

Comparisons of crest line extraction algorithms applied to marine dunes

Julien Ogor and Benoit Zerr

Lab-STICC Lab (UMR CNRS 6285), France and

ENSTA Bretagne - 2 rue François Verny, 29806 Brest Cedex 9, France.

Email: julien.ogor@ensta-bretagne.fr; benoit.zerr@ensta-bretagne.fr

Abstract—Marine dunes are bedforms that, because of their locations and dimensions, can interfere with human activities at sea. New MultiBeam Echo-sounder Systems (MBES) enable to collect always more accurate and denser data revealing the diversity of shape, size and dynamics of these features with finer details. Until now, research on dunes have mostly consisted in describing or modeling their morphology and migratory behavior. The development of new automatic tools allowing to quantitatively describe their morphology and dynamics seems necessary. The dune crest extraction is the first step towards the automation of marine dune analysis. Crest detection algorithms relying on differential geometry concepts were designed for discrete surfaces (meshes). In this paper, we endeavor to quantitatively compare the results of four of these algorithms applied on marine dunes and conclude on their applicability for dune crest detection.

I. INTRODUCTION

The advent of MultiBeam Echo-sounder System (MBES) enabled scientists to collect larger amount of data over extensive areas and faster. With this new technology, new bedforms have been discovered or at least their geometry is better known. This is especially the case for marine dunes. Singlebeam Echo-sounder systems only provided information over transects. Now, the dune geometry is described at a high resolution and more accurately than before. Consequently, one has more insight into the richness of dunes in terms of size, shape, dynamics and migration.

These particular bedforms are getting more and more extensively studied for several reasons. First, they can disturb human activities at sea. For instance, marine renewable energies are a promising source of energy. But, prior to the installation of infrastructures at sea, one must make sure that the installation site is not in the middle of a dunefield or at reach of a migratory dunefield. Indeed, dunes can be several tens of meters high and could then damage the current or wind turbines. Furthermore, the migration of these bedforms means changes in the bathymetry and, so, risks for the navigation safety. Nearshore sand banks and dunes are also very accessible aggregate sources. This activity needs to be monitored since these banks are biology hotspots or natural coastal defences important when it comes to coastal risk management.

Dunes have been studied from two different, yet, complementary perspectives. Some researchers try to

understand the mechanisms under the dune formation and migration by designing models [1]–[3]. Others work on in situ measurements (bathymetry, currents, granulometry, etc.). They describe freshly-discovered dunefield, analyze the changes of more intensively studied dunefields [4]–[7] or try to quantify the dune migration rates [8], [9]. This approach is very qualitative. Dune characteristics (length, width, height, etc.) are manually estimated which is a time-consuming task. These two approaches share the same goal even though they appear different. The models enable to understand the role of waves, wind or tidal currents on the sediment transport and ultimately on dunes when the second analyzes the dune geometry. The hydrodynamics and the bathymetry influence one another and this is the reason why these two approaches are complementary.

Dune crests give much information about the dune nature, dimensions or dynamics. Algorithms based on dune crests have already been designed for estimating their migration. But the crest lines are manually or at best semi-automatically drawn. Knowing their locations would also help in improving the models since these characteristic lines disturb the flow and sediment transport. Thus, to improve our knowledge on dune crests is crucial for all the afore-mentioned activities related to dunes.

To design an algorithm able to automatically detect the dune crests would be helpful to save time, to objectively detect the crests and also to be able to exploit the available bathymetric databases.

Algorithms have been designed to extract characteristic lines on surfaces [10]–[13]. Crest lines are mathematically well-defined in differential geometry. These algorithms work with discrete surfaces called triangular meshes. They interpret the mathematical definition for it to be applicable onto surfaces with a discrete representation. The basics of differential geometry can be found in [14].

These algorithms are generally used for visualization and graphics purposes [15]. Research on characteristic lines have mainly been conducted for applications in human perception, image and data analysis, medical image analysis, face recognition or non-photorealistic rendering. The algorithms

performances are often compared based on a purely visual analysis. In this context of this study, the results need to be as realistic as possible since the crests are then to be used for deriving information on the dune geometry (length, height, asymmetry, etc.) and dynamics (deformation or migration). Thus, it seems relevant to quantitatively compare the obtained crests and analyze whether the detected crests match the dune crests.

In this paper, we aim at designing a comparison framework to quantify the closeness of four detection algorithms [16]–[18] and discuss their applicability for dune crest detection. In section II, differential geometry concepts are presented and the functioning of the four algorithms is explained. The next section describes the bathymetric data accessible to the authors and the regions that were selected for conducting the comparisons. Section IV shows the crest extraction results as well as the quantitative comparisons. The applicability of these algorithms to marine dunes is also discussed in this section. In the final section, conclusions on this work are drawn and perspectives are presented.

