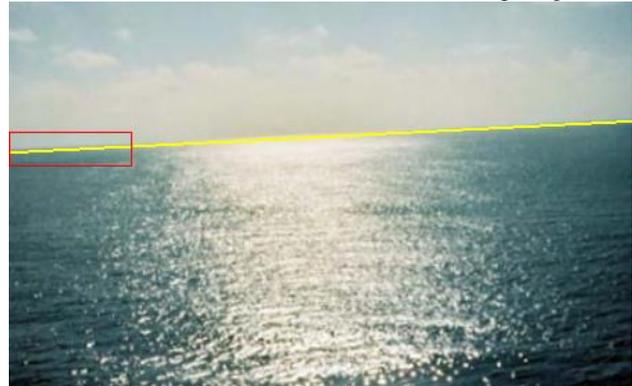
A pixel is discarded if $energy_{i,j} < Th_{Min}$ (dark pixels) or $energy_{i,j} > Th_{Max}$ (pixels saturated with light) and it then cannot be chosen as the pixel with maximal edge magnitude in relevant algorithms (such as H-LSC). Note that Th_{Min} and Th_{Max} are the two thresholds determining which pixels are to be discarded: the low one and the high one, respectively. The effect of filtering out of pixels from the image is shown in Fig. 1 for the rotated Horizon_3 image. Note that the horizon appears jagged due to the marking of the line in the image itself subject to pixel resolution.

B. Adding weights to the neighboring pixels used in edge magnitude calculation

When an extended neighborhood is used for the edge magnitude calculation as, for example, in the H-REM method, we propose giving a decreasing weight to the pixels in the neighborhood based on their distance to the current pixel: a closer pixel will get a higher weight. Thus, the edge calculation of Equation (7), for example, can be replaced by

$$edge_{i,j} = \frac{1}{L} \left| \sum_{k=i-L}^{i-1} \alpha^{|i-k|} Y_{k,j} - \sum_{k=i+1}^{i+L} \alpha^{|i-k|} Y_{k,j} \right|, \quad (10)$$

H-REM result for rotated Horizon_3 w.o. filtering out pixels



H-REM result for rotated Horizon_3 with filtering out pixels

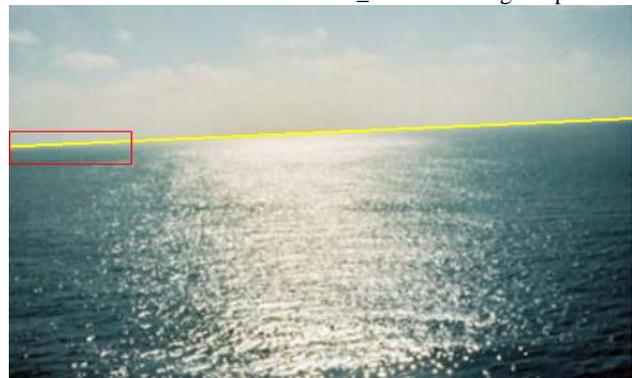


Figure 1. H-REM detection results for the rotated Horizon_3 image with and without (w.o.) filtering out of pixels. The horizon line is marked (in yellow). Note the error introduced in the top figure, especially in the area marked with the rectangular frame.

