

Research Article

# Application of Gradient-based Edge Detection on Defined Surface Feature Boundary

Ainaa Farhanah Mohd Razali\* and Mohd Fauzi Ismail

Faculty of Mechanical Engineering, Universiti Teknologi MARA, Malaysia

[ainaa.mohdrazali@gmail.com](mailto:ainaa.mohdrazali@gmail.com); [mohdfauzi305@uitm.edu.my](mailto:mohdfauzi305@uitm.edu.my)

\*Correspondence: [ainaa.mohdrazali@gmail.com](mailto:ainaa.mohdrazali@gmail.com)

Received: 17<sup>th</sup> October 2020; Accepted: 15<sup>th</sup> November 2020; Published: 20<sup>th</sup> March 2020

**Abstract:** In surface topography, the detection of edges or surface feature boundary should be made prior to the characterization process to ensure precise measurement of the surface feature. However, problem arises when the existing edge detection from the image processing area is being directly adopted on the surface without firstly defining the exact location of edges. Thus, an approach has been carried out with the aim of developing the definition of edges to identify its exact location. The process of developing the definition of edges is conducted based on theoretical understanding and demonstration on the simulated surface topography data with the help of MATLAB software. Gradient-based edge detection operators are applied to see their performance on the defined surface feature boundary. Overall findings of this study showed that there are slight discrepancies of the edge position signified by the gradient-based operators to manually-defined edges. Despite of such outcome, it is then recommended to use either the Sobel operator or the Prewitt operator as a tool to signify the edge before proceeding with characterization process in future study due to their stability and clarity in producing the surface gradient function.

**Keywords:** *Edge detection; Feature boundary; Structured surface; Surface topography*

---

## 1. Introduction

Nowadays, the application of surface metrology has brought a remarkable significance in the industry sectors, especially in manufacturing industry. It is important to produce a better product performance because it is known that surface measurement can control the manufacturing process and functional performance of a component. According to X. Jiang et al. [1] from his research study, surface metrology is defined as the study of measuring small-scale structural features on surfaces which is the topography of the surface that has historically been known as significant due to its sturdy connection with other disciplines, including machining control and manufacturing processes, component quality, tribology, and surface engineering. The ease of providing surface metrology as one of the measuring systems was certainly seen as the best method to recognize how special surface topography is and how it plays an important role in a system especially in terms of component performance. Further down on surface metrology, a surface can be divided into two groups which are stochastic surface and structured surface. Despite having two groups of surfaces, structured surface has been widely used in industry for a variety of application due to its characteristic that has a dominant deterministic structure design with a high respect ratio pattern able to produce a specific function [2,3,4]. The study of such surface is therefore necessary since surface quality is able to greatly affects the suitability of components for specific applications [5].

Ideally, in surface topography, the detection of the edges should be developed prior to the characterization process to ensure precise feature measurement. When object or surface is being

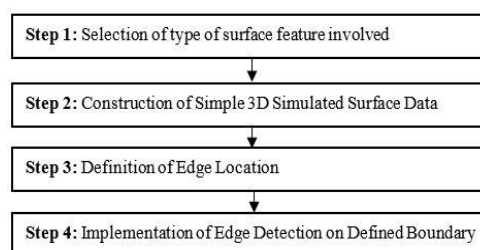
observed, the clearest part that can be seen firstly is usually edges and lines. Essentially, edges can allocate the boundaries between regions in a surface which helps in extracting features as they can provide useful information [6]. Throughout the years, several researchers have proposed many approaches in identification and extraction of the edges in structured surfaces by developing edge detection algorithms adopted from image processing in their research [7,8,9]. However, those approaches may not be suitable to be used directly in the surface metrology field since this field requires structured surface to be measured, visualized, and quantified of their feature characteristics on the surface itself. In contrast to the field of surface metrology, image processing tools provide the position of feature boundary or edges based on discontinuities or major variations on a grey level of image only [10], and not emphasizing on the importance of vertical or height positioning.

