Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 07, No. 03, September 2021.

A NEW ADAPTED CANNY FILTER FOR EDGE DETECTION IN RANGE IMAGES

Mohamed Cheribet[1] and Smaine Mazouzi[2]

(Received: 25-May-2021, Revised: 20-Jul.-2021, Accepted: 11-Aug.-2021)

ABSTRACT

Image segmentation remains as one of the most important tasks for image analysis and understanding. It deals with raw images in order to prepare them to be usable in automatic high-level processes, such as classification or information retrieval. We present in this paper a new adapted edge detector for range images. Its principle is inspired from the Canny detector, so the inherent features of range images will be considered. Usually, Canny detector is used with greyscale or color images, where its direct application with depths does not provide satisfactory results. From the raw image, containing measured depths, a relief image that consists of an image of normal vectors to the local surfaces is computed. So, angles between neighboring vectors are used to compute an angle-based gradient. The latter is integrated in the Canny algorithm, so an edge map is produced for the range image. Real images from the ABW database were used in experimentation, where the proposed new detector has outperformed the original Canny one by a ratio of 18 %.

KEYWORDS

Segmentation, Edge detection, Canny detector, Range images.

1. INTRODUCTION

Contrary to 2-dimentionnal (2D) color and grayscale images, range images allow non-ambiguous representation of observed scenes. They are so preferred in robotic vision, where the image analysis should be reliable and must provide a concise representation of the objects that they contain. In the past decades, it was not easy to produce depth images, given that the range devices were rare and mostly used within research laboratories and range image datasets were also few [1]. Since the commercialization of the Kinect device, range images have found an unceasing interest and datasets were multiplied during the last decade. However, the produced range images are highly distorted and affected by a high level of noise, which makes them hard to process according to conventional image analysis representations and methods.

Furthermore, image segmentation remains the most important and critical task for image analysis and recognition. It consists in the partition of an image in its intelligible parts that depend on the application to which it is dedicated. Therefore, choosing the well representation method and the well detection one is very important to ensure the expected efficiency of the whole image analysis system. The representation of raw data in images takes into account the nature of the features to be extracted and how they will be computed. Image segmentation aims at extracting from the raw image, according to the considered features, pixels of interest that can be used to delimitate parts of interest within the image, which are called regions. However, several artifacts make hard image segmentation, where every method should model and deal with such artifacts. For both main kinds of image artifacts, caused respectively by uncertainties and inaccuracies that are produced during image acquisition, several authors have proposed different methods for edge detection, based on stochastic processing [2] or fuzzy processing [3], where they have dealt with such kinds of artifacts. In [4], Mario and Morabito introduced and evaluated a fuzzy edge detector based on fuzzy divergence for edge detection, where a fuzzy entropy minimization was applied for threshold's selection. In another work, introduced by Fang et al. [5], the authors, aiming to deal with noise and intensity non-uniformity, used a fuzzy energy functional of both edges and regions. Region energy is composed of a hybrid fuzzy region term and a local fuzzy region term. The edge energy allows to maintain the appearance of smoothness by regularizing the pseudo-level set function (LSF) during the curve evolution.

1. M. Cheribet is with Department of Computer Sciences, Badji Mokhtar University, Annaba, Algeria. Emails: mcheribet@gmail.com, m.cheribet@univ-skikda.dz
2. S. Mazouzi is with Department of Computer Sciences, University of 20 Août 1955, Skikda, Algeria. Emails: mazouzi_smaine@yahoo.fr, s.mazouzi@univ-skikda.dz

Like for 2D images, range image segmentation methods are numerous. They differ according the feature representation method or according to the detection one. Therefore, range image segmentation methods can be split into three groups; namely: edge-based, region-based and classification- and clustering-based. Edge-based methods proceed by detecting discontinuities in the image data, using several techniques, mostly geometrical [6]-[7]. In range images, the proposed methods in the literature aim at extracting the borders of the regions in order to delimitate the latter [7]. Edge-based methods are well known for their reduced processing time. However, they suffer from lack of expressivity of the parts of images in terms of regions. Moreover, edge-detectors, such as Canny detector, are highly sensitive to noise and distortion [8], which makes them less-suited for range images.

Region-based methods rely on some homogeneity criteria to extract regions, based on the fact that those regions are composed of homogeneous pixels, according to the considered criterion. These methods are robust against noise, which makes them the most used for range images [9]-[10], [11]-[12]. Unfortunately, region-based methods are time-consuming and in most of the cases they depend on the seed from where the region starts to be extracted. Such a fact does not encourage their utilization with real-time applications, such as with robots and drones. Classification-based methods rely mostly on machine-learning techniques to label the pixels of the image according to a semantic criterion [13]-[14]. An active field of research was born from combining depth and color in RGBD (Red, Green, Blue, Depth) images, called object-based image segmentation, where machine-learning techniques are massively applied. Classification methods require learning, which cannot be always performed, due to lack of training data. Furthermore, extracted regions according these methods are not contiguous, which requires further processing.

In range images, noise processing is a dilemma. If it is strong, by using wide filters, or by applying the filter several times, some edges within the image will be smoothed and erased, so they will not be detected during segmentation. Otherwise, if the smoothing is under-performed, in order not to erase edges, the remaining noise within the image disturbs the detection and results in discontinuous and dislocated edges with wide regions of noise [15]. Fixing the level of smoothing is a challenge, in particular with acquisition constraints such as in robot vision systems. For instance, with range images, Canny detector produces mediocre results independently of the level of noise smoothing or other image enhancement techniques. Such an issue has motivated us to propose a new data representation in range images and reformulate some Canny steps, so that the resulting edge detector will be well-suited for range images.

So, in this paper, we are inspired from Canny detector for range images in order to propose a novel detector, well-suited for range images. Contrary to the conventional use of Canny, where the gradient vector at every pixel of the image is calculated based of the raw image data (color or gray level), we use angles formed by adjacent normal vectors to the surface to calculate both the magnitude and the direction of the gradient. Such calculation of the gradient allows to have high magnitudes on the pixels that are on the edges separating regions. However, magnitudes of the gradient are low on the pixels within the homogenous regions. In range images, adjacent pixels that belong to highly inclined surfaces have distant range values which do not allow to use the raw range as data to compute the gradient. Nevertheless, the orientations of the normal vectors are close within this type of surfaces, allowing to consider them for edge detection or region extraction.

The remainder of the paper is organized as follows: Section 2 introduces some well-referenced works having dealt with range image segmentation. Section 3 is devoted to the proposed method, in which we show how the surface-based image is computed and how the new gradient vector is computed; that is what consists our adaptation of the Canny detector for range images. Experimentation, results and a discussion are introduced in Section 4. Finally, Section 5 overviews the contribution of the introduced work and underlines its potential perspectives.

