



**HAL**  
open science

# A quantitative error measure for the evaluation of roof edge detectors

Baptiste Magnier, Alexandre Le, Albert Zogo

► **To cite this version:**

Baptiste Magnier, Alexandre Le, Albert Zogo. A quantitative error measure for the evaluation of roof edge detectors. IST 2016 - 2016 IEEE International Conference on Imaging Systems and Techniques, Proceedings, Oct 2016, Chania, Greece. 10.1109/IST.2016.7738264 . hal-01940350

**HAL Id: hal-01940350**

**<https://hal.science/hal-01940350>**

Submitted on 30 Nov 2018

**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.

# A Quantitative Error Measure for the Evaluation of Roof Edge Detectors

Baptiste Magnier, Alexandre Le and Albert Zogo

Ecole des Mines d'Alès, LGI2P, 69 avenue Parc scientifique Georges Besse, 30000 Nîmes, France

Email: baptiste.magnier@mines-ales.fr, alexandre.le@mines-ales.org, albert.zogo@mines-ales.org

**Abstract**—Crest line extraction remains a hard task in image processing. Indeed, these roof edges represent narrow edges on the image surface and whatever undesirable pixel close to or on the crest line may disturb the detection. This communication presents a new crest line detection overall evaluation. Comparing the ground truth contour image and the candidate crest line image, the proposed algorithm is based upon a new criterion that take into account the list of ground truth, the recall and their associated spacial nearness. Doubtlessly, an efficient evaluation penalizes a misplaced edge point proportionally to the distance to the true contour.

This quantitative performance evaluation proves its efficiency on several crest line images of different types, bringing a favorable indicator for tow closest edge images or a poor in the presence of a degraded/distorted candidate edge image.

## I. INTRODUCTION

In computer science, all the systems, especially automated information processing structures, must be evaluated before being developed in the industry or the medical data for principal examples. Image processing is no exception to this rule. In image segmentation, all these methods have to be tested and their quality must be measured, whether it is edge detection, point matching, region segmentation or image restoration (using the well known measures: PSNR or Similarity [1]).

In image processing applications, edge detection remains a key point as it can provide geometrical information [2][3]. Moreover, effective and precise edge detection [4] can prove highly useful. In most cases, due to the undesirable noise, textures, some other disturbances or parameters of the edge extractor themselves, most of edge detectors do not lead directly to object edges; then contours must be searched among numerous contour points. Furthermore, as shown in [5], crossing edges and corners which are cornerstone in image processing are not well detected with most of edge extractors. Nevertheless, evaluation of all image segmentation method type is an important labor. To perform this task, error measures must be quantified between the ground truth solution and a candidate edge image. These measures could be either quantitative or qualitative.

Qualitative assessments require interaction with human observers, and the performance indicators are based on subjective evaluation of the quality of the detected contours [6]. However, evaluations made by different human beings judging the quality of contour detector outputs might be very diverse.

Considering the error quantified as the difference between a truth solution and a candidate edge image, quantitative

measures are assigned numerical values. As described in [7], quantitative assessments are generally missing in proposed methods on edge detection. Nevertheless, in literature, some authors suggest [8][9] or compare different performance evaluations of contour detector measures [10][11].

Digital images embody different type of edges. Crest lines (or called roof edges) represent a special type of contours, as shown in Fig. 1. Classical edge detectors are optimized to extract step or ramp edges [4]; however, they fail to detect crest lines in images. Indeed, a step/ramp edge extractor will return two edges of both sides of the crest line because crest lines in an image represent narrow ridges or valleys on the image surface. Roof edges are defined as thin nets inside the image; describing roads [12] or rivers in satellite images, blood vessels in medical images [13] or roots in underground images. Therefore, finding these dense and thin structures is a significant task in image processing [14][15][16][17][18][19].

Concerning a curve in a grey level image  $I$ , ridges correspond maxima, valleys to minima [20][21]. Indeed, considering the image surface  $S$ , in the Cartesian space,  $\vec{S}$  defines all pixel coordinates:  $\vec{S}(x, y) = (x \ y \ I(x, y))^T$ . Let  $S_T(x, y) = \{\vec{S}_x(x, y), \vec{S}_y(x, y)\}$  be the tangent plan of the surface  $\vec{S}(x, y)$  in all points where:

$$\begin{cases} \vec{S}_x(x, y) = \frac{\partial \vec{S}(x, y)}{\partial x} = (1 \ 0 \ I_x(x, y))^T \\ \vec{S}_y(x, y) = \frac{\partial \vec{S}(x, y)}{\partial y} = (0 \ 1 \ I_y(x, y))^T, \end{cases}$$

with  $I_x$  and  $I_y$  the partial derivatives of  $I$  respectively, along the  $x$  and the  $y$  axis. Ridges and valleys are given by the points where the values of  $\vec{S}(x, y)$  are maxima or, respectively, minima in the orthogonal direction of the curve at  $(x, y)$ .