When the method of image processing is adopted directly to the surface metrology area without firstly defining the boundary, the problem will arise since the exact location of the boundary is unknown. This statement can be proven by referring to a study conducted by Nicola Senin et al. [11] on a method of extracting individual surface features from areal surface topography maps, where there are errors presence on the surface resulted from the inaccurate identification of boundary features and the presence of unwanted regions located on features walls. It is very important to clearly identify the surface feature interior so that it can be separated from the remaining topography or background to avoid the less significant feature from dominating the surface which can lead to over-segmentation. Hence, with the presence of error on the surface, the accuracy of edge lines and appropriateness of edge detection procedure are questionable.

An approach is introduced in this paper with the aim of developing the definition of edges to identify its exact location on a structured surface. This approach is simply based on the method of manually identifying the location of edges by using the theoretical concept of changes in slope, changes in vector direction, and edge detection operators as references to produce a surface feature boundary. Therefore, the goal of this paper is to develop the definition of edges by manually identifying its exact location on a surface referring to the theoretical concept before applying the gradient-based edge detection on the defined surface feature boundary and producing the result of a clear and thick edge. Defined surface feature boundary is a boundary of feature on a surface that can clearly be distinguished by human interpretation. On top of that, this paper focuses only on defining the edges on simulated surface topography data rather than on the surface of the existing specimen so that the edge detection theory can be carefully observed whether it is applicable or not.

## 2. Methodology

This section consists of four sequential steps involved in the process of conducting observation on the proposed approach. It began with selecting the type of surface feature, followed by the process of describing the data structure, finding the exact position of pixels that can be defined as an edge, and lastly implementing the edge detection on the defined surface feature boundary.



**Figure 1.** Procedure of new approach based on defining the edge on the structured surface

### 2.1. Preparation of Data Structure

Preparation of surface feature is the first step involved in this study. A proper description of the surface feature is necessary to ensure its extraction. In this study, the type of surface feature involved was the structured surface with areal surface features that produces topographical surface and represents the data as height function  $z(x,y)$ . A construction of a dataset containing topography

information of variable height values referring to z-axis with random noise arranged along the rows and columns of the x-axis and y-axis is capable of creating a simulated surface topography data. The topography data was plotted on a three-dimensional (3D) cartesian coordinate system with 0 origin and a pixel scale of a constant data interval of  $1\mu\text{m } \Delta x$  and  $1\mu\text{m } \Delta y$  as shown in Fig. 2. The colors in the surface topographical map signify the height of the surface feature.

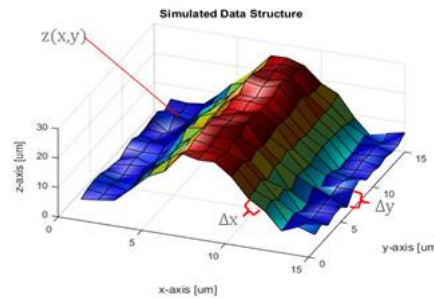


Figure 2. Simple 3D simulated surface topography

## 2.2. Definition of Edge Location

At this point, the logical edge location is manually identified. Theoretically, a feature is defined as a region that consists of boundary and interior where feature boundary has a constant slope while the feature interior is located above certain slope [12]. As shown in Fig. 3, using human interpretation, the dots indicate the boundary location where the maximum curvature is present while the thicker black lines are an edge bounded by two edge points. However, it is not convincing for us to just say that the thick lines are the logical edges location because the thick lines are surrounded by many pixels. The accuracy of that line itself is questionable because the exact location of edge data points is unknown; whether it is located at the first pixel, the second pixel or even the third pixel. In order to extract the feature boundary or edge lines, exact pixel locations of the most significant features of the surface need to be determined. Some parameters that can be considered when defining the significant edges are changes in slope and changes in vector direction.