II. CREST DETECTION AND DIFFERENTIAL GEOMETRY

A. Differential geometry: basic notions

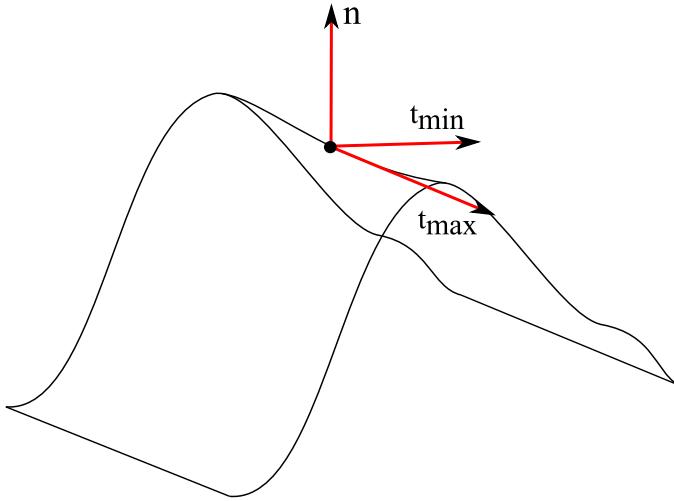


Fig. 1. Local basis at a crest

For the reader to fully understand the functioning of the algorithms, it is necessary to introduce notions of differential geometry.

First, the curvature is a local measure of the slope change in a chosen direction. When this direction is tangent to the surface, the curvature is called the normal curvature in this direction at the tangent point.

At each point, two special normal curvatures can be estimated; they are called the principal curvatures and correspond to the minimum and maximum of the normal curvatures. These principal curvatures are classically noted k_{\min} and k_{\max} .

The associated directions are named the principal directions (t_{\min} and t_{\max}) and form an orthonormal basis with the surface normal (see Figure 1).

At a crest, the basis is oriented as shown on Figure 1. There, the minimal curvature is minimum. Henceforth, the crest lines are defined as the loci of the surface where k_{\min} is locally minimal in t_{\min} direction. Mathematically the crest points are defined as follows:

$$e_{\min} = \nabla k_{\min} \cdot t_{\min} = 0$$

$$\text{and } P_{\min} = \nabla e_{\min} \cdot t_{\min} > 0$$

e_{\min} is called minimal extremality.

Crest detection algorithms rely on this common definition, but check on these conditions differently. A triangular mesh is a 3D discrete representation of a surface. The surface is approximated by triangular faces. The triangle summits are called vertices. Two vertices are connected by an edge. The four selected algorithms enable detecting crests with a great flexibility in their locations. Indeed, the resulting crests are allowed to cross the triangular faces. It makes these algorithms less dependent than others on the mesh resolution and shape. The other methods output lines running through the mesh vertices [12] or even simply detect the faces crossed by crest lines [13]. In other words, the obtained crests are highly-dependent on the mesh structure.

B. Hildebrandt's algorithm

Hildebrandt's algorithm only requires to know k_{\min} and t_{\min} at each vertex of the mesh. To evaluate e_{\min} value at a vertex V , e_{\min} is calculated for each face containing V . e_{\min} is supposed constant on a face. An average of the e_{\min} values on the face containing vertex V gives an estimate of e_{\min} at V . On edges where there is a e_{\min} sign change, a possible crest segment is drawn. It corresponds to the segment of zero level of e_{\min} . Finally, the following conditions are tested on face T to validate the previously detected segment:

$$|\sum_{v_i \in T} k_{\min}(v_i)| > |\sum_{v_i \in T} k_{\max}(v_i)| \text{ and}$$

$$\nabla e_{\min} \cdot \sum_{v_i \in T} t_{\min}(v_i) > 0$$

C. Ohtake's algorithm

Unlike Hildebrandt's algorithm, Ohtake's requires to input e_{\min} values. This time the tests below are made on each edge $[v_1 v_2]$:

$$k_{\min}(v_i) < -|k_{\max}(v_i)| \text{ with } i=1,2 \text{ and } e_{\min}(v_1)e_{\min}(v_2) < 0$$

$$e_{\min}(v_i) * [(v_{3-i} - v_i) \cdot t_{\min}] < 0 \text{ with } i=1,2$$

If the edge satisfies the above conditions, a crest point is defined where e_{\min} is equal to zero. e_{\min} is assessed to change linearly along the edge so that the crest point location can be estimated. When two edges of a face are crossed by a crest, a segment is drawn between the corresponding crest points. The idea behind the last condition is to check whether the curvature value decreases when moving towards the crest point.

D. Cazals' algorithms

Cazals proposed two algorithms. One is very similar to Ohtake's as it is based on the third order differential properties of the surface. On the contrary, the second algorithm uses P_{\min} values (4th order quantities). We will respectively call them Cazals3 and Cazals4 algorithms in the remainder of this article. The only distinction between Ohtake's and Cazals3 algorithms is on the last condition. Instead of testing a crest point, it focuses on a crest segment. On a triangle, if moving towards the crest segment in the t_{\min} direction k_{\min} increases for at least two vertices of the face, the tested segment is effectively a minimum of k_{\min} . For Cazals4 algorithm, the third specification is on the sign of P_{\min} at the ends of the crest segment $[r_1 r_2]$:

$$P_{\min}(r_i) > 0 \text{ with } i=1,2$$

Though different techniques to estimate k_{\min} at each vertex of the mesh are proposed in the articles describing the algorithms, the algorithms are compared with the same k_{\min} and e_{\min} values. Otherwise the comparisons would not only be made on the detection capacities of the algorithms. In the following, the minimal curvature and its derivatives are calculated with the Cazals's jet method. This method is based on the fitting of a bivariate polynomial on the neighborhood of each vertex. The polynomial degree is set to 4 in order to estimate the second-order derivative of k_{\min} used in Cazals' algorithm.