Even though the evaluation of crest lines lingers an open problem as it is for edge detection, some formulas suggest efficient ways to evaluate edge detection. Note that the main difference between step/ramp edge detection and roof

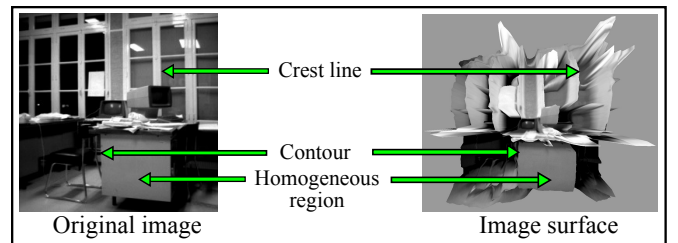


Fig. 1. Crest lines in an image.

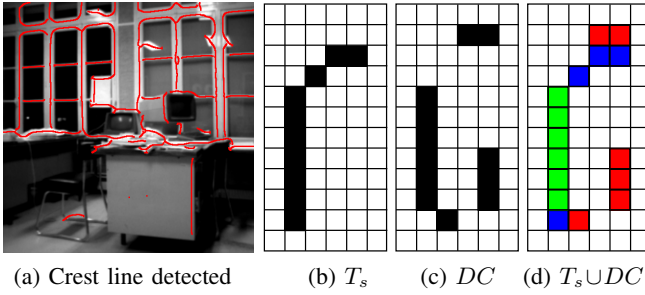


Fig. 2. Crest lines extracted in red on the image in Fig. 1, and illustration of TP (green), FP (red) and FN (blue) points. In (c),  $DC$  is contaminated with 6 FPs and 4 FNs, illustrated in (d). Note that  $FN \cup TP = T_s$ .

edge/crest line extraction is that, in the first case, edges are chosen between two pixels, whereas in the second crest lines site in the middle of the ridge/valley (single pixel: maxima -top of ridges- or minima -bottom of valleys-). A non exhaustive review is proposed in the following Section, leading to a contour/crest line detectors assessment.

## II. A REVIEW OF DISTANCE-BASED ERROR MEASURES

To provide an overall evaluation of the error assessment of edge detectors, the measure has to take into account both the amount of false positive points, false negative, but also of the displacement or distortion between the candidate edge image and the ground truth.

### A. Image Pixel Positions Assessment

Position is the first criterion to be assessed over an edge image. Its basic evaluation is compounded of statistics. To that effect, ground truth edge image or ground true skeleton ( $T_s$ ) and detected contours/crest lines ( $DC$ ) are merged. Afterward, all points are partitioned into three sets:

- True Positive points (TPs), common points of  $T_s$  and  $DC$ ,
- False Positive points (FPs), spurious detected edges of  $DC$ ,
- False Negative points (FNs), missing skeleton points of  $DC$ .

First, FPs appear in the presence of noise, texture or other contours influencing the detection in the image. Then, FNs represent a hole in a contour of  $DC$  (i.e. a discontinuous edge), generally caused by a blur or a poor pronounced edge/crest line in the original image  $I$ . Finally, a failing threshold of the segmentation could create both FPs and FNs. Computing only FPs and FNs [19][22] or combining these two statistics allows to display evaluations like Receiver Operating Characteristic (ROC) [23] or Precision-Recall (PR) [6]. As illustrated in Fig. 7 (top left), these procedures do not use spatial interpretation because they consider only a one-to-one matching process.

In the results, presented in Section IV, TPs, FPs and FNs (respectively represented in green, red and blue, see illustrations in Fig. 2 (b), (c) and (d) for more details) are outlined the proportions of points by color set within the resultant image. Regardless the image dimension and non spatial tolerance, an indicator reflecting the spacial precision of points called  $KPI_p$  can be created:

$$KPI_p = 1 - \frac{1}{\frac{P_b}{1+\epsilon_{FN}} + \frac{1-P_b}{1+\epsilon_{FP}}}, \quad \text{with } P_b \in [0, 1], \quad (1)$$

where  $\epsilon_{FN}$  and  $\epsilon_{FP}$  represent, respectively the proportion of FNs and FPs in the union of  $T_s$  and  $DC$  (denoted by the usual set-theoretic operation  $T_s \cup DC$ ) and  $P_b$  represents the weight of FNs (in our experiments,  $P_b=0.5$ ). Finally, this type of formula is called Key Performance Indicator ( $KPI$ ), with  $KPI \in [0, 1]$ . For a given experiment, a  $KPI$  value close to 1 means a poor segmentation. Alternatively, a  $KPI$  value close to 0 translates a good segmentation.

### B. Pratt's Figure of Merit criterion

In literature, a widely used similarity measure [24] between ground truth contours and detected edges, in respect with these criteria, is Pratt's Figure of Merit ( $FoM$ ):

$$FoM = \frac{1}{\max(\text{card}(T_s), \text{card}(DC))} \cdot \sum_{p \in DC} \frac{1}{1 + k \cdot d_{T_s}^2(p)}, \quad (2)$$

with  $d_{T_s}$  the distance between the  $p^{th}$  pixel of  $DC$  and the nearest pixel of  $T_s$  and  $k \in \mathbb{R}_+^*$  a positive real. The edge detection is considered as good if  $FoM$  is close to 1 and poor when this value tends to 0.