Figure 3. Feature with a compound edge surrounded by background

Before going deeper into the definition of the slope, a gradient is a mathematical measurement used to describe how steep or gradual the slope is. It measures how much elevation changes as you move horizontally over a surface and it is described as rise over run. On the other hand, the slope can be described as the steepness of the ramps or stairs by considering the horizontal and vertical movement along them. There is a whole lot of fancy notation that goes along with it, but to make it easier, slope is referred to as how steep the line is. Towards detecting and tracing the feature boundaries, the calculation of slope acts as the first step to achieve the objective. Basically, it measures the slope by calculating the ratio of the vertical change (y-axis) and horizontal change (x-axis) between two points on a line.

$$\text{slope, } m = \frac{\Delta y}{\Delta x} = \frac{\text{rise}}{\text{run}} = \frac{y_2 - y_1}{x_2 - x_1} \quad (1)$$

In three-dimensional approach, the shape of the surface is generated with the function of  $z=f(x,y)$ . Since the equation has one dependent variable on the left and two independent variables on the right, it has a vertical axis  $z$  and two horizontal axes,  $x$  and  $y$ . This point would therefore have slopes in the direction of  $x$  and  $y$ . To define these slopes, partial derivatives will be used where the computation of derivative in  $x$ -direction and  $y$ -direction will take place. Recalling the  $f(x,y)$  function, its gradient is a vector containing components given by partial derivatives of  $f$ . Conceptually, from the definition of  $f$ , the components concerned are  $f_x$  and  $f_y$ , where  $f_x(x_0,y_0)$  is the partial derivative of  $f$  in relation to  $x$  while  $f_y(x_0,y_0)$  is the partial derivative of  $f$  in relation to  $y$ . Both

partial derivatives are the rate of change of  $z=f(x,y)$ . The gradient can be seen as always perpendicular to the surface and the slope is defined to be the angle between the gradient, that is parallel to and in the direction of the unit vector  $(x,y,z) = (0,0,1)$ . Since this study focuses on surface structure, the usual definition of the slope must be reversed where it equals the ratio of the run to the rise for the normal vector. Because there is a unit rise, the slope can be read as:

$$\text{slope}, m = \sqrt{f_x^2 + f_y^2} \quad (2)$$

Other than slope calculation, changes in vector direction can also be considered in order to define edges. The integral of a vector field flux through a surface requires a direction that will define an inside and outside relative against a surface that is the boundary between two different regions such as the feature and its surrounding. Hence, to see how much of the vector field is escaping from the feature interior to the background area, the vector field can be analyzed into two orthogonal components which are part normal to the surface and part tangent to the surface [13].

### 2.3. Implementation of Gradient-based Edge Detection

Edges consist of meaningful features and contain a significant information. To differentiate between significant and insignificant information, help from edge detectors is very important as it can aid in reducing and filtering out the features that may be regarded as less relevant [14]. In edge detection approach, the method of edge detection is classified into two groups which are gradient-based and Laplacian-based. However, this paper will focus more on gradient-based that comprises of classical operators such as Sobel, Prewitt, and Roberts. These gradient-based edge detectors identify the edges by searching for the maximum and minimum intensity values in the first-order derivative, and evaluating the distribution of intensity values in a given pixel setting [15]. The key to determine the boundary is then to determine such critical slope value or threshold.

Since edge detection approach takes the idea from the image processing method, the edge part is usually being declared when the gray value of intensity rapidly changes. The basic formula for all the first order derivative or gradient edge detectors is shown below.

$$\nabla f = G[f(x, y)] = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (3)$$

Note that  $\nabla f$  is the gradient vector points towards the maximum rate of change of  $f(x,y)$ . The magnitude of that vector is an important parameter in edge detection. Gradient magnitude is shown with:

$$|G| = \sqrt{G_x^2 + G_y^2} \approx |G_x| + |G_y| \quad (4)$$

The magnitude defines the strength of the edge on certain location. To determine its direction, the gradient direction is then given as:

$$\theta = \tan^{-1} \left( \frac{G_y}{G_x} \right) \quad (5)$$

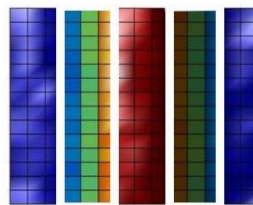
Theoretically, along with having to work with the basic formula mentioned above, the gradient-based edge detectors such as Sobel, Prewitt, and Roberts also transform the input image with their respective convolution masks to produce the gradient [16,17] as summarized in Table 1. Such convolutions masks are designed to respond to the edges that run vertically and horizontally relative to the pixel grid at maximum. The angle of orientation for both Sobel and Prewitt used the same basic formula with 3X3 convolution masks. However, there is a slight difference for Roberts operator as shown below since it uses convolution masks of 2X2. Despite having a small size of convolution masks that is able to make the edge detection process very fast, this will also create weaknesses such as Roberts will respond weakly to the edges, unless the edges difference is very sharp [18]. For the results, gradient-based edge detectors such as Sobel and Prewitt are expected to produce clearer edges.