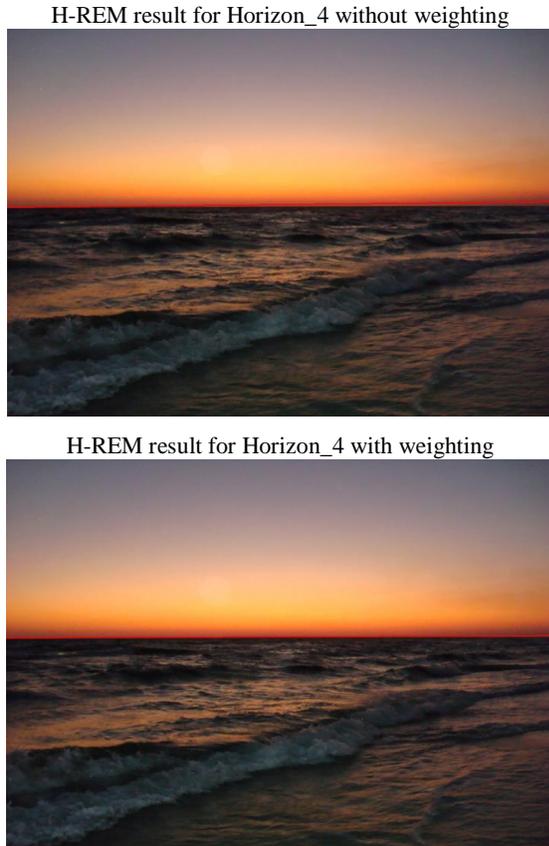


Figure 2. H-REM detection results for the Horizon_4 image with and without weighting. The horizon line is marked (in red). Note the error introduced in the top figure on the right side of the line.



Figure 3. H-REM detection results for the Horizon_3 image with and without histogram equalization (hist. eq.). The horizon line is marked (in yellow). Note the error introduced in the top figure.

where $0 < \alpha < 1$ is a real number. The expression given in (10) results in better edge magnitude estimation and better algorithm performance compared to using equal weights for all pixels in the neighborhood. This is shown in Fig. 2 for the Horizon_4 image. Note the error in the detected horizon line position when no weighting is used, i.e., the edge magnitude is calculated based on Equation (7).

C. Introducing safety intervals in edge magnitude calculation

Sometimes it makes sense to calculate the edge at pixel (i, j) using the pixels not directly above and below it, but starting from a certain distance (e.g., 1, 2 or 5 pixels) away. The reason may be blurring of the edges, which occurred, for example, due to errors introduced by compression of the processed image, especially when a simple digital camera is used.

Using this idea, Equation (7) of H-REM will turn into

$$edge_{i,j} = \frac{1}{L} \left| \sum_{k=i-D-L}^{i-D} Y_{k,j} - \sum_{k=i+D+1}^{i+D+L} Y_{k,j} \right|. \quad (11)$$

Here D is a positive integer parameter denoting the safety interval, i.e., the distance of the closest pixel used in the calculation of the current pixel (i, j) .

D. Histogram operations prior to executing the algorithms

A histogram operation prior to horizon detection may improve the detection performance if it increases the difference between sea and sky pixels and/or improves the similarity between pixels in the same region (sea or sky). Then the performance of algorithms based on regional covariances, such as H-COV-LUM, as well as local or regional edge magnitudes, such as H-LSC or H-REM, is expected to improve.

In this work we examine the use of histogram equalization of the luminance component of the image. This operation attempts to make the luminance levels more uniformly distributed while at the same time producing an image with a smaller total number of luminance levels. The result of applying histogram equalization (yielding 64 levels of luminance) to the Horizon_3 image and using the H-REM algorithm is shown in Fig. 3. Note the better precision of the detection in the bottom part of the figure.

All the proposed ideas are generally “safe to use”, meaning the algorithms’ performance is either improved or is not affected. For the images presented in Figs. 1-3 a visual improvement in the accuracy can be observed.

IV. COMPARISON METHODS AND CRITERIA

Next, we describe the images and criteria used for the comparison of the algorithms in this work.

A. Images used to compare the algorithms

The results of horizon detection for a group of 9 real-life marine images are presented in this work. Image input format is true color (24 bit per pixel) non-compressed BMP. Resolutions used vary from 249x169 to 900x675 pixels. Most images contain a horizon line separating the sea and the sky, clearly distinguished by the human eye. However, sometimes the horizon line is slightly distorted by camera optics and sea waves or concealed by marine vessels. It is clear that camera lens distortion may affect the accuracy of the horizon detection, however, for the methods and the images considered here this effect was found to be insignificant. To further challenge the selected algorithms, several images contained clouds or sun light effects near the surface of the sea water.

B. Comparison criteria

The algorithms were compared with respect to accuracy and speed. The accuracy was measured for the detected horizon angle relative to a horizontal line (in degrees) as well as the position of the line relative to the bottom left corner of the image (in pixels, sub-pixel resolution was not considered). The errors provided in the next section for these two horizon line parameters are measured relative to the line height and angle as determined visually. This means that the horizon line, as determined by the eye, was marked in the image manually and was considered the ground truth. Then the absolute difference of the determined line position and the one found visually was calculated and averaged on the group of test images. The same procedure was done for the determined line orientation relative to the one considered as ground truth. In addition, the algorithms' speed was measured in terms of run time (in seconds).