This measure, however, behaves abnormally, as shown in [25] and recalled in [11], two different edge candidate images produce the same  $FoM$ , whereas one is visually closer to the ground truth. Moreover, as pointed out in [11] and illustrated in Fig. 3,  $FoM$  is particularly sensitive to displacements, i.e. the errors grow up very fast when a small translation of the detected edges appears.

### C. Baddeley's Delta Metric (BDM)

Inspired by Hausdorff distance, Baddeley's Delta Metric (BDM) [8] is a measure which intends to estimate the dissimilarity between each element of two edge images:

$$\Delta_w^k(T_s, DC) = \frac{1}{|I|} \cdot \left[ \sum_{p \in I} |w(d_{T_s}(p)) - w(d_{DC}(p))|^k \right]^{1/k}, \quad (3)$$

where  $k \in \mathbb{R}_+^*$  and  $w$  represents a weighting concave function (in general  $w(x) = \min(\sqrt{n^2 + m^2}, x)$  for an image of size  $m \times n$ , with  $m$  and  $n \in \mathbb{N}^*$  [26]).

The main drawback of the BDM is its hypersensitivity to false positive points, i.e. this measure tends to over penalize images with false detections. Indeed, when a false positive pixel is far from the true edge, the  $|w(d_{T_s}(p)) - w(d_{DC}(p))|$  value creates a high impact on the evaluation, thus penalizing the measure.

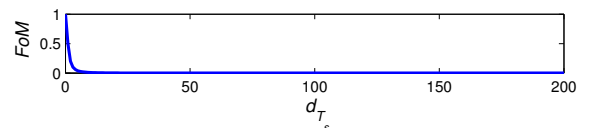


Fig. 3.  $FoM$  is very sensitive to the translations, here,  $\text{card}(DC) = 200$ .

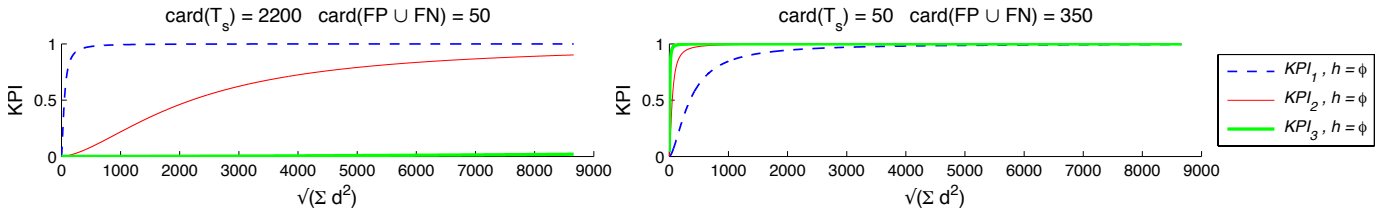


Fig. 4. Evolution of the  $KPI_{\{1,2,3\}}$  in function of the distance of the mistake points, with both different  $card(T_s)$  and  $card(FP \cup FN)$ .

### III. IMAGE SHAPE RESPECT ASSESSMENT

Even though the evaluation of crest lines lingers an open problem as it is for edge detection, some formulas suggest efficient ways to evaluate edge detection (see above). An evaluation of spatial position is far from being enough, since a dispersion (FP/FN points) around well-detected points (TP points) might be observed in the resultant image (see Fig. 8 for superposition of ground truth and segmentation/candidate images). To evaluate this dispersion on respect of image shape, a measure must be regarded. To that effect, we considered the Yasnoff *et al.* [27] measure as a start:

$$M_{YAS} = \frac{100}{card(FP \cup TP)} \cdot \sqrt{\sum_{p \in FP} d_{T_s}^2(p)}. \quad (4)$$

This measure has been used for binary images in pattern recognition, compared to other methods and thus proving its efficiency. The first noticeable fact is the possible values of this measure are between zero and infinity. Because it is not plain to draw out conclusions about this measure within its value window, we perform a transformation to only observe values between zero and one and build a  $KPI$  (see Section II-A) so that one would correspond to infinity in the distance and error:

$$KPI : [0; \infty[ \rightarrow [0; 1[ \\ u \rightarrow 1 - \frac{1}{1+k \cdot u^h}, \quad (5)$$

where  $k \in \mathbb{R}_+^*$  and  $h$  represents a power for observation. The parameter out of this formula corresponds to the distance  $\sqrt{\sum d_{T_s}^2}$  represented by  $u$ .