**Table 1.** Convolution mask and the angle of orientation involved

Operator	Convolution Mask		Angle of orientation
	$G_x$	$G_y$	
Sobel	$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$	$\theta = \tan^{-1} \left( \frac{G_y}{G_x} \right)$
Prewitt	$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$	$\theta = \tan^{-1} \left( \frac{G_y}{G_x} \right)$
Roberts	$\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$	$\theta = \tan^{-1} \left( \frac{G_y}{G_x} \right) - \frac{3\pi}{4}$ (6)

### 3. Results and Discussion

This section discusses the results into two parts which are the result of definition of edge location and the result of implementation of edge detection operators on the defined surface feature boundary. For the first part, when simple 3D surface topography has been simulated, the data are analysed to manually define the edge location. The need in defining the edge location first is necessary in order to overcome the problem that may occur if the image processing method is being applied directly on the surface. Figure 4 shows the top view of the logical edge location based on human interpretation. The white lines indicate the feature boundary that separates the feature interior area from its surrounding. To ensure the truth of the statement, the process of manually defining the edge location is conducted based on the theoretical concept of change in slope and change in vector direction.

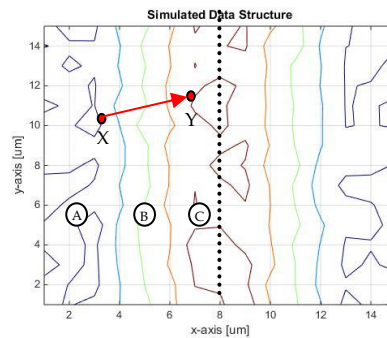


**Figure 4.** Logical edge location based on human interpretation

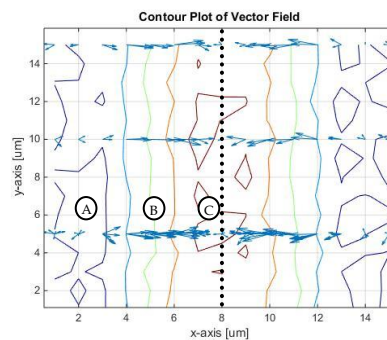
To make it easier in developing the definition of edges on the surface, the simulated 3D surface topography data is being viewed in contour plot as shown in Fig. 5 so that the location of a steep and gradual slope between the contour lines can be clearly seen. The colour of the contour lines indicates the surface height of the topographical map while the dotted line indicates the centre of the surface. In order to find out where the slope is very steep, an area with a big change of location is determined based on observation at which area the contour lines is close to each other since the closer the lines, the steeper the slope will be. In this case, it is to compare the areas A and C which have loosely packed lines that are inconsistently spread out, while the parallel lines that are close to each other are definitely in area B. Hence, the area A and C had a flatter slope. Referring to the three lines on the left side of the topographic map, the orange coloured line is at the highest height compared to the other two lines with green and light blue colours. To calculate the gradient of a slope between point X and point Y on the topographic map, the inclination changes and distance between those two points are observed carefully. In MATLAB, the gradient is automatically calculated using gradient function. An early overview of the development of the surface feature boundary can be seen through the process of identifying the location of slope by observing the contour plot. As a result, it can be seen that the place where the slope change takes place proves that the location of the edge identified through human interpretation earlier is the same.

To ensure it is clearly understood, an observation on vector field direction can also be made. Figure 6 shows a contour plot of the vector field where it helps in analysing whether the slope location mentioned in Figure 5 is at the right location or not. From previous discussion, area B