III. DATABASE

A large amount of MBES data were made available to us by SHOM (the French Hydrographic office) and the FPS Economy (Belgium Economy ministry) continental shelf service. These two offices respectively survey the French and Belgian continental shelves and so collect data over dunefields. Now, they have clearly distinctive objectives. In this regard, their databases are complementary.

The Belgian continental shelf is not as extensive as the French one and has already been entirely surveyed. Dredging being a major activity on the Belgian continental shelf, control boxes have been defined to monitor the impact of aggregate extraction on the sediment stocks. When the dredging is deemed to intensive, areas can be closed to protect the natural resources. Consequently, these boxes are surveyed regularly (few times per year). And, for some boxes, MBES data have been recorded for about fifteen years. These regions of interest are located in shallow areas (few tens of meters deep). The use of cutting-edge acquisition systems enables us to see great details of the complex seabed morphology.

The French available datasets were acquired on the continental shelf off the coast of Brittany, France. Since the whole continental shelf has been surveyed yet, most of the campaigns consist in running long lines. The French data can

sparingly contain dunes. Furthermore, the dunes are very likely to be partially described as survey lines are not overlapping. The depth ranges from a couple of meters to about two hundred meters deep. Though the main objective during these campaigns is to explore as much as possible the continental shelf, boxes are also surveyed when interesting dunefields are discovered. These interesting regions are surveyed less frequently than the Belgian boxes. Nevertheless, the French data show the diversity of dune shapes and sizes.

For both databases, acquisition equipment was changed from one survey to another. Henceforth, the data quality varies according to the survey. This must be considered as it necessarily modifies the triangular meshes and potentially disturb the crest detection. It is only problematic when comparing the detected crests on a same area at two different times.

In this paper, the comparisons between the crest detection algorithms will be made on two datasets, one from every database. As already mentioned, these databases differ in their objectives and contents. For this reason, it appears relevant to analyze the results on sets from both databases.

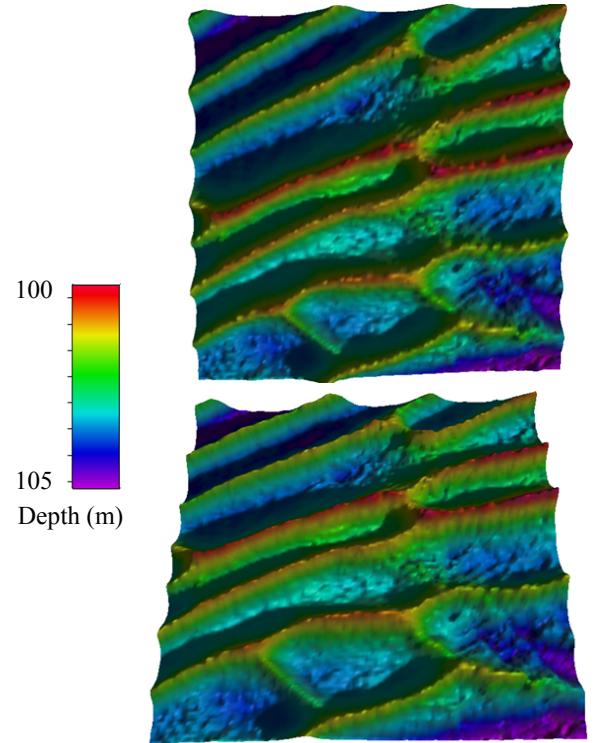


Fig. 2. Bathymetry of zone 1

Note that the vertical exaggeration is set to 10 on figures 2 and 3. Zone 1 is a 400x400 m area where dunes are about two or three meters high (see Figure 2). The dunes are parallel and occasionally split or merge. When looking between the dunes, one can notice that the triangular mesh is rough. This

roughness can be real seabed roughness or due to the data accuracy. Indeed, these dunes are more than 100 meters below the sea surface. The triangular mesh resolution was defined so that no edge is shorter than the MBES resolution, 3 m for zone 1.

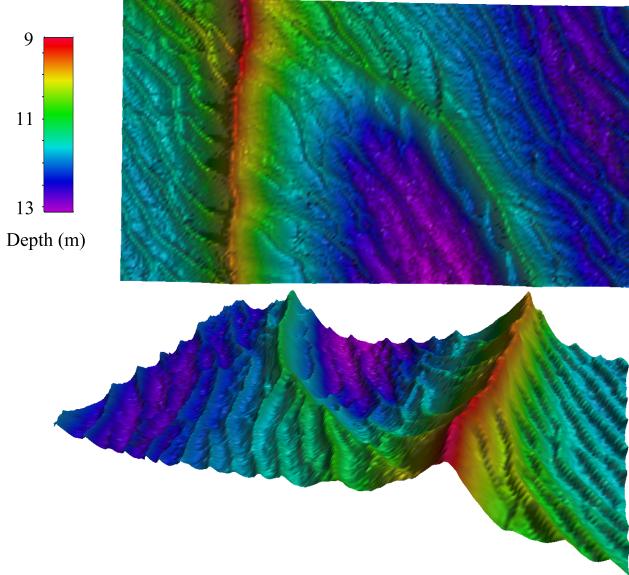


Fig. 3. Bathymetry of zone 2

The bathymetry in zone 2 is typical of the Belgian data (Figure 3). That is to say that there are dunes on top of which lie ripples. The area is 200x100 m wide with a mean depth of 11 m. There are two dunes in this zone. One is 2.5 m high and crosses zone 2. The other one is lower (~1.5 m high) and splits into ripples at one end. The ripples are about 20 to 30 cm high in average. The resolution of the triangular mesh on zone 2 is limited to 1 m.