2. RELATED WORK

Range image segmentation depends strongly on the objects that can appear in images. According to such a fact, range image segmentation methods can be split into four families, depending on the used homogeneity criterion. These are: plane-based, curve-based, algebraic surface-based and continuity C1-based methods. In plane-based methods, suited geometrical criteria are used, such as plane equation and plane orientation. The considered criteria are then used to perform a region extraction or eventually an

edge detection, according to the adopted method. It is also necessary to deal with surfaces that have the same orientation but belong to different parallel surfaces. The method of Panrin and Medioni [15] belongs to this family, where the authors used a split and merge technique with the surface normal vector as image feature. Homogeneity is based on the angles between normal vectors and is used for region fusion as post-processing. A split and merge technique based on the gradient was also used in both Taylor [16] and Bhavsar [17].

In curve-based methods, authors use thresholding methods on the mean and the Gaussian curvatures. However, the estimation of curvatures is problematic in this category of methods. Indeed, the noise due to measurement errors and depth discontinuities does not allow a good estimation. Therefore, to obtain a good estimations of curvature, it is necessary to deal with noise and eliminate the disturbances due to discontinuities by an appropriate treatment. Besl and Jain [18] used this method to set initial seeds for growing regions. Detecting discontinuities for a curve-based segmentation was proposed by Yokoya and Levine [19]. In another work, Kasvand [20] pre-processed pixels belonging to local neighborhood in order to mitigate effects due to discontinuities.

Segmentation into algebraic surfaces, which are not strictly plane, concerns two categories of 2.5-dimensional (2.5D) and 3-dimensional (3D) surfaces. 2.5D surfaces that match polynomial functions of two variables can be applied only for scalar images. For 3D surfaces, which correspond to quadric or super-quadric surfaces, they require more complex processing than 2.5D ones. Gupta and Bajcsy [21]-[22] presented a segmentation method which produces as a result the description of the range image by super-quadrics such as ellipsoids. Jiang and Bunke [23] used the growing region method under 2.5D constraints of plane approximation. This special growing method is based on line segments, where regions are formed by lists of segments. So, growing a region is executed segment after segment and linear segments are delimited according to a profile division (by column or by row) of the processed image.

In continuity C1-based methods, segmentation is based on a given criterion defining the homogeneity of a surface, which is called C1-continuity. Two principles can be distinguished:

1) Merging of segments resulting from a more constrained segmentation [18].
2) The growing of detected border points is performed to form closed borders [24].

Recently, Deep Neural Networks have been widely used for image classification and object detection and recognition. However, in several cases, post-processing based on extracted edges is required to accomplish the recognition task, such as in the work introduced by Y. Wong et al. [25], where after having recognized Racing Bib Numbers (RBNs) by using YOLOv3, they proceeded by non-maxima suppression in order to predict a single bounding box for each target object.

After reviewing the different methods, we can conclude that most of them are based on former implementations that were used with color or grayscale images. Following the same roadmap, we present in the remainder of this paper a novel edge algorithm, inspired from Canny detector, aiming at accurately detect edges in range images.

3. ADAPTED EDGE DETECTION FOR RANGE IMAGES

In this section, we present our method for edge detection in range images and introduce the necessary basic algorithms to implement it. In an earlier work [26], we have introduced an overview of the proposed detector. So, in the current paper, we extend the work by introducing the full method and its extensive experimentation, as well as a result comparison with those of other 2D-image dedicated detectors.

3.1 Original Canny Detector

The Canny edge detector [27] is an edge detection that uses a multi-stage algorithm to detect a wide range of edges in 2D images. Filter-based edge detection consists in locating high impulsion responses of the used filter. The approach used by Canny is based on the quality of criteria required for an optimal edge detector. Canny's algorithm, which uses a gradient calculation operator, such as Sobel, is designed to be optimal according to three criteria:

- *Good detection:* the algorithm should mark the whole real edges in the image as much as possible.

- *Good localization:* edges marked should be as close as possible to the real edges.
- *Minimal response:* a given edge in the image should only be marked once and where possible image noise should not create false edges.

To satisfy these requirements, Canny used the calculus of variations, a technique which finds the function which optimizes a given functional. The optimal function in Canny detector is described by the sum of four exponential terms, but can be approximated by the first derivative of a Gaussian.

Canny detector allows to detect edges that correspond to significant and quick variations in image data. However, it can ignore slow variations resulting in losing edges in the final edge map. Indeed, on the one hand, if the used threshold of the gradient norm is low, the resulting edge map contains all the true edges, but with a high amount of noise. On the other hand, if the threshold is high, noise is low, but several true edges could be ignored.

Before introducing the proposed detector for range images, we review the steps that the original Canny detector follows. They are as follows:

1. *Noise reduction*: The first step is to reduce noise before detecting edges. The 2D filter uses a Gaussian [28] that is expressed as follows:

$$G_\sigma(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-x_0)^2+(y-y_0)^2}{2\sigma^2}} \quad (1)$$

2. *Gradient calculation:* After noise reduction, the next step is to apply a gradient calculation, which returns intensities of edges. The gradient operator, (Roberts, Prewitt, Sobel) [29] for example, returns an estimation of the first derivative in the horizontal direction (G_x) and the vertical direction (G_y). The gradient norm N and its direction θ are then calculated at every pixel in the image:

$$N(x,y) = \sqrt{G_x(x,y)^2 + G_y(x,y)^2} \quad (2)$$

$$\theta = tan^{-1}\left(\frac{G_y(x,y)}{G_x(x,y)}\right) \quad (3)$$

   The edge direction angle is rounded to one of four angles, representing horizontal, vertical and the two diagonals (0, 90, 45 and 135 degrees).

   First and using the gradient norm N, a thresholding is performed aiming at keeping only true candidates of edges. Such an operation results in a first edge map, represented by a 2D array *EdgeMap:*

$$EdgeMap(x,y) = \begin{cases} 1 & if N(x,y) > threshold \\ 0 & else \end{cases} \quad (4)$$

3. *Non-maximum suppression*: The gradient map records an intensity at each pixel in the image. A high intensity indicates a high probability that an edge is present at that pixel. However, this intensity is not enough to decide whether a pixel corresponds to an edge or not. Only pixels corresponding to local maxima are considered edge pixels and are kept for the next step of the algorithm. A local maximum is located at the pixel where the derivative of the gradient norm is null.

4. *Double threshold:* Two different thresholds are used and two edge maps are calculated, one with a low threshold and the other with a high threshold. Low threshold is used to identify the non-relevant pixels (intensity lower than the low threshold). So, it can detect the majority of edges and even noise. High threshold is used to identify the strong pixels (intensity higher than the high threshold). So, it only finds true and noise-free edges. However, it can miss some true edges. Moreover, non-maxima suppression should be applied to both edge maps.