In this work, a displaced edge expects to be penalized in function of the false pixels number (i.e.  $card(FP \cup FN)$ ) and in function of the distance to the position it should be located at. Indeed, since  $FN \cap FP = \emptyset$ , using the Euclidian distance, the  $d_{T_s}$  is defined as:  $\forall p \in FP \cup FN$ ,

$$d_{T_s}(p) = \begin{cases} \text{Inf}\{\|p-t\|, t \in T_s\}, & \text{if } p \in FP \\ \text{Inf}\{\|p-t\|, t \in TP\}, & \text{if } p \in FN. \end{cases}$$

Inspired by Eq. 4, the first  $KPI$  formulation for shape respect is the following:

$$KPI_1 = 1 - \frac{1}{1 + \left( \frac{\sqrt{\sum_{p \in FP \cup FN} d_{T_s}^2(p)}}{card(FP \cup FN)} \right)^h}. \quad (6)$$

The latter by considering the whole image area, return non observable values by virtue of the image area importance. For

example, two different images  $I_1$  and  $I_2$  both containing only one FP point at a same distance of  $T_s$  ( $T_s = 10$  in  $I_1$  and  $T_s = 100$  in  $I_2$ ) obtain a common  $KPI_1$  value (according to the formula) through the evaluation interpretation is different, because of the unlike ratio values.

This led the optimum  $KPI$  research to a second formulation. That is the reason why we approached a formula to quantitatively observe dispersion regardless the image nature. This new term measures a pure dispersion of misplaced points compared to the cardinality of the ground truth  $T_s$ :

$$KPI_2 = 1 - \frac{1}{1 + \left( \frac{\sqrt{\sum_{p \in FP \cup FN} d_{T_s}^2(p)}}{card(T_s)} \right)^h}. \quad (7)$$

Thereafter, this consideration partially resolves the problem stated above. Indeed, considering two images with identical ground truth points  $T_s$  and dispersion distance, but different ratio, so that in one image a single FP point behaves as  $n$  FP points ( $n > 1$ ) on another image; these images have the same interpretation in terms of  $KPI_2$  which is not the case with ratios 1/100 and 90/100. The table I illustrates this drawback. A same reasoning is applicable to FN points. That means the distance divided by the cardinality of  $T_s$  must be balanced with the proportion of misplaced points, i.e.  $card(FP \cup FN)$ .

As demonstrated above and illustrated in Fig. 4,  $KPI_1$  and  $KPI_2$  suffer from different drawbacks. In edge/crest line detection assessment, a performing  $KPI$  has to take into account all the following input parameters  $T_s$ ,  $FP$ ,  $FN$  and  $d_{T_s}^2(p)$  ( $\forall p \in FP \cup FN$ ). Ultimately, this new  $KPI$  is proposed by the following formula:

$$KPI_3 = 1 - \frac{1}{1 + \left( \frac{\sqrt{\sum_{p \in FP \cup FN} d_{T_s}^2(p)}}{card(T_s)} \cdot \frac{card(FP \cup FN)}{card(T_s)} \right)^h}. \quad (8)$$

This last formulation takes into consideration all observable cases and theoretically observable. Since distance evaluation remains local so that there is no difference in the evaluation of these two images, it is to be noticed that image dimensions

TABLE I  
ERROR QUANTIFIED BY  $KPI_2$  ON TWO DIFFERENT CANDIDATE EDGE IMAGES COMPARED TO THE SAME GROUND TRUTH

$card(T_s)$	$d$	$card(FP \cup FN)$	$KPI_2$
100	5	90	0.2462
100	50	1	0.2616

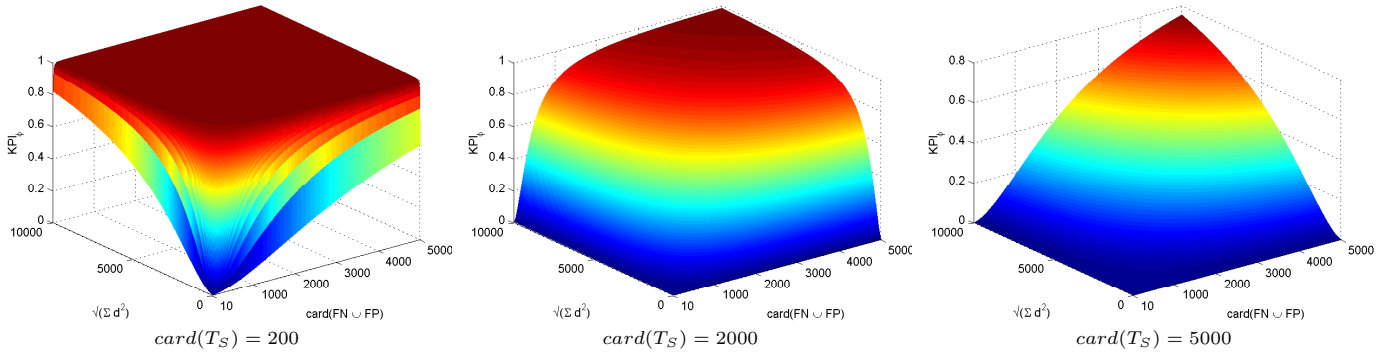


Fig. 5. Evolution of the  $KPI_3$  in function of the mistake points distance and  $\text{card}(FP \cup FN)$ , for different  $\text{card}(T_s)$  and fixing  $h = \phi$ .

are not considered. Moreover, a huge image may contain a skeleton with a small number of pixels. On the graphs presented in Fig. 4, we can notice how  $KPI_3$  evolves with the image characteristics (number of ground truth and misplaced points), compared to  $KPI_1$  and  $KPI_2$ .