The crest detection algorithms are basically designed to extract any locus of a mesh that locally look like Figure 1. In this regard, both zones are challenging. Furthermore, it will be interesting to observe how the algorithms will deal with the mesh roughness of zone 1 and the ripples of zone 2. In fact, the final goal is to correctly detect the dune crests only.

IV. RESULTS

In this section we present the results of the 4 algorithms applied to zone 1 and 2. The goal is to quantify the resemblance between the obtained crests and to determine whether dune crest detection can realistically be done with one of these algorithms.

Figure 6 shows that dune crests are fairly well-detected by all four algorithms. Nevertheless, as expected, dune crests are not the only seabed features extracted. The mesh rugosity visible on Figure 2 between the dunes causes the detection of small, spurious lines. With Hildebrandt's algorithm, the crests

seem smaller. It is problematic. This tendency to fragment the crests is problematic and confirmed by the figures in Table III. Hildebrandt's extract more lines than the others. The extracted lines are effectively shorter. In fact, the longest crests are ten times smaller than the dune crests detected with the three other algorithms. This tendency to construct small lines has not been witnessed in our work. The explanation is to be sought in the nature of the data used in this study.

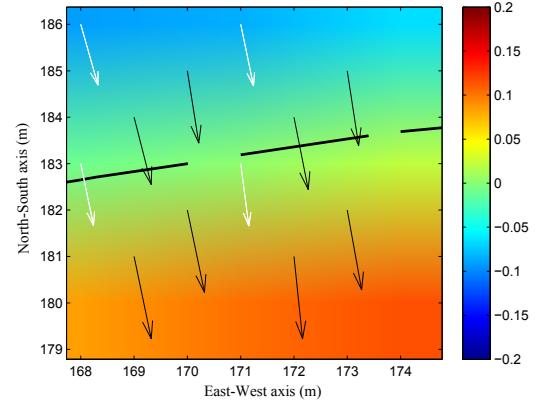


Fig. 4. Explanations on Hildebrandt's fragmented crest lines

Figure 4 displays a case where Hildebrandt's algorithm cut into pieces a dune crest. The white arrows give t_{\min} directions at the vertices and the blue correspond to the gradient of e_{\min} located at the centers of the faces. For visualization concerns, the vectors are scaled. In fact, dune crest segments are not detected as they do not comply with the third condition. Yet, on Figure 4, the faces containing these undetected segments seem to satisfy this condition. The issue comes from the fact that this condition is based on the inner product of vectors nearly orthogonal. e_{\min} barely varies on a face so that its gradient is almost vertical. t_{\min} vectors are tangent to the surface. But, the surface is flat. A dune is few meters high but hundreds of meters wide. Thus, t_{\min} vectors are almost horizontal. Their z component changes sign according to the dune flank. This little variation is sufficient for modifying the results of the third test. The specificity of our data makes the algorithm fail.

Ohtake's algorithm seems to correctly detect the crest but to be less sensitive to the seabed roughness than Cazals' algorithms. It is the algorithm that extracts fewest crests. The length quartiles confirm that the detected crests tend to be longer than with the other algorithms. The sinuosity is the ratio between the distance between the crest extremities and its length. It illustrates that Ohtake's crests are linear as expected for dune crests.

Nevertheless, Ohtake's dune crests are sometimes split as Hildebrandt's even though this is harder to notice. The only quantity depicting this phenomenon is the five longest

crests that are not as long as the ones from Cazals' algorithms.

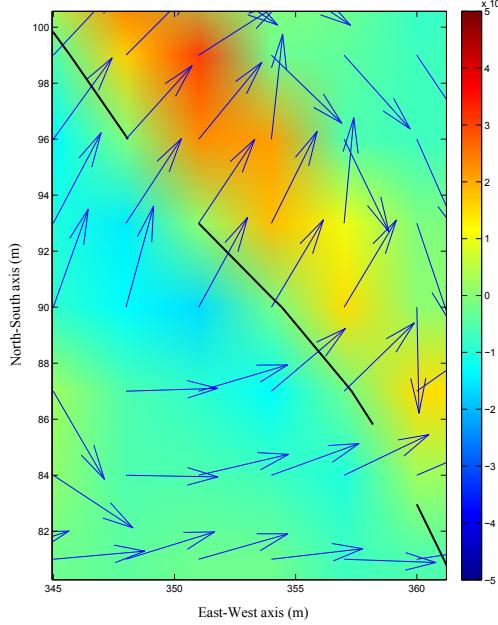


Fig. 5. Failure case of Ohtake's algorithm

In Figure 5, Ohtake's algorithm is not capable of entirely detect the dune crests. As for Hildebrandt's algorithm, Figure 5 shows a particular configuration. Indeed the crest segments that should be detected are almost parallel to the mesh edge it crosses. Besides, t_{\min} vectors are perpendicular to the crest. Hence the problem arising here is also a problem with the inner product of vectors almost perpendicular. A little variation of t_{\min} direction can influence the detection of a crest point and then cause the splitting of a dune crest. Unfortunately, the quality of the detection depends on the mesh. To reduce this fragmentation effect, a new version of the algorithm is proposed in [19]. The core idea is to merge crests that lined up to reconstruct full length crests .