5. *Hysteresis thresholding:* The differentiation of the edges on the generated map is done by hysteresis thresholding. This requires two thresholds; a high threshold and a low one, which will be compared to the intensity of the gradient of each pixel. If the intensity of the gradient of each point is less than the low threshold, the pixel is rejected. If it is greater than the high threshold, the pixel is accepted as forming an edge. If it is between low threshold and high threshold, the pixel is accepted if it is connected to an already accepted pixel. Therefore, the two obtained edge maps are merged where the second, obtained with the high threshold, is preferred, if needed. So,

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 07, No. 03, September 2021.

we begin with the edges from the second map, that are mostly noise-free and then complete with edges from the first map.

The pseudo-code below shows a recursive version of thresholding by hysteresis:

```
Algorithm Hysteresis Thresholding
Inputs
Map1: Edges map 1 (with lower threshold)
Map2: Edges map 2 (with higher threshold)
Outputs
Map2: Final edges map
Begin
NeighboringStack←Empty Stack
For each edge point Edges(i,j) of Map2 Do
  NeighboringStack.Push(Edges(i,j))
  While NeighboringStack is not empty Do
    V←NeighboringStack.pop
    For each point P of the 8 Neighbors of V Do
      If P is an edge-point in Map1 but not in Map2 Then
            Add P to Map2
            NeighboringStack.Push(P)
      EndIf
    EndFor
  EndWhile
EndFor
End.
```

3.2 Proposed Edge Detector for Range Images

Range images represent the distances to the surface points from an observer, usually situated at the range camera. Therefore, the calculation of the gradient vector by first derivation of the raw image function is not suited for range images. This is due to the discontinuities that exist between different objects or between surfaces of the same object.

Unlike 2D images, where the intensity values of the neighboring pixels are enough to calculate the gradient, in range images pixels in a local 2D neighborhood do not necessarily represent true 3D neighborhood, resulting in what is called 3D bias (Figure 1).

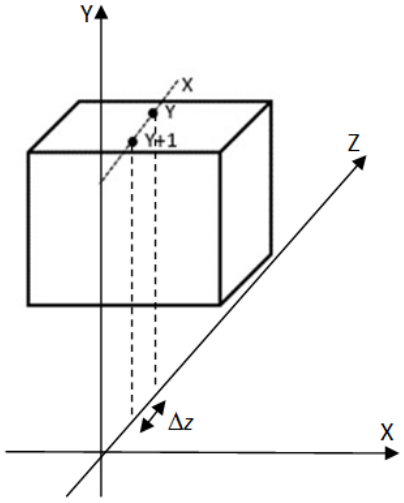


Figure 1. 3D bias in range images: (x,y) and (x,y+1) are close in 2D, but distant according Z ($\Delta z >> 1$).

The discontinuity of the data constitutes a characteristic of range images compared to color or grayscale images. Applying the original Canny filter to range images results in both high amount of outliers and lost edges. In the image in Figure 1, the distance $\Delta z = |I(y + 1, x) - I(y, x)|$ is high when the surface on which the two points are situated, is oriented closely orthogonal to the observation direction. Indeed, pixels (x, y) and (x, y + 1) are neighbors in the plane (X, Y), but they are far from each other in space. In 2D color or grayscale images, this situation does not happen, because the differences are uniform regardless of the surface orientation.

Therefore, a well-appropriated set of features should be synthesized and then applied for gradient computation in range images. For our proposed detector, we compute a new surface-based image, by

fitting the plane equation at every pixel of the raw image and then we consider the parameters of the obtained equation as the set of features for gradient vector calculation. Therefore, we can summarize our contribution in two points:

1) Generating a new surface-based image that allows to quantify variations of normal vectors to surfaces. Such variations are the basis for the calculation of the new gradient vector.
2) Adapting Sobel filter to use angles from the surface image rather than raw data from the range image (as in 2D images).

The calculation of the plane equation at a given pixel is performed either by cross-product considering two pairs of vectors, defined by three adjacent pixels, or by a multi-regression technique, as in several works in the literature [1]. Nevertheless, it has been noticed that most of the reviewed methods compute approximations of the fitted plane for an entire surface, which makes the calculation of the equation less accurate. To overcome such issue and as we focused only on how the edge can be detected locally, we were interested in how to represent the surface locally and to be able to calculate an appropriate gradient vector allowing edge detection according to the Canny detector steps. So, we will be only restricted to a limited neighborhood around each pixel, aiming at checking whether it is an edge pixel or not.

Moreover, range images are highly noisy by nature and require a smoothing operation to reduce noise without erasing edges. This should be done in the preprocessing step. In addition, the calculation of the gradient vector involves the use of the features obtained from a set of pixels belonging to a local neighborhood. Thus, an error occurring at a given pixel can be recovered by taking into account the features of the neighboring pixels.

According to our adaptation, the Canny steps for range images are modified only for the method to calculate the gradient vector and to synthesize suited features that must be used. The other steps remain unchanged (Figure 2). Indeed, the latter steps do not depend on what features have been used.

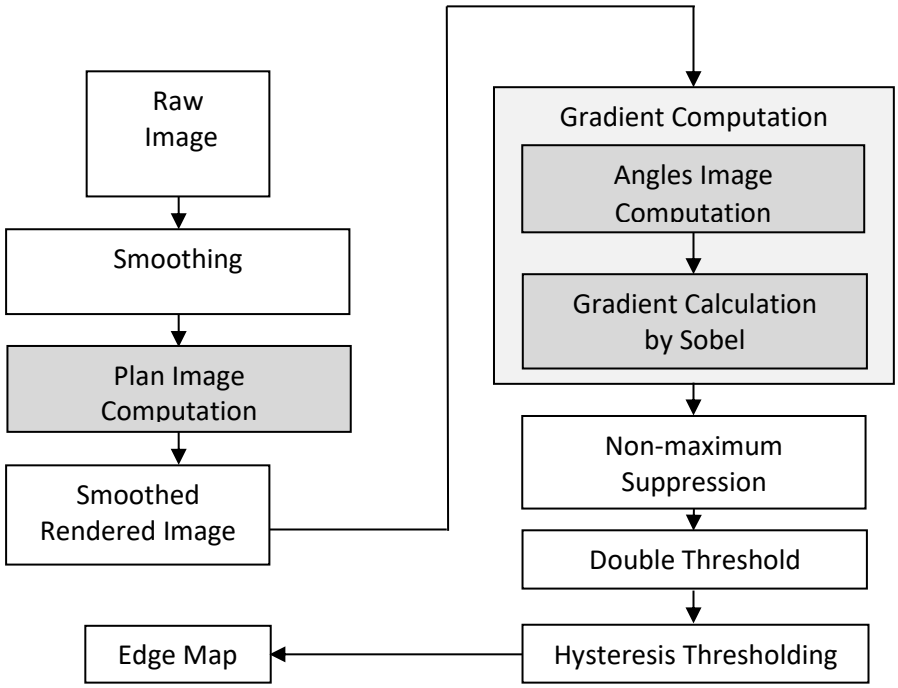


Figure 2. Flowchart of the overall proposed detector. The calculation of the smoothed image is moved outside the Canny algorithm, so the surface-based image can be computed. Steps in gray are those where adaptation is performed.