An undeniable parameter of  $KPI_3$  formula is the power of the denominator term. It may be called a power of observation. Inasmuch as  $KPI_3$  depends on its value, it evolves more or less quickly around 0.5 and embodies a range of observable cases. We determined average values for the distance term  $d_{T_s}$ . The advice to choose values between 1 and 2 can be easily checked. Otherwise, the more  $KPI_3$  evolution will be abrupt, the less the transition between 0.5 and 1 will be marked (i.e. the slope of the  $KPI_3$  curve, for example  $h = 1$  in Fig. 6). Moreover, fixing  $h = 1$ ,  $KPI_3$  stagnates far from 1 when  $d_{T_s}$  becomes high. Additionally, when  $h = 2$ ,  $KPI_3$  starts to increase slowly and the slope becomes sharp around 0.5 to converge quickly towards 1. Finally, to fix the power at the golden ratio  $\phi \simeq 1.6180339887$  in order to ensure an evolution of  $KPI_3$  that would not be too abrupt from 0 to 1 and also not penalize when  $d_{T_s}$  is not elevated (contrary to  $KPI_3$  with  $h = 1$ , see Fig. 6). Fig. 5 illustrates  $KPI_3$  values in function of the distance of the mistake points and  $\text{card}(FP \cup FN)$ . For a small skeleton  $T_s = 200$ , it shows that  $KPI_3$  draws near 1 quickly when  $\text{card}(FP \cup FN)$  and  $d_{T_s}$  are growing. Alternatively, for a bigger skeleton,  $KPI_3$  rises slower. These three evolution surfaces show the coherence and robustness of  $KPI_3$  regarding our starting conditions: a displaced edge is penalized in function of the false pixel number and also of the distance to the position it should be located at.

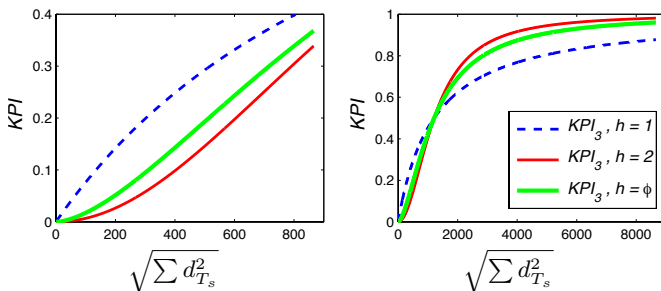


Fig. 6. Evolution of the  $KPI_3$  in function of the mistake points distance, for different powers  $h$ , with  $\text{card}(FP \cup FN) = 4000$  and  $\text{card}(T_s) = 2200$ .

## IV. EXPERIMENTAL RESULTS

In natural images, obtaining true contours or crest lines is not trivial, for this reason, some authors create their own benchmark and then evaluate the detection. New crest line images are proposed in the rest of this paper to evaluate roof edge extractors<sup>1</sup>.

### A. Construction of Test Images

Images are first created by building elementary shapes like lines or circles, since a skeleton can be decomposed into lines or arcs. Then, to create more complex images, we sum, then concatenate elementary images. Here is an example Fig. 7, a vertical line, a set of circular arcs and a tree.

### B. Results of Quality Measures

In the proposed experiments, to create candidate images, ground truth images are progressively degraded with FNs (i.e. substitution of pixels in  $T_s$ ), adding FPs (i.e. randomly noise pixels) or a displacement of each edge point of  $T_s$  and then compared with the initial image. Indeed, Fig. 7 shows  $KPI_{\{p,2,3\}}$  values in function of these 3 different deteriorations. Note that shape evaluation measurements ( $KPI_{\{2,3\}}$ ) depend on the image nature, i.e. edge geometry, especially for displacements and the edge shapes can be interpreted as decent. Curves presented in Fig. 7 and values available in Table II indicate that  $KPI_p$  (precision) is accurate and grows along the importance of the translations. It is the same for  $KPI_{\{2,3\}}$  (shape). However, on the one hand,  $KPI_2$  is sensitive to displacements and FPs but, on the other hand, does not penalize enough FNs addition (particularly visible for the vertical line case and in Fig. 9 (Arch vs Circle).