Cazals' algorithms seem equivalently efficient for detecting dune crests (see Table III and Figure 6). But, Cazals4 detects more crests in the rough areas. It is visible with the lower values of sinuosity and the length quartiles. The difference are probably due to the fact that Cazals4 algorithm relies on P_{\min} values. P_{\min} values are derived from the 4th order differential properties of the surface that are less stable than the 3rd order quantities used in Cazals3 algorithm.

Table I contains information to compare the crest lines of two algorithms. It shows the number of mesh edges crossed

by the crests derived from both algorithms. In other words, it estimates the number of faces crossed by the crests from both algorithms. One can note more than a half of Hildebrandt's crest segments are common with the crests from the other algorithms. However, Ohtake's crests are almost all detected by Cazals' algorithms with more than 2700 Ohtake's crest segments out of 3100 that are also detected by Cazals' algorithms. And, Cazals3 crests are a part of Cazals4's (4100/4300 segments in common). On zone 2, the algorithms

TABLE I
CLOSENESS OF THE DETECTED CRESTS ON ZONE 1

	Hildebrandt	Ohtake	Cazals3	Cazals4
Hildebrandt	3524			
Ohtake	1432	3089		
Cazals3	1649	2783	4383	
Cazals4	1759	2763	4123	5253

detect dune crests and ripple crests. The observations are valid on zone1. It is essential to compare it with the results on Zone 2. Hildebrandt crests on Zone 2 are not as fragmented as on Zone 1 (see Figure 7), still, none is longer than 50 m (see Table IV). With this dataset, it is Cazals4 method that detects more crests. The difference with Cazals3 is more obvious. The length quartiles are much lower for Cazals4 crests. This is combined why lower values of sinuosity. Yet, the longest crests are still equivalently constructed for both Cazals' algorithms.

Ohtake's crests are less numerous again. They are longer in average as Ohtake's algorithm is less sensitive to the surface roughness. But the longest crests are not as well detected as with Cazals' methods because of the issue discussed above.

Table II validates the observations made for Zone 1 about the closeness of the four algorithm crest lines. Hildebrandt's crests are so distinct from the others probably because it uses e_{\min} values different from the others. In the end, Cazals3 algorithm is the closest to be a dune crest detection algorithm since gives good dune crests and with less small crest lines than Cazals4. Yet, the crests representing the other seabed features (local roughness, rocks, ripples, wrecks,etc.) are still to be removed from the detections.

TABLE II
CLOSENESS OF THE DETECTED CRESTS ON ZONE 2

	Hildebrandt	Ohtake	Cazals3	Cazals4
Hildebrandt	5491			
Ohtake	2597	4650		
Cazals3	2911	4379	5815	
Cazals4	2959	4339	5624	7193

TABLE III
COMPARISONS OF THE CRESTS DETECTED BY THE FOUR ALGORITHMS ON ZONE 1

Algorithm	Number of crests	Length(m) (quartiles)	Sinuosity (quartiles)	Five longest crests (m)
Hildebrandt	874	1.6 3.2 6.1	0.98 1 1	43 41 37 36 30
Ohtake	254	2.9 5.4 9.7	0.97 0.99 1	440 428 277 265 205
Cazals3	515	1.9 3.6 9	0.95 0.99 1	442 433 306 281 278
Cazals4	696	1.5 3.6 7.9	0.83 0.98 1	447 433 282 278 270

TABLE IV
COMPARISONS OF THE CRESTS DETECTED BY THE FOUR ALGORITHMS ON ZONE 2

Algorithm	Number of crests	Length(m) (quartiles)	Sinuosity (quartiles)	Five longest crests (m)
Hildebrandt	749	0.7 1.6 4.3	0.94 0.99 1	47 40 34 33 33
Ohtake	266	1.2 3.5 10.4	0.97 0.99 1	96 85 84 81 78
Cazals3	417	0.6 1.1 5.3	0.95 0.99 1	143 113 88 84 83
Cazals4	799	0.1 0.8 2.1	0.91 1 1	143 113 89 84 83

V. CONCLUSIONS AND FUTURE WORK

In this article we presented a framework for quantitative comparisons of crest detection algorithms. The comparison procedure was thought for the algorithms' detection performances on dunes. Yet, the procedure can surely be adapted for uses on other types of data. All four algorithms detect dune crests but not with the same quality. As shown, Cazals' algorithms are able to entirely extract the dunes when Ohtake's and Hildebrandt's algorithms tend to produce segmented crest lines. The main issue is that these algorithms are designed to detect all the crests in a surface and not specifically dune crests. Henceforth, the seabed roughness or seabed features are detected as well. Consequently, using such an algorithm to only detect the dune crests requires finding a solution for getting rid of the undesired crest lines. A possibility is to analyze dune crest properties and figure out whether or not one of their properties is unique. In other words, one must look for a property that would help in differing dune crests from other crests. Ultimately, the undesired lines could be filtered out.