In the next part, we will present our new proposed method for calculating the gradient. This method has the advantages of being fast and remaining well suited to the peculiarity of range images that was introduced above.

3.3 Angle-based Edge Detection

Our proposed method consists in calculating a gradient vector, which measures the level of variation in

surface orientation. Therefore, the estimation of the gradient vector at a given pixel (*x, y*), requires the calculation of plane equations according to the directions defined by the neighboring pixels. The norm and the direction of the gradient are calculated by using the angle between the obtained normal vectors at the pixels in the neighborhood. However, for a given pixel, we calculate the equations of the planes defined according to the eight possible subsets of 4 pixels in the neighborhood, as shown in Figure 3.

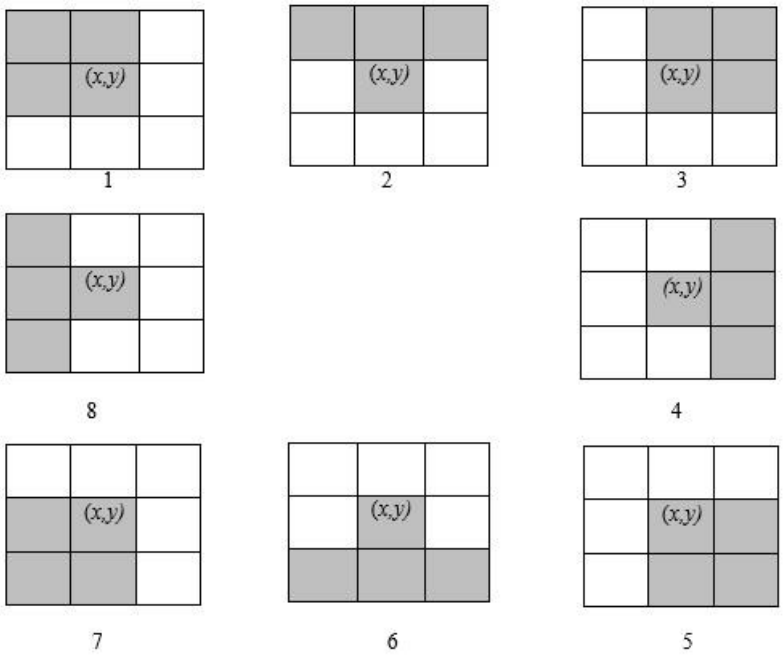


Figure 3. Used subsets of pixels to fit the plane equations for one pixel (*x,y*). All combinations of the central pixel with its 3 neighbors are considered.

To approximate the norm of the gradient that allows us to test whether a given pixel can be an edge pixel or not, we compute the equations of the planes in the neighborhood. For each plane, we use the central pixel and three of its neighbors:

Given a pixel (*x, y*), we use the set of neighboring pixels $\{(x + \Delta x, y + \Delta y); \Delta x, \Delta y = -1 .. 1\}$ to define eight planes. Table 1 below shows the list of pixels involved in the calculation of each equation.

Table 1. Sets of pixels involved in the calculation of the eight plane equations.

1	*(x, y) ; (x-1, y) ; (x-1, y-1) ; (x, y-1)*
2	*(x, y) ; (x-1, y-1) ; (x, y-1) ; (x+1, y-1)*
3	*(x, y) ; (x, y-1) ; (x+1, y-1) ; (x+1, y)*
4	*(x, y) ; (x+1, y-1) ; (x+1, y) ; (x+1, y+1)*
5	*(x, y) ; (x+1, y) ; (x+1, y+1) ; (x, y+1)*
6	*(x, y) ; (x+1, y+1) ; (x, y+1) ; (x-1, y+1)*
7	*(x, y) ; (x, y+1) ; (x-1, y+1) ; (x-1, y)*
8	*(x, y) ; (x-1, y+1) ; (x-1, y) ; (x-1, y-1)*

By considering the set *S* of the involved pixels, the equation of the plane $ax + by + cz + d = 0$ is calculated by linear multi-regression. As all the surfaces are visible and oriented towards the observer, the parameter c is necessarily non-zero. Therefore, the equation of the plane can be expressed by $z = \alpha x + \beta y + \gamma$, where α, β and γ are obtained after minimizing the objective function $\Phi(\alpha, \beta, \gamma)$ that expresses the least squares as follows [30]:

$$\Phi(\alpha,\beta,\gamma) = \sum_{p \in S} \left(\alpha x_p + \beta y_p + \gamma - I(x_p, y_p)\right)^2 \quad (5)$$

In the next step, we calculate the angle $\phi_{x,y}$ formed by the two normal vectors for each pair of pixels $\{(x, y) ; (x + \Delta x, y + \Delta y)\}$. Then, we calculate the gradient vector by introducing a modified Sobel operator, as follows:

According to the plane equation $z = \alpha x + \beta y + \gamma$, the normal vector is $\vec{V}_1(\alpha, \beta, -1)$. The angle $\phi_{x,y}$ defined between the normal vector and the vertical plan *(XY)*, expressed by the vector $\vec{V}_2(0,0,1)$, is calculated as follows:

$$Acos\ \phi_{x,y} = \frac{\vec{V}_1 * \vec{V}_2}{||\vec{V}_1|| * ||\vec{V}_2||} \quad (6)$$

The set of angles $\{\phi_{x,y}\}$formed by 3D normal vectors to the surfaces and the vertical plane *(XY)* are considered to calculate the gradient vector, instead of the pixel values $I(x, y)$ (Figure 4).

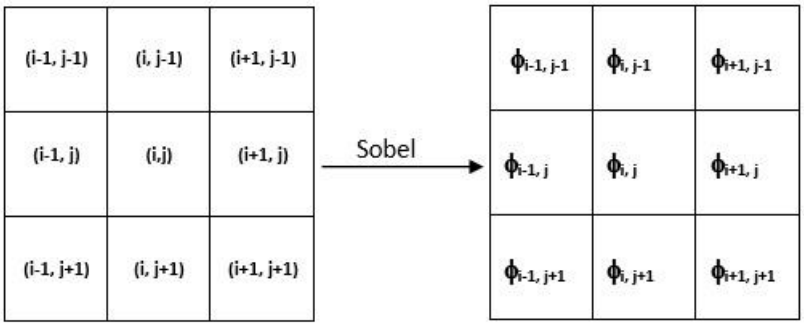


Figure 4. New features for gradient calculation using a modified angle-based Sobel detector. Raw ranges are transformed into angles.