Fig. 8 and 9 present other images with their evaluations in Table II. Circle vs line and Comb images show how  $KPI_3$  behaves in the presence of two spatially distant edges and also contours images having  $DC$  and  $T_s$  close enough. Building Circle arc and Tree images with multiple sub images where each sub image possesses a valley which is different, thus a specific morphological operation is applied to the binary image

<sup>1</sup>These images and the edge detection quality assessment are created using *Matlab* software and available on: <http://www.mathworks.com/matlabcentral/fileexchange/58415-evaluation-of-roof-edge-detectors-with-a-quantitative-error-measure>

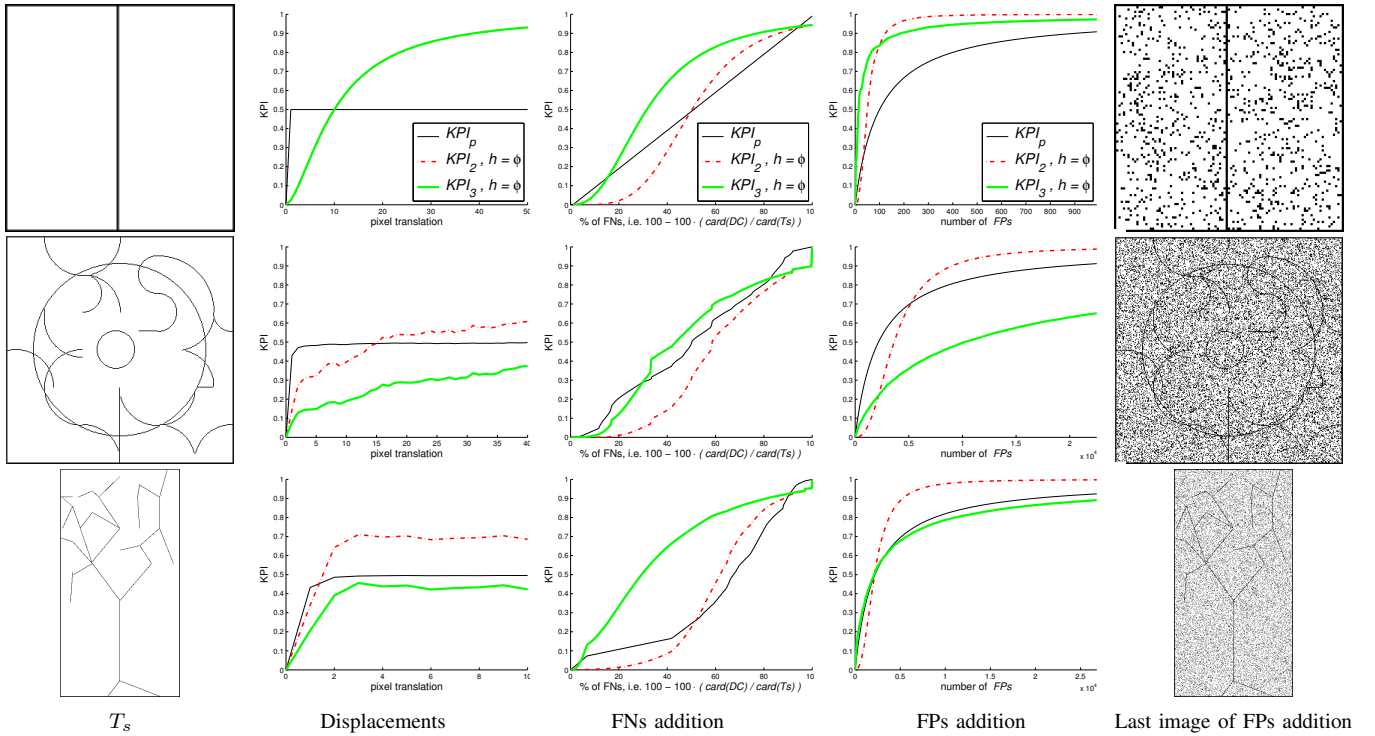


Fig. 7. Evolution of the  $KPI_{\{p,2,3\}}$  in function of the FPs, FNs and displacements.  $card(T_s) = 100$ , for the top image ( $100 \times 101$ ),  $card(T_s) = 2176$ , for the middle image ( $300 \times 300$ ) and  $card(T_s) = 2200$ , for the bottom image ( $580 \times 300$ ). Note that  $KPI_2$  and  $KPI_3$  curves are the same for pixel translation evaluation of the first image because  $card(T_s) = card(DC)$ . FNs are created by adding horizontal lines of null pixel values and FPs by randomly adding isolated pixels of value 1. For a better visualization, black and white colors are inverted and the reader is referred to the web version of this article.

until obtaining an image skeleton (involving a 8-connected neighborhood center element). Then, this skeleton is compared to ground truth valley images (presented in Fig. 7 left) and evaluated through the proposed  $KPI_{\{p,2,3\}}$ . Table II states also when Circle arc and Tree images endure a translation before the morphological operation, obviously,  $KPI_2$  penalizes sharply edge displacements (contrary to  $KPI_3$ ).