V. RESULTS

The accuracy comparison for the algorithms described above (height and angle deviations) is given in Table 1 in terms of the mean errors for the 9 test images. In some of these images the horizon is not horizontally aligned (e.g., see Fig. 4). As it can be seen, the angular deviation is very small on average for the H-HC, H-REM and H-LSC algorithms. We can speculate that the accuracy of H-HC results from the accuracy of the edge detection method by Canny [11] and of the employed Hough transform [12]. The height deviation is smallest for the H-COV-LUM method, based on regional covariances in luminance images. However, the fastest algorithm is H-REM, based on maximal edges and least-squares optimization, as seen from the run time comparison in Table 2. The H-LSC and H-HC methods are slower than H-REM, but the H-COV-LUM and H-MED algorithms are much slower than all of these three methods due to the required computations of regional

covariances or local medians in the process of the horizon detection. The run-time was measured in a MATLAB environment.

A. Visual results

Visual results are provided in Figs. 4-7. As it can be seen, all the algorithms provide good results for the Horizon_1 image (Fig. 4), although the H-COV-LUM method slightly misses the horizon line. This is due to the effect of the clouds and the sunlight reflection in the sea water. A similar effect can be seen for the Horizon_5 image (Fig. 5), where the H-COV-LUM method provides a slightly less accurate estimate of the horizon line than the other algorithms. H-REM is influenced here by the strong regional edges in the area of the waves, but produces a line with a slight deviation from the horizon. The other three methods achieve visually similar performance with very good detection of the horizon.

Figs. 4 and 5 show that H-COV-LUM (denoted H-COV in the figures) is an efficient algorithm for locating the position of the center of the horizon line, but sometimes the line is slightly rotated compared to the optimal one introducing an angular error. The algorithm copes well with images where the sky and the sea are uniform in appearance even when marine vessels are present, but it may be confused by clouds, sun reflection effects (Fig. 4) and strong waves (Fig. 5). The solution to this may be a pre-processing stage which removes some of the clouds, light reflection effects and waves from the image resulting in more uniform sky and sea areas. This is currently under research.

In Fig. 6, all of H-COV-LUM, H-HC and H-REM methods provide good estimates of the horizon line, while H-LSC is less accurate. As for H-MED, its performance is inferior to the others due to a bigger angular error. This method is more affected by the closer ship concealing the horizon line. The reason for the performance decrease for H-LSC and H-MED is that both detect maximal edges at pixels of the larger marine vessel instead of the horizon that is partly hidden. Then the least-squares technique or the linear regression employed to find the optimal line passing through the maximal edge locations produce a line that is shifted downwards relative to the optimal horizon line. The solution to this problem can be calculating the optimal line many times using partial data and then choosing the one passing through or close to the maximal number of edge pixels. This idea is the subject of future research.

The H-HC algorithm, on the other hand, is robust to the hindrances introduced by the sea vessels in the image of Fig. 6 due to the use of the Hough transform that detects the line passing through the maximal number of pixels in the edge map. Thus, even though some of the ship pixels are detected as edges, this does not confuse the method as long as more pixels are marked as edges on the real horizon line.

In Fig. 7 the detection results for one more image are shown. Despite the fact that the sky in the image is very cloudy, good detection results can be observed for H-MED,

H-HC and H-LSC algorithms while in H-REM a small angular error is introduced. The worst performance here is that of H-COV since it is confused by the presence of the clouds resulting in a line located a noticeable distance above the horizon.