The new gradient vector components (G_x^{ϕ}, G_y^{ϕ}) are obtained by the Sobel horizontal and vertical operators. The angle $\theta = Atan\ (\frac{G_y^{\phi}}{G_x^{\phi}})$ defines the gradient vector direction.

The obtained norms and angles of the gradient vector, so computed at every pixel of the range image, are used as inputs for the further steps of the Canny algorithm, which remain unchanged for our proposed detector.

4. EXPERIMENTATION

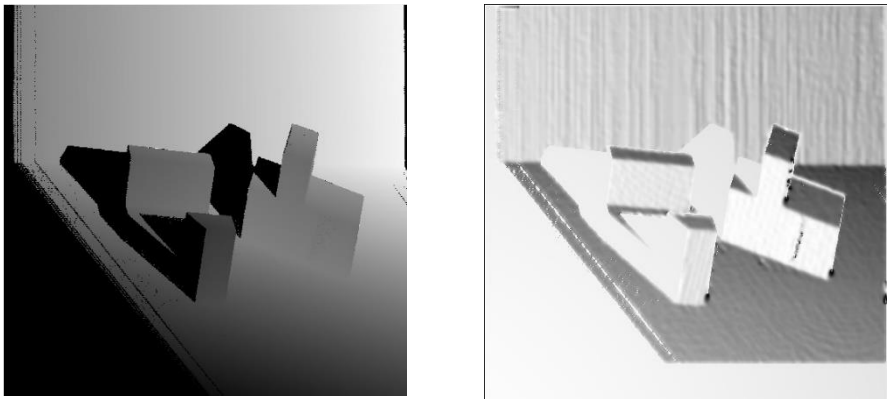
For the experimentation of our method, we used the ABW database that was dedicated to range image segmentation [1]. As for the original Canny detector, Gaussian smoothing is performed in order to reduce the noise in the image. After several tests by varying the parameter σ of the Gaussian filter in the range [0.1 .. 1.0], we have set σ to 0.8, for which edges in the tested images were correctly detected. Using Gaussian smoothing instead of other noise filtering techniques is due to the nature of noise within range images. Indeed, in such images, noise in mainly due to distortions in image data rather than to impulsive outliers, where median filters could be well-suited.

4.1 Qualitative Evaluation

As in our earlier work, published in [26], we begin by introducing some samples of range images from the ABW database, with their edge detection results using the original Canny detector and the proposed one. Such a visual presentation of the results allows the reader to have an idea on the advanced visual quality of edge detection by using our proposed detector. Figure 5.a shows a range image from the ABW database. It is displayed according to its raw depth data. The black regions are the shadows where the laser ray did not reach. At each pixel, the depth is represented by a gray level ranging from 0 to 255. In Figure 5.b, which represents a rendered image of the raw data, using a simple rendering algorithm with simulation of a light source, we can notice high level of noise as distortions of the plane surfaces, due to less precise range measurements.

The result of the application of the original Canny detector, using the raw image data, is shown in Figure 6.a for the high threshold and in Figure 6.b for the low threshold. The first image (Figure 6.a) shows clearly an under-detection of edges, where we can notice that several edges have not been detected, in particular those which form the borders between the surfaces of the same object. For such edges, there is a continuum in range data, so the gradient vector will be continuous at these points. On the other hand, we can clearly notice an over-detection of edges in the second image (Figure 6.b). Indeed, many

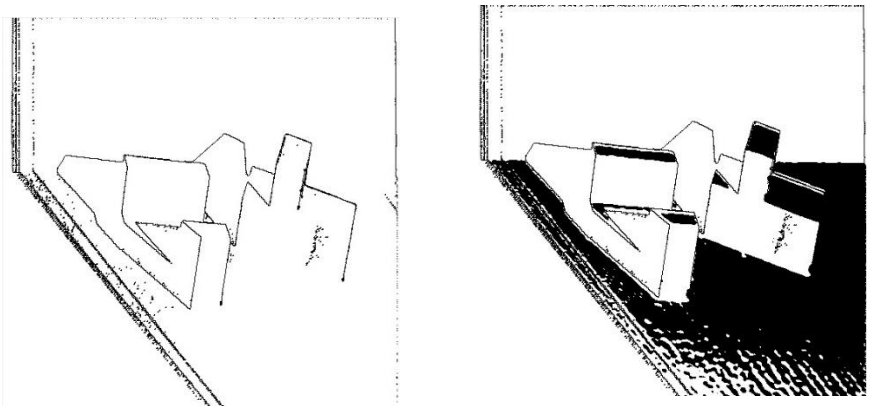
Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 07, No. 03, September 2021.



(a) (b)

Figure 5. A sample image from the ABW dataset, (a) Range image, (b) Rendered image showing the nature of noise in such images.

pixels inside different planar surfaces were detected as edge pixels, when they are not. Otherwise, most of the edges defining borders between surfaces have been detected. However, the pixels of some surfaces highly inclined were detected all as edge pixels. The later stages of Canny's algorithm; namely non-maximum suppression and thresholding by hysteresis, cannot improve such results in the first image and generate a lot of outliers in the second.

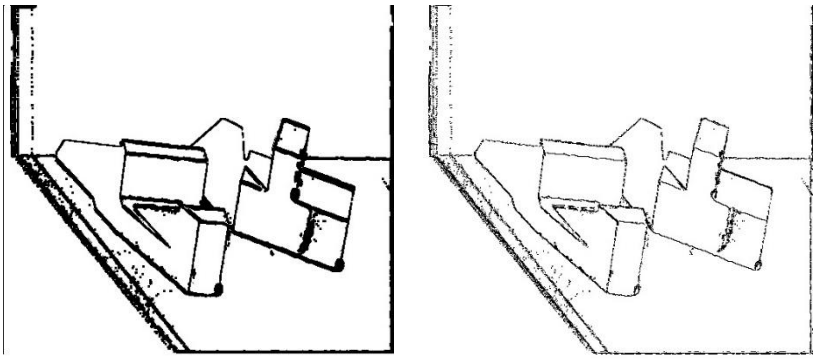


(a) (b)

Figure 6. Edge detection in the image abw.test.3 with the original Canny detector that uses the raw data: (a) Detected edges with a higher gradient threshold (0.2), (b) Detected edges with a lower gradient threshold (0.12). An under-detection is noticed in (a), where an over-detection is noticed in (b).

After applying our proposed detector, we were able to obtain the results presented in Figure 7. The detection results obtained before non-maxima removal and hysteresis thresholding steps are shown in Figure 7.a. We can notice, unlike the original Canny detector, that our adapted detector which uses a modified gradient, based on the variation of the normal angles, allowed to produce an edge map, usable for the further Canny steps where an adequate post-processing can be performed.