Results presented in these experiments illustrate the importance of a coherent evaluation. Indeed, penalizing a detected edge/crest line having either FPs and FNs or a small displacement is necessary proportionally to both the number of the mistake points and distance from ground truth. However, this penalty must not have a great influence in the presence of weak errors. Furthermore, the edge detector evaluation process must evolve from a good criterion to a poor criterion without

steep translations.  $KPI_3$  formula portrays better this state, and the use of this criterion enables to say whether a crest line extractor algorithm is efficient. Eventually, this crest line detector evaluation adapts to step/ramp edge detection evaluation, taking into account a spatial distance of the candidate edge from the true contours, contrary to the crest line where there is no distance (more details at the end of Section I).

## V. CONCLUSION AND FUTURE WORKS

This work aimed at a roof edge extraction evaluation using ground truth contour image and candidate edge image. Combining the number of true pixels compared to the total number of false pixels and their distances to the ground truth enable to build a criterion which predicts if an image is correctly segmented or not. The proposed algorithm allows to evaluate precisely the efficiency of a crest line detection method for different types of images of different sizes. Additionally, the method proves its efficiency to measure under- and over-edge detection. This approach could be employed to determine the best fitting crest line detectors parameters, like thresholds in the last stage of the detection.

Future works suggest an evaluation of several methods used in the crest lines detections. Another work consists of a multi-scale roof edge detection evaluation [22][21][16][18]. Indeed, adapting the proposed evaluation would measure the accuracy of crest line scale detection of multi-scale images.

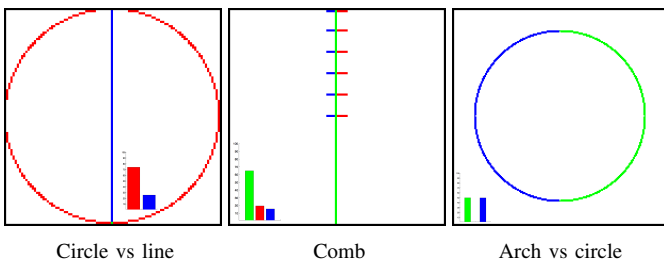


Fig. 9. Representation and ratio of TPs, FPs and FNs in: Circle vs line  $100 \times 100$ , Comb  $100 \times 100$  and Arch vs line  $201 \times 201$  pixels.

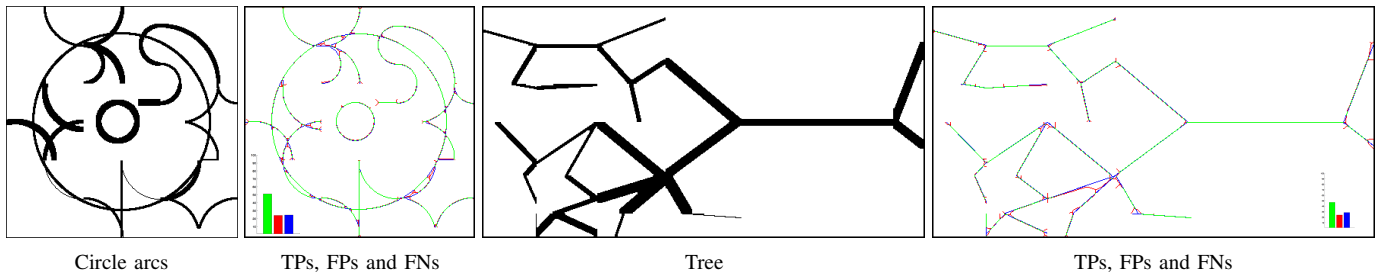


Fig. 8. Representation and ratio of TPs, FPs and FNs in: Circle arcs image 300×300 and Tree image 580×300. For a better visualization and an interpretation of the reference to color in legend, the reader is referred to the web version of this article.

TABLE II  
DIFFERENT  $KPI_{\{p,2,3\}}$  VALUES AND FOR THE IMAGES IN FIG. 8 AND 9

Image	x translation	y translation	%TP	%FP	%FN	$card(T_s)$	$card(FP \cup FN)$	$\sqrt{\sum d^2}$	$KPI_P$	$KPI_2$	$KPI_3$
Circle arcs	0	0	51.20	24.15	24.64	2176	1400	76	0.244	0.002	0.004
	+1	+2	10.79	44.35	44.86	2176	3488	519	0.446	0.174	0.089
	-2	+2	5.40	47.04	47.55	2176	3887	816	0.473	0.343	0.169
	+30	+20	1.22	47.10	51.69	2176	4063	1286	0.495	0.540	0.299
Tree	0	0	47.22	24.16	28.61	2200	1531	153	0.265	0.007	0.013
	-5	+10	1.03	46.96	52.00	2200	4105	1811	0.496	0.667	0.422
	+5	+10	1.59	46.92	51.48	2200	4079	1927	0.493	0.687	0.447
	+50	+100	0.49	42.95	56.56	2200	3837	3175	0.507	0.817	0.644
circle/line	0	0	0.26	74.02	25.71	100	828	384	0.620	0.996	0.968
Comb	0	0	64.90	19.50	15.60	124	54	23	0.176	0.016	0.060
arch/circle	0	0	50.10	0	49.90	1012	505	1531	0.5	0.388	0.661