Instead of filtering out lines, another solution would consist in not detecting the undesired crest lines in the first place. In geomorphometry [20], [21], notions of scale are used to describe landscape elements. The idea is that a feature of the terrain has not the same aspect according to the scale at which it is considered. The algorithms see crests where the seabed is rough but if the mesh was coarser, the seabed might seem smooth. Similarly, ripples could be smoothed out. The impact of the mesh resolution on the detections needs to be explored.

ACKNOWLEDGMENT

The authors would like to thank DGA and Region Bretagne for their financial support. The author would like to thank the Many thanks are also due to the FPS Economy Continental

Shelf Service for giving us a full access to their remarkable database. To finish, the authors are grateful to SHOM for providing us data on interesting areas and welcoming us onboard the hydrographic and oceanographic vessel *Beaufort-Beaupré* for a campaign.

REFERENCES

- [1] Y. A. Catano-Lopera, J. D. Abad, and M. H. Garca, "Characterization of bedform morphology generated under combined flows and currents using wavelet analysis," *Ocean Engineering*, vol. 36, no. 910, pp. 617–632, Jul. 2009. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0029801809000031>
- [2] S. Le Bot, D. Idier, T. Garlan, A. Trentesaux, and D. Astruc, "Dune dynamics: from field measurements to numerical modelling. Application to bathymetric survey frequency in the Calais-Dover Strait," *Marine Sandwave Dynamics*, pp. 101–108, 2000. [Online]. Available: <http://www.shom.fr/fileadmin/SHOM/PDF/04-Activites/sedimentologie/marid123/TPlidierleb.pdf>
- [3] A. A. Nmeth, S. J. M. H. Hulscher, and H. J. de Vriend, "Modelling sand wave migration in shallow shelf seas," *Continental Shelf Research*, vol. 22, no. 1819, pp. 2795–2806, Dec. 2002. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0278434302001279>
- [4] B. W. Flemming, "The role of grain size, water depth and flow velocity as scaling factors controlling the size of subaqueous dunes," in *Marine Sandwave Dynamics, International Workshop*, 2000, pp. 23–24. [Online]. Available: <http://www.shom.fr/fileadmin/SHOM/PDF/04-Activites/sedimentologie/marid123/TPflem prem.pdf>
- [5] T. Garlan, A. Cartier, M. Franzetti, P. Le Roy, J. Duarte, J. Pombo, M. Peix, P. Guyomard, Y. Le Faou, I. Gabelotaud, and others, "Complex morphology and organisation of dunes in a giant dunes field," *Proc. MARID 2013*, pp. 113–118, 2013.
- [6] K. Van Landeghem, G. Besio, H. Niemann, C. Mellett, D. Huws, L. Steinle, S. O'Reilly, P. Croker, D. Hodgson, D. Williams, and others, "Amplified Sediment waves in the Irish Sea (AmSedIS)," in *Van Lancker, V. and Garlan, T. Ed., 4th Conference MARID. Bruges, Belgium, VLIZ Special Publication*, vol. 65, 2013, p. 338p. [Online]. Available: <http://www.vliz.be/imisdocs/publications/246051.pdf>
- [7] P. L. Barnard, D. M. Hanes, D. M. Rubin, and R. G. Kvitek, "Giant sand waves at the mouth of San Francisco Bay," *Eos, Transactions American Geophysical Union*, vol. 87, no. 29, pp. 285–289, 2006. [Online]. Available: <http://onlinelibrary.wiley.com/doi/10.1029/2006EO290003/abstract>
- [8] J. Xu, F. L. Wong, R. Kvitek, D. P. Smith, and C. K. Paull, "Sandwave migration in Monterey Submarine Canyon, Central California," *Marine Geology*, vol. 248, no. 3–4, pp. 193–212, Feb. 2008. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0025322707002800>