The final detection result is shown in Figure 7.b. This result is obtained after having applied non-maxima suppression and hysteresis thresholding on the image gradient of Figure 7.a. The best value for the lower threshold according to the proposed detector was 0.12 radian (6.86°) and that of the higher threshold was 0.20 radian (11.46 °). Such values were obtained by varying the two thresholds in the range [0.05 .. 0.25], by 0.05 as step. We have considered the values for which the detector result was the best for the set of ABW training images. We conclude that the visual results of edge detection were satisfactory for the whole set of test images. We can also say that the edges were correctly detected, including those belonging to intra-object boundaries. The latter are difficult to detect as it has been reported in all the works having dealt with range images segmentation.



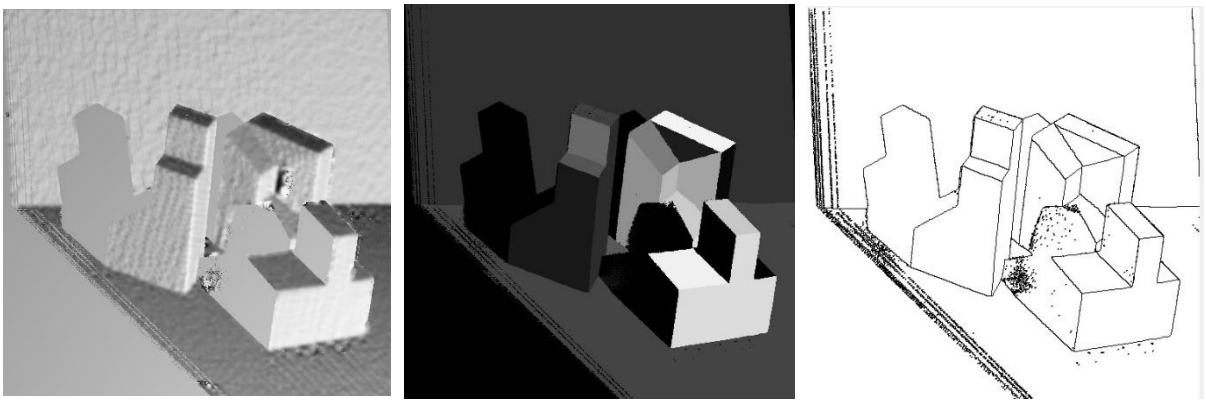
(a) (b)

Figure 7. Edge detection in the image abw.test.3 with the proposed detector: (a) Detected edges before non-maxima suppression and hysteresis, (b) Detected edges after non-maxima suppression and hysteresis (final result).

4.2 Quantitative Evaluation

As far as we know, there is no native method for edge detection in range images; all the proposed methods we found in the literature are region-based or Machine Learning-based [31]. Moreover, the pure range images such as those of ABW, or those of other RGBD datasets that provide depth modality, such as OSD (Object Segmentation Database) [32], do not provide a ground-truth edge detection. Therefore, it will be hard to quantitatively evaluate edge detection-based methods for range images.

Nevertheless, we were able to generate the ground truth of edge maps from the region-based ground truth segmentation. Figure 8 shows a sample image from the ABW dataset; namely abw.test.8, where we can see in Figure 8.b and Figure 8.c the ground truth of regions and the edges we have generated from it.



(a) (b) (c)

Figure 8. (a) A sample image from ABW dataset, (b) Ground truth of regions, (c) Generated ground truth of edges.

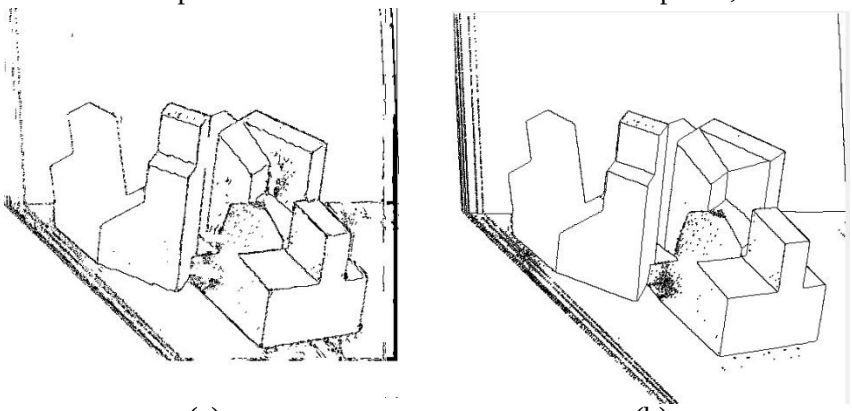
After having extracted the edge map from the ground truth region-based segmentation, provided by the ABW dataset, we consider the Dice index as a metric to evaluate the quality of the edge detection and to compare the results obtained by the proposed detector with those of the original Canny detector.

The Dice index [33] allows to express the gap between an edge map, produced by our detector and the corresponding ground truth edge map. This index is calculated based on the following elements:
True Positives (*TP*): Number of correctly detected edge pixels.

False Positives (*FP*): Number of pixels wrongly detected as edge pixels (absent in ground truth).
True Negatives (*TN*): Number of true edge pixels that were not detected.
Depending on *TP*, *FP* and *TN*, the Dice index is expressed as follows:

$$\kappa = \frac{2 \times TP}{2 \times TP + FP + TN} \tag{7}$$

The Dice index for the edge map produced in Figure 9.a regarding the ground truth edge map in Figure 9.b is 0.8642, where the three parameters were as follows : *TP* = 12353 pixels, *FP* = 3214 and *TN* = 666.



(a) (b)