TABLE 1. MEAN HEIGHT DEVIATION (PIXELS) AND ANGLE DEVIATION (DEGREES) FOR THE FIVE ALGORITHMS

Algorithm	Mean height deviation	Mean angle deviation
H-LSC	1.92	0.23 °
H-COV-LUM	1.11	0.47 °
H-HC	1.67	0.13 °
H-MED	1.83	0.44 °
H-REM	2.28	0.19 °

TABLE 2. MEAN RUN TIMES (SECONDS) FOR THE FIVE HORIZON DETECTION ALGORITHMS

Algorithm	Mean Time (sec.)
H-LSC	0.33
H-COV-LUM	2.24
H-HC	0.45
H-MED	1.46
H-REM	0.14

VI. CONCLUSIONS

Five different algorithms for horizon detection in marine images were examined in this work. The techniques employed by these algorithms vary from using regional covariances of sky and sea regions (H-COV-LUM) to using edge detection and Hough transform (H-HC), using maximal edge detection and the least-squares method (H-LSC), using median filtering and linear regression (H-MED) and using regional edge magnitudes and the least-squares method (H-REM). The algorithms were implemented and compared for a group of test images with respect to accuracy as well as run time or speed. The most accurate method with respect to the angular error was found to be H-HC, while the other algorithms do not lag far behind. The H-COV-LUM algorithm provided the highest accuracy when estimating the height of the horizon line above the bottom left corner of the image. Also when comparing the algorithms' speed, the fastest method was H-REM. We conclude that all the algorithms examined in this work can be used for horizon detection in still marine images. They successfully deal with the biggest challenges of horizon line detection, such as varying illumination effects as well as the presence of the sun, waves, ships and clouds in the image. In addition to that, the algorithms can be used in images taken by infrared cameras, an idea that is currently being researched.

ACKNOWLEDGMENT

We would like to thank the administration of Ort Braude Academic College of Engineering and the Department of Electrical Engineering for providing the opportunity and financial means to conduct this research.

REFERENCES

- [1] T. Libe, E. Gershikov, and S. Kosolapov, "Comparison of methods for horizon line detection in sea images", Proc. CONTENT 2012, Nice, France, 2012, pp 79-85.
- [2] G. Bao, Z. Zhou, S. Xiong, X. Lin, and X. Ye, "Towards micro air vehicle flight autonomy research on the method of horizon extraction", Proc. IEEE Conf. on Instrumentation and Measurement Technology, 2003, pp. 1387-1390.
- [3] G.-Q. Bao, S.-S. Xiong, and Z.-Y. Zhou, "Vision-based horizon extraction for micro air vehicle flight control", IEEE Trans. on Instrumentation and Measurement, vol. 54, 2005, pp. 1067-1072.
- [4] S. M. Ettinger, M. C. Nechyba, P. G. Ifju, and M. Waszak, "Vision-guided flight stability and control for micro air vehicles", Proc. IEEE Conf. on Intelligent Robots and Systems, Lausanne, Switzerland, 2002, pp. 2134 – 2140.
- [5] S. Fefilatyeve, V. Smarodzinava, L.O. Hall, and D.B. Goldgof, "Horizon detection using machine learning techniques". Proc. International Conference on Machine Learning and Applications, 2006, pp. 17-21.
- [6] S. Fefilatyeve, D.B. Goldgof, and L. Langebrake. "Towards detection of marine vehicles on horizon from buoy camera". Proc. SPIE, 2007, pp. 6736:673600.
- [7] K. Nonami, F. Kendoul, S. Suzuki, W. Wang, and D. Nakazawa, Autonomous flying robots, Tokyo, Dordrecht, Heidelberg, London, New York: Springer, 2010.
- [8] Y. Wang, Z. Liao, H. Guo, T. Liu, and Y. Yang, "An approach for horizon extraction in ocean observation", Proc. IEEE Congress on Image and Signal Processing, 2009, Tianjin, China, pp. 1-5.
- [9] S. Kosolapov, "Robust algorithms sequence for structured light 3D scanner adapted for human foot 3D imaging", Journal of Comm. and Computer, vol. 8, 2011, pp. 595-598.
- [10] E. Gershikov and M. Porat, "On color transforms and bit allocation for optimal subband image compression", Signal Proc.: Image Communication, vol. 22, Jan. 2007, pp. 1-18.
- [11] J. Canny, "A computational approach to edge detection", IEEE Transactions on PAMI, vol. 8, 1986, pp. 679-697.
- [12] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures", Comm. ACM, vol. 15, 1972, pp. 11–15.
- [13] D. Dusha, W. Boles, and R. Walker, "Fixed-Wing Attitude Estimation Using Computer Vision Based Horizon Detection", Proc. AIAC, 2007, Melbourne, Australia, pp. 1-19.
- [14] S. Kosolapov, "Evaluation of robust algorithms sequence designed to eliminate outliers from cloud of 3D points", Proc. SMTDA, 2010, Chania, Crete, pp. 383-389.

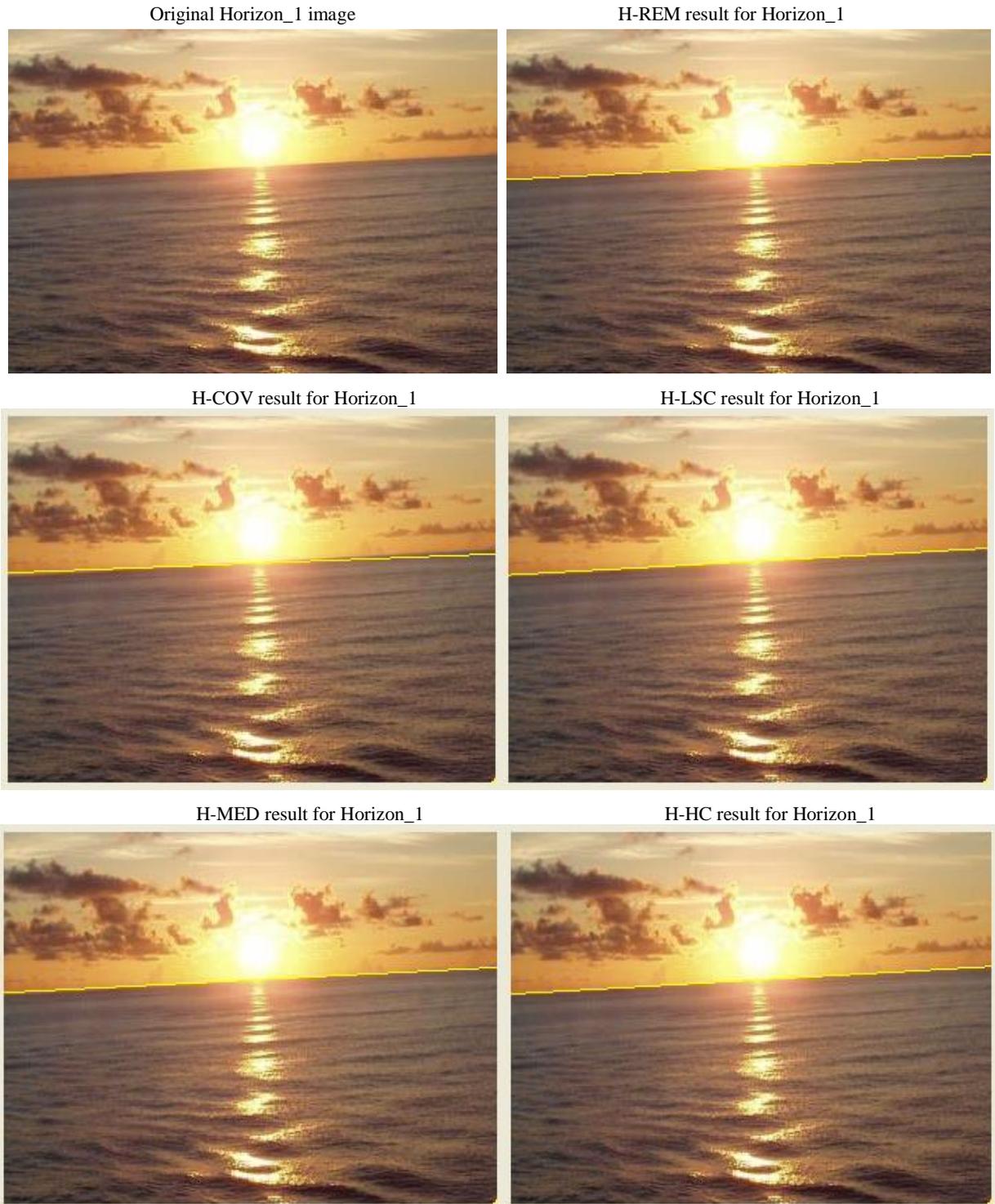


Figure 4. Horizon detection results for image Horizon_1. The (yellow) line marks the detected horizon. Note the clouds and reflected light effects in this image.
H-COV stands here for H-COV-LUM.

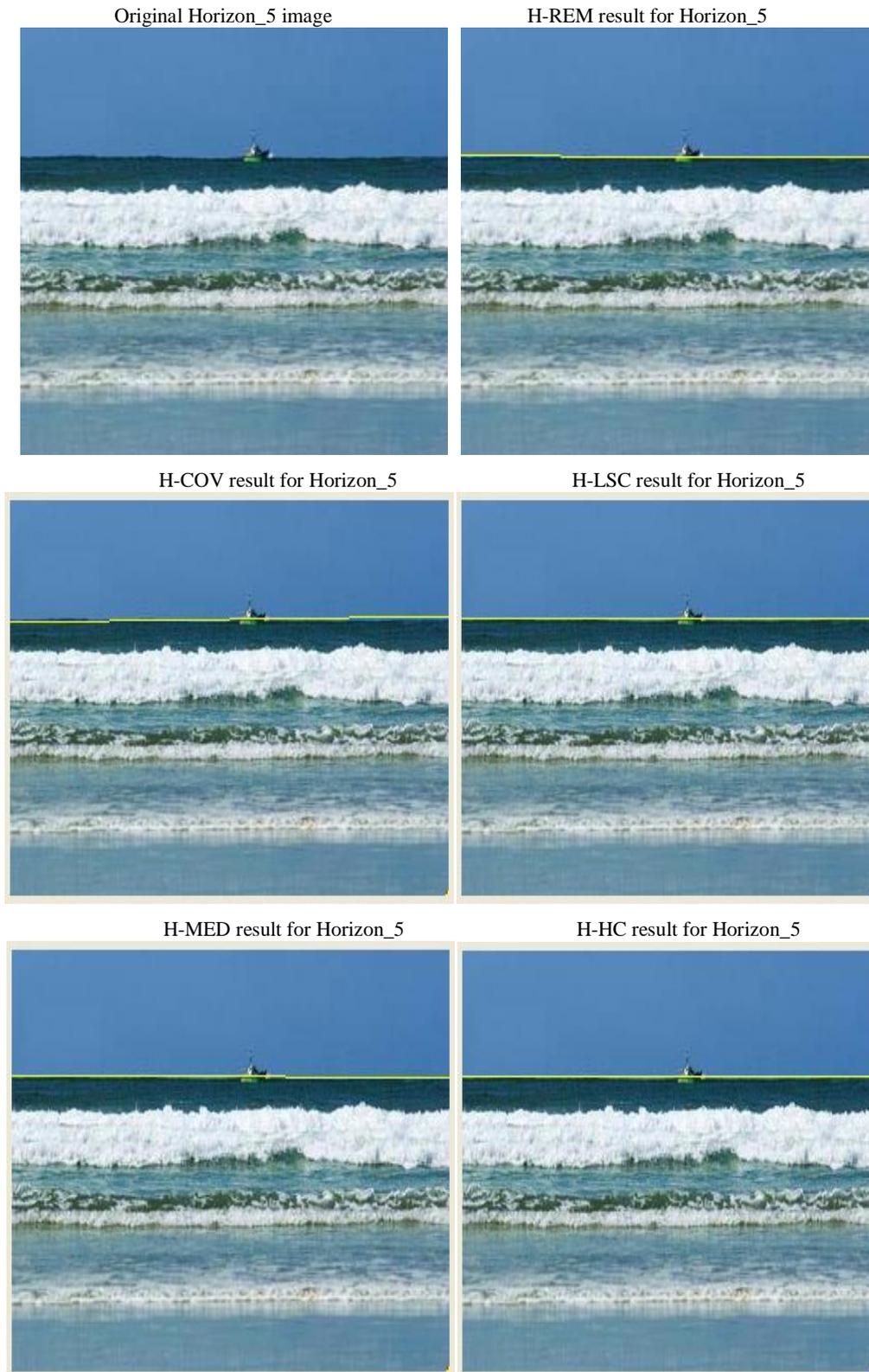


Figure 5. Horizon detection results for image Horizon_5. The (yellow) line marks the detected horizon. Despite the waves and the ship present, all the algorithms detect the horizon correctly.

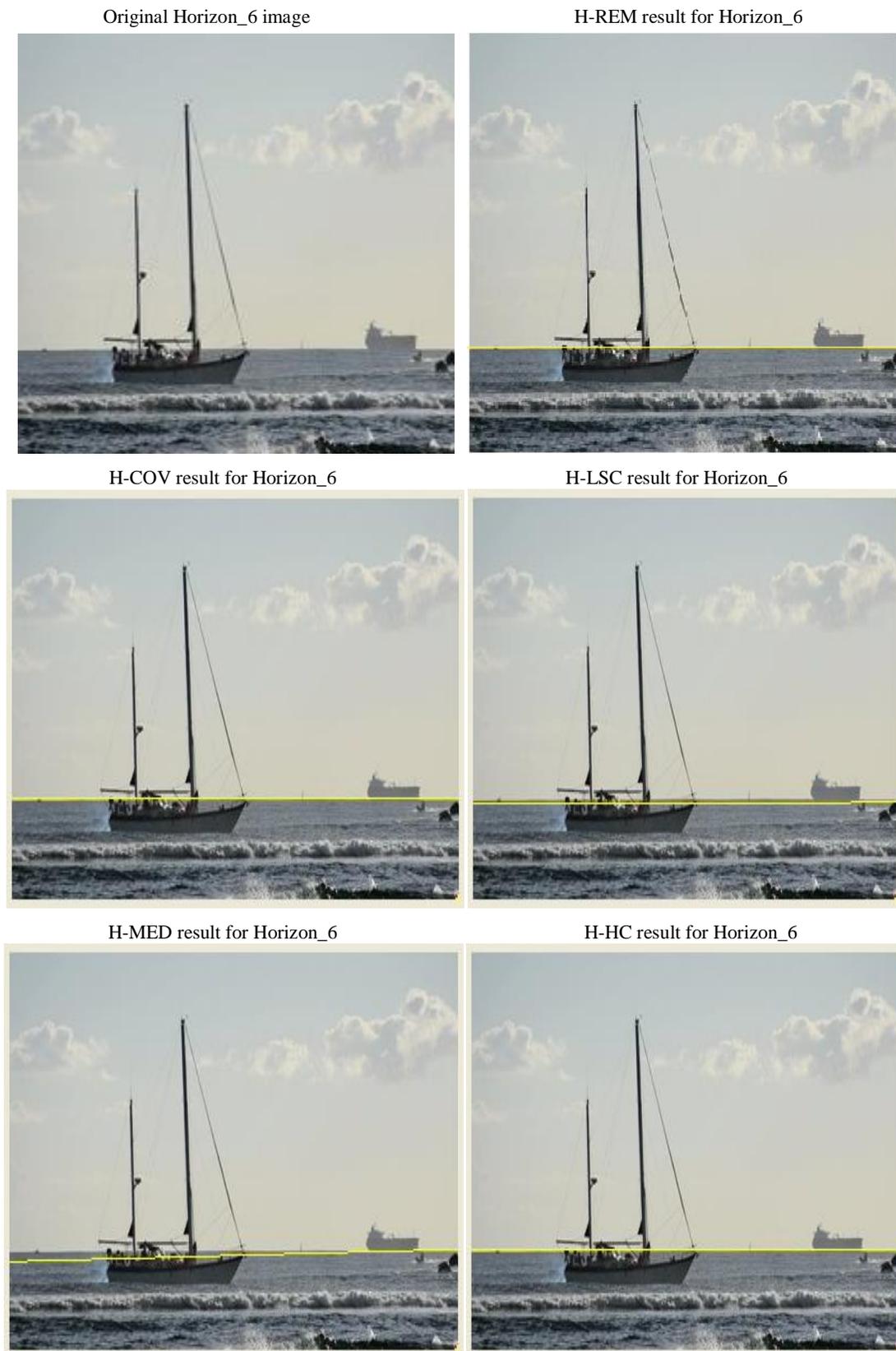


Figure 6. Horizon detection results for image Horizon_6. The (yellow) line marks the detected horizon. Note the waves and the sea vessels present in this image, especially the closer one blocking the horizon.



Figure 7. Horizon detection results for image Horizon_7. The (yellow) line marks the detected horizon. Note that a significant area of the image is covered by clouds introducing a challenge for correct horizon detection.