- [9] M. a. F. Knaapen, "Sandwave migration predictor based on shape information," *Journal of Geophysical Research: Earth Surface*, vol. 110, no. F4, p. F04S11, Dec. 2005. [Online]. Available: <http://onlinelibrary.wiley.com/doi/10.1029/2004JF000195/abstract>
- [10] Y.-w. Guo, Q.-s. Peng, G.-f. Hu, and J. Wang, "Smooth feature line detection for meshes," *Journal of Zhejiang University SCIENCE*, vol. 6A, no. 5, pp. 460–468, May 2005. [Online]. Available: <http://www.zju.edu.cn/jzus/article.php?doi=10.1631/jzus.2005.A0460>
- [11] S.-K. Kim and C.-H. Kim, "Finding ridges and valleys in a discrete surface using a modified MLS approximation," *Computer-Aided Design*, vol. 37, no. 14, pp. 1533–1542, Dec. 2005. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S001044850500117X>
- [12] G. Stylianou and G. Farin, "Crest lines for surface segmentation and flattening," *IEEE Transactions on Visualization and Computer Graphics*, vol. 10, no. 5, pp. 536–544, Sep. 2004.
- [13] K. Watanabe and A. G. Belyaev, "Detection of Salient Curvature Features on Polygonal Surfaces," *Computer Graphics Forum*, vol. 20, no. 3, pp. 385–392, Sep. 2001. [Online]. Available: <http://onlinelibrary.wiley.com/doi/10.1111/1467-8659.00531/abstract>
- [14] M. P. Do Carmo and M. P. Do Carmo, *Differential geometry of curves and surfaces*. Prentice-hall Englewood Cliffs, 1976, vol. 2. [Online]. Available: <http://www.ulb.tu-darmstadt.de/tocs/25335766.pdf>
- [15] V. Interrante, H. Fuchs, and S. Pizer, "Enhancing transparent skin surfaces with ridge and valley lines," in *Proceedings of the 6th conference on Visualization'95*. IEEE Computer Society, 1995, p. 52. [Online]. Available: <http://dl.acm.org/citation.cfm?id=833836>
- [16] F. Cazals and M. Pouget, "Topology driven algorithms for ridge extraction on meshes," 2005. [Online]. Available: <https://hal.inria.fr/inria-00070481/>
- [17] K. Hildebrandt, K. Polthier, and M. Wardetzky, "Smooth Feature Lines on Surface Meshes," in *Symposium on geometry processing*, 2005, pp. 85–90. [Online]. Available: <http://www.konrad-polthier.de/articles/feature/featureSGP05.pdf>
- [18] Y. Ohtake, A. Belyaev, and H.-P. Seidel, "Ridge-valley lines on meshes via implicit surface fitting," *ACM Transactions on Graphics (TOG)*, vol. 23, no. 3, pp. 609–612, 2004. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1015768>
- [19] S. Yoshizawa, A. Belyaev, H. Yokota, and H.-P. Seidel, "Fast, robust, and faithful methods for detecting crest lines on meshes," *Computer Aided Geometric Design*, vol. 25, no. 8, pp. 545–560, Nov. 2008. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0167839608000435>
- [20] I. S. Evans, "Geomorphometry and landform mapping: What is a landform?" *Geomorphology*, vol. 137, no. 1, pp. 94–106, Jan. 2012. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0169555X11001462>
- [21] P. Fisher, J. Wood, and T. Cheng, "Where is Helvellyn? Fuzziness of multi-scale landscape morphometry," *Transactions of the Institute of British Geographers*, vol. 29, no. 1, pp. 106–128, 2004. [Online]. Available: <http://onlinelibrary.wiley.com/doi/10.1111/j.0020-2754.2004.00117.x/full>