## REFERENCES

- [1] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.
- [2] V. Torre and T. Poggio, "On edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 2, pp. 147–163, 1986.
- [3] D. Ziou and S. Tabbone, "Edge detection techniques: an overview," *International Journal on Pattern Recognition and Image Analysis*, vol. 8, no. 4, pp. 537–559, 1998.
- [4] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 679–698, 1986.
- [5] P. Montesinos and B. Magnier, "A new perceptual edge detector in color images," in *Advanced Concepts for Intelligent Vision Systems*. Springer, 2010, pp. 209–220.
- [6] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 530–549, 2004.
- [7] G. Papari and N. Petkov, "Edge and line oriented contour detection: State of the art," *Image and Vision Computing*, vol. 29, no. 2, pp. 79–103, 2011.
- [8] A. J. Baddeley, "An error metric for binary images," *Robust Computer Vision*, vol. 5978, 1992.
- [9] R. Román-Roldán, J. F. Gómez-Lopera, C. Atae-Allah, J. Martínez-Aroza, and P. L. Luque-Escamilla, "A measure of quality for evaluating methods of segmentation and edge detection," *Pattern Recognition*, vol. 34, no. 5, pp. 969–980, 2001.
- [10] S. Chabrier, H. Laurent, B. Emile, C. Rosenberger, and P. Marche, "A comparative study of supervised evaluation criteria for image segmentation," in *European Signal Processing Conference*. IEEE, 2004, pp. 1143–1146.
- [11] C. Lopez-Molina, B. De Baets, and H. Bustince, "Quantitative error measures for edge detection," *Pattern Recognition*, vol. 46, no. 4, pp. 1125–1139, 2013.
- [12] I. Laptev, H. Mayer, T. Lindeberg, W. Eckstein, C. Steger, and A. Baumgartner, "Automatic extraction of roads from aerial images based on scale space and snakes," *Machine Vision and Applications*, vol. 12, no. 1, pp. 23–31, 2000.
- [13] C. Kirbas and F. Quek, "A review of vessel extraction techniques and algorithms," *ACM Computing Surveys*, vol. 36, no. 2, pp. 81–121, 2004.
- [14] D. Ziou, "Line detection using an optimal IIR filter," in *Pattern Recognition*, 1991, pp. 465–478.
- [15] P. Montesinos and S. Dattenny, "Sub-pixel accuracy using recursive filtering," in *Scandinavian Conference on Image Analysis*, 1997, pp. 523–530.
- [16] T. Lindeberg, "Edge detection and ridge detection with automatic scale selection," *International Journal of Computer Vision*, vol. 30, no. 2, pp. 117–156, 1998.
- [17] C. Steger, "An unbiased detector of curvilinear structures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 2, pp. 113–125, 1998.
- [18] N. Armande, P. Montesinos, and O. Monga, "Thin nets extraction using a multi-scale approach," *Scale-Space Theory in Computer Vision*, pp. 361–364, 1997.
- [19] B. Magnier, P. Montesinos, and D. Diep, "Ridges and valleys detection in images using difference of rotating half smoothing filters," in *Advanced Concepts for Intelligent Vision Systems*. Springer, 2011, pp. 261–272.
- [20] R. M. Haralick, "Ridges and valleys on digital images," *Computer Vision, Graphics, and Image Processing*, vol. 22, no. 1, pp. 28–38, 1983.
- [21] B. Tremblais, A.-S. Capelle-Laize, and B. Augereau, "Algorithms for the extraction of various diameter vessels," *Cellular and Molecular Biology*, vol. 53, no. 2, pp. 62–74, 2007.
- [22] B. Magnier, A. Aberkane, P. Borianne, P. Montesinos, and C. Jourdan, "Multi-scale crest line extraction based on half gaussian kernels," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 5105–5109.
- [23] K. Bowyer, C. Kranenburg, and S. Dougherty, "Edge detector evaluation using empirical roc curves," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1. IEEE, 1999.
- [24] I. E. Abdou and W. K. Pratt, "Quantitative design and evaluation of enhancement/thresholding edge detectors," *Proceedings of the IEEE*, vol. 67, no. 5, pp. 753–763, 1979.
- [25] T. Peli and D. Malah, "A study of edge detection algorithms," *Computer Graphics and Image Processing*, vol. 20, no. 1, pp. 1–21, 1982.
- [26] C. Lopez-Molina, H. Bustince, J. Fernández, P. Couto, and B. De Baets, "A gravitational approach to edge detection based on triangular norms," *Pattern Recognition*, vol. 43, no. 11, pp. 3730–3741, 2010.
- [27] W. Yasnoff, W. Galbraith, and J. Bacus, "Error measures for objective assessment of scene segmentation algorithms," *Analytical and Quantitative Cytology*, vol. 1, no. 2, pp. 107–121, 1978.