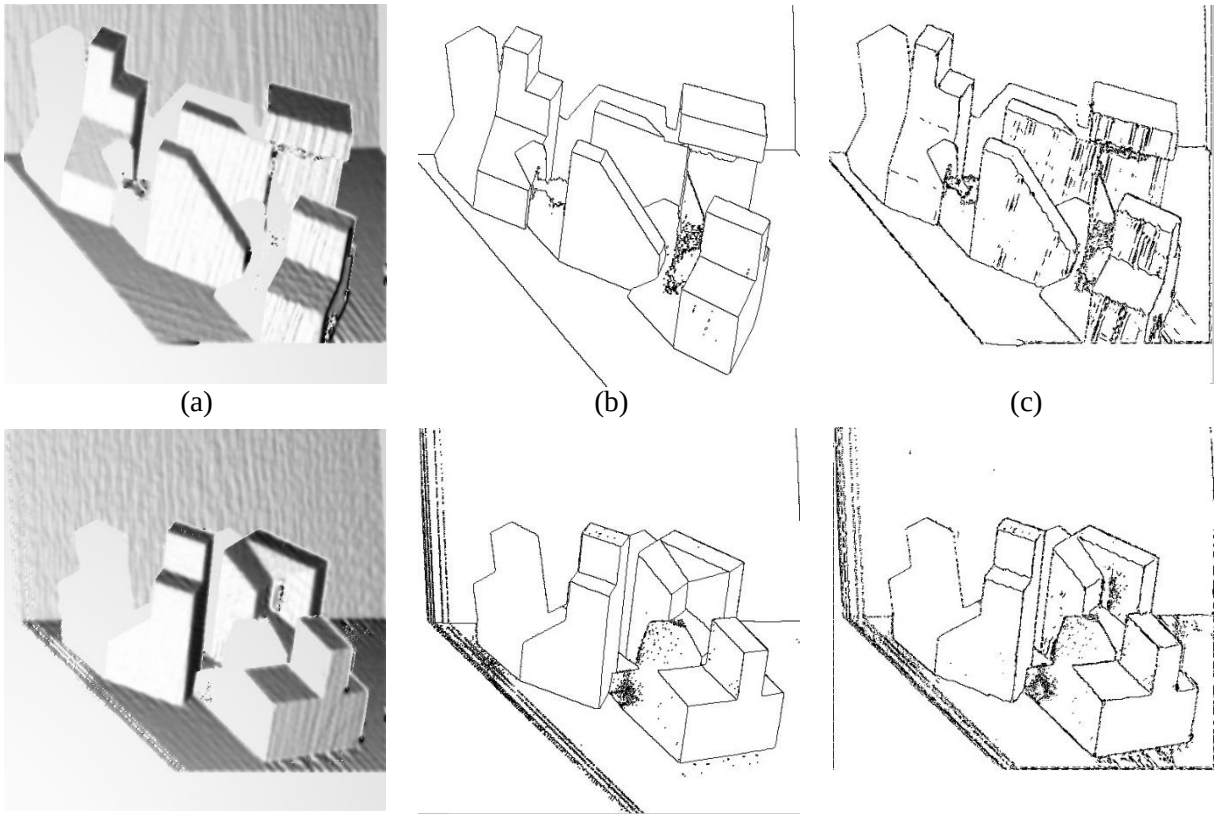
Figure 9. Comparison between detected edges and ground truth: (a) Detected edges by the proposed detector, (b) Ground truth edges.

For the whole 30 images of the ABW database, the results according to the Dice index are introduced in Table 2.

Table 2. Mean, max. and min. Dice index for the whole ABW dataset.

Mean Dice	Standard-deviation	Max. Dice	Min. Dice
0.8238	0.0335	0.8642 Obtained with abw.test.8	0.7629 Obtained with abw.test.0

Figure 10 shows the visual detection results for both images having scored highest and lowest; namely, abw.test.8 and abw.test.0.



(a) (b) (c)

(d) (e) (f)

Figure 10. Edge detection results corresponding to the lowest (abw.test.0) and the highest Dice index (abw.text.8), (a) Rendered smoothed abw.test.0, (b) Ground truth detected edges, (c) Detected edges by the proposed detector, (d) Rendered smoothed abw.test.8, (e) Ground truth detected edges and (f) Detected edges by the proposed detector.

Table 3. Detection result comparison according to Dice index, involving the proposed detector and the original Canny detector.

	Mean Dice	Standard Deviation	Max. Dice	Min. Dice
Proposed detector on surface-based image	0.8238	0.0335	0.8642	0.7629
Original Canny on raw image	0.6972	0.0539	0.7813	0.5998

According to the results introduced in Table 3, we can affirm that the proposed detector for range images has considerably enhanced the detection of the edges in range images. For the mean value of the Dice index, the enhancement is about 18%, which will be very helpful for the post-processing of range images. Moreover, it is more stable, given that its standard deviation is significantly lower than that of the original Canny detector.

5. CONCLUSION

Segmenting range images, given highly noisy nature of the latter, is considered as a hard task. In practice, the nature of the processing and the nature of the image make certain segmentation methods advantageous over others. In this paper, a new edge detector for range images is proposed. The main contribution consists in an adaptation of some steps of the original Canny detector, so the new detector can be appropriately applied for edge detection in range images. Mainly, we have introduced a new method for gradient calculation after having noticed that the classical derivative calculation based on raw data cannot be applied to range images. This is due to discontinuity between different objects or between surfaces of the same object. Therefore, instead of using raw data, gradient calculation is based on a new generated surface-based image that allows to compute angles between normal vectors and object surfaces. By using such angles, we calculate new gradient norms and directions and use them for the further steps of the Canny algorithm. Like most edge detectors, our new detector is fast and can be used for real-time applications. Experimental results and their comparison with those obtained by the original Canny algorithm have allowed to state that the proposed detector is efficient and well-suited for range images. In future work, combination of range data (Depth) and color data (RGBD images) could be considered to further test and evaluate the proposed new detector.

REFERENCES

[1] A. Hoover, G. Jean-Baptiste, X. Jiang, P. J. Flynn, H. Bunke, D. B. Goldgof, K. W. Bowyer, D. W. Eggert, A. W. Fitzgibbon and R. B. Fisher, "An Experimental Comparison of Range Image Segmentation Algorithms," IEEE Trans. on Pattern Analysis and Machine Intell., vol. 18, no. 7, pp. 673–689, 1996.

[2] C. S. Won, R. M. Gray and M. Robert, Stochastic Image Processing, Springer Science & Business Media, ISBN 978-1-4419-8857-7, 2004.

[3] H. Bustince, M. Pagola, A. Jurio, E. Barrenechea, J. Fernández, P. Couto and P. Melo-Pinto, "A Survey of Applications of the Extensions of Fuzzy Sets to Image Processing," Bio-inspired Hybrid Intelligent Systems for Image Analysis and Pattern Recognition, Springer, Heidelberg, vol. 256, pp. 3-32, 2009.

[4] M. Versaci and F. C. Morabito, "Image Edge Detection: A New Approach Based on Fuzzy Entropy and Fuzzy Divergence," International Journal of Fuzzy Systems, vol. 23, no. 5, pp. 918–936, 2021.

[5] J. Fang, H. Liu, L. Zhang, J. Liu and H. Liu, "Region-edge-based Active Contours Driven by Hybrid and Local Fuzzy Region-based Energy for Image Segmentation," Information Sciences, vol. 546, no. 6, pp. 397-419, 2021.

[6] T.J. Fan, G.G. Medioni and R. Nevatia, "Segmented Description of 3-D Surfaces," IEEE Journal on Robotics and Automation, vol. 3, no. 6, pp. 527–538, 1987.

[7] X. Jiang and H. Bunke, "Edge Detection in Range Images Based on Scan Line Approximation," Computer Vision and Image Understanding, vol. 73, no. 2, pp. 183–199, 1999.

[8] M. Basu, "Gaussian-based Edge-detection Methods: A Survey, " IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews), vol. 32, no. 3, pp. 252–260, 2002.