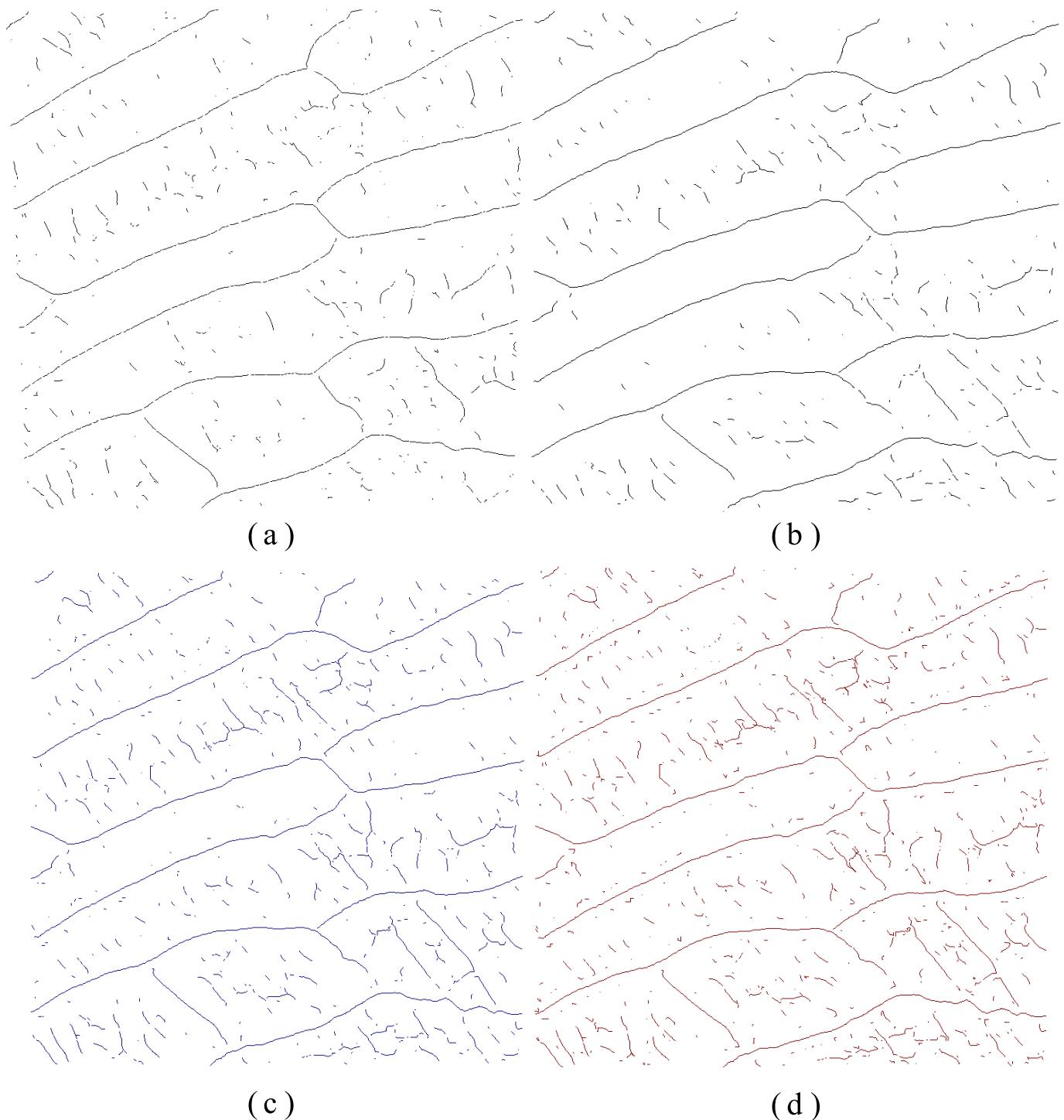
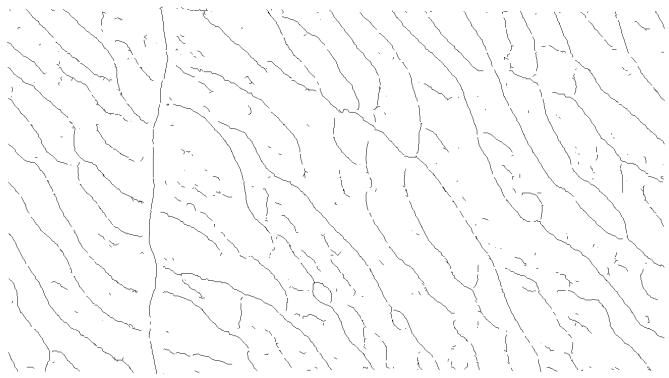
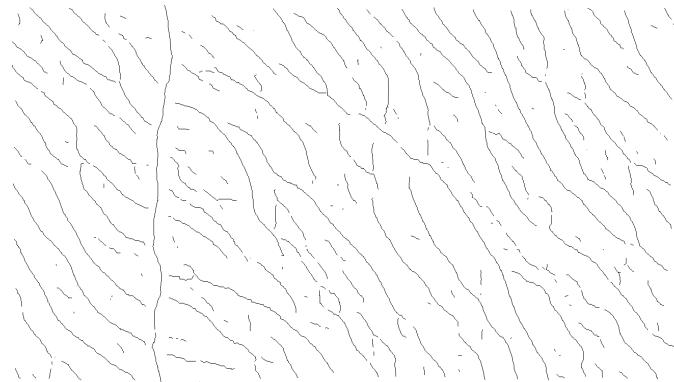


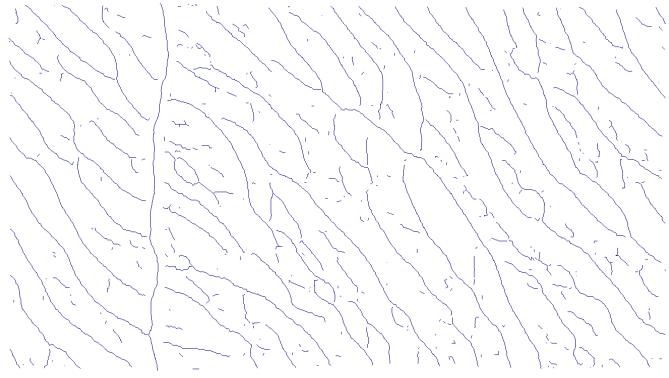
Fig. 6. Results of the four crest detection algorithms on zone 1: (a) Hildebrandt ; (b) Ohtake ; (c) Cazals3; (d) Cazals4.



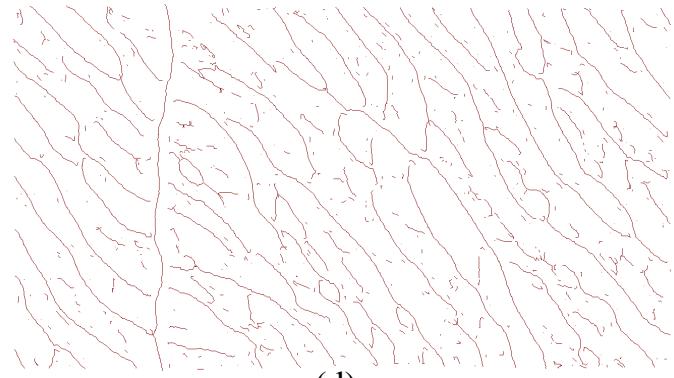
(a)



(b)



(c)



(d)

Fig. 7. Results of the four crest detection algorithms on zone 2: (a) Hildebrandt ; (b) Ohtake ; (c) Cazals3; (d) Cazals4.