[9] Y. Ding, X. Ping, M. Hu and D. Wang, "Range Image Segmentation Based on Randomized Hough Transform," Pattern Recognition Letters, vol. 26, no. 13, pp. 2033–2041, 2005.

[10] A. Bab Hadiashar and N. Gheissari, "Range Image Segmentation Using Surface Selection Criterion," IEEE Transactions on Image Processing, vol. 15, no. 7, pp. 2006–2018, 2006.

[11] D. Holz and S. Behnke, "Fast Range Image Segmentation and Smoothing Using Approximate Surface Reconstruction and Region Growing," Intelligent Autonomous Systems, vol. 12, pp. 61–73, 2013.

[12] D. Holz and S. Behnke, "Approximate Triangulation and Region Growing for Efficient Segmentation and Smoothing of Range Images," Robotics and Autonomous Systems, vol. 62, no. 9, pp. 1282–1293, 2014.

[13] S. Gupta, R. B. Girshick, P. Andres Arbelaez and J. Malik, "Learning Rich Features from RGB-D Images for Object Detection and Segmentation," Proc. of the European Conference on Computer Vision, arXiv:1407.5736, pp. 345–360, 2014.

[14] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez and J. Garcia-Rodriguez, "A Survey on Deep Learning Techniques for Image and Video Semantic Segmentation, " Applied Soft Computing, vol. 70, pp. 41–65, 2018.

[15] B. Parvin and G. Medioni, "Segmentation of Range Images into Planar Surfaces by Split and Merge," Computer Vision Pattern Recognition, pp. 415-417, 1986.

[16] R. W. Taylor, M. Savini and A. P. Reeves, "Fast Segmentation of Range Imagery into Planar Regions," Computer Vision, Graphics and Image Processing, vol. 45, no. 1, pp. 42-60, 1989.

[17] A. V. Bhavsar and A. N. Rajagopalan, "Inpainting Large Missing Regions in Range Images," Proc. of the 20th IEEE International Conference on Pattern Recognition, pp. 3464-3467, Istanbul, Turkey, 2010.

[18] P. J. Besl and R. C. Jain, "Segmentation through Variable-order Surface Fitting," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 10, no. 2, pp. 167-192, 1988.

[19] N. Yokoya and M. D. Levine, "Range Image Segmentation Based on Differential Geometry: A Hybrid Approach," IEEE Trans. on Pattern Analysis and Machine Intell., vol. 11, no. 6, pp. 643-649, 1989.

[20] T. Kasvand, "The k1k2 Space in Range Image Analysis," Proc. of the 9th IEEE International Conference on Pattern Recognition, IEEE Computer Society, pp. 923-926, Rome, Italy, 1988.

[21] A. Gupta and R. K. Bajcsy, "Integrated Approach for Surface and Volumetric Segmentation of Range Images Using Biquadrics and Superquadrics," Applications of Artificial Intelligence X: Machine Vision and Robotics, International Society for Optics and Photonics, vol. 1708, pp. 210-227, 1992.

[22] A. Gupta and R. Bajcsy, "Volumetric Segmentation of Range Images of 3D Objects Using Super Quadric Models," CVGIP: Image Understanding, vol. 58, no. 3, pp. 302-326, 1993.

[23] X. Jiang and H. Bunke, "Fast Segmentation of Range Images into Planar Regions by Scan Line Grouping," Machine Vision and Applications, vol. 7, no. 2, pp. 115-122, 1994.

[24] A. Davignon, "Contribution of Edges and Regions to Range Image Segmentation," Applications of Artificial Intelligence X: Machine Vision and Robotics, International Society for Optics and Photonics, vol. 1708, pp. 228-239, 1992.

[25] Y. C. Wong, L. J. Choi, R. S. S. Singh, H. Zhang and A. R. Syafeeza, "Deep Learning-based Racing BIB Number Detection and Recognition," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 5, no. 3, pp. 181-194. 2019.

[26] C. Mohamed and M. Smaine, "Edge Detection in Range Images Using a Modified Canny Filter," Proc. of the IEEE International Conference on Theoretical and Applicative Aspects of Computer Science (ICTAACS), vol. 1, pp. 1-7, Skikda, Algeria, 2019.

[27] J. Canny, "A Computational Approach to Edge Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-8, no. 6, pp. 679-698, 1986.

[28] J. P. D'Haeyer, "Gaussian Filtering of Images: A Regularization Approach," Signal Processing, vol. 18, no. 2, pp. 169-181, 1989.

[29] S. Bhardwaj and A. Mittal, "A Survey on Various Edge Detector Techniques," Procedia-Technology, vol. 4, pp. 220-226, 2012.

[30] D. J. Olive, "Multiple Linear Regression," Linear Regression Book, Springer, Cham, pp. 17-83, 2017.

[31] S. Mazouzi and Z. Guessoum, "A Fast and Fully Distributed Method for Region-based Image Segmentation," Journal of Real-time Image Processing, vol. 18, no. 3, pp. 793-806, 2021.

[32] A. Richtsfeld, T. Morwald, J. Prankl, M. Zillich and M. Vincze, "Segmentation of Unknown Objects in

Indoor Environments," Proc. of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4791–4796, Vilamoura-Algarve, Portugal, 2012.

[33] L. R. Dice, "Measures of the Amount of Ecologic Association between Species," Ecology, vol. 26, no. 3, pp. 297-302, 1945.

ملخص البحث:

تبقى تجزئة الصّور من أهم المهام، المتعلقة بتحليل الصّور وفهمها. وهي تتعامل مع الصّور الخام من أجل إعدادها لتكون قابلة للاستخدام في عمليّات أوتوماتيكية عالية المستوى؛ مثل التّصنيف أو استرجاع المعلومات.

نُقدّم في هذه الورقة كاشف حواف جديداً معدَّلاً للصّور ذات المدى. وقد استلهمنا مبدأه من كاشف (كاني)، بحيث يتمّ أخذ السِّمات المتأصّلة في الصّور ذات المدى بعين الاعتبار. وفي العادة، يُستخدم كاشف (كاني) في الصّور ذات التدرّج الرّمادي أو الصّور الملوّنة، حيث يؤدي تطبيقه المباشر مع الأعماق الى عدم الحصول على نتائج مُرضية.

من الصّورة الخام، المحتوية على الأعماق المقاسة، يتمّ حساب صّورة تخفيف تتكون من صورة للمتّجهات العمودية على السّطوح المحليّة. وهكذا يجري استخدام الزوايا بين المتجهات المتجاورة لحساب درجة ميْل بناءً على الزّوايا. وقد استخدمت صّور حقيقية من مجموعة بيانات (ABW) في عملية التجريب، إذ تفوّق الكاشف المقترح على كاشف (كاني) الأصلي بنسبة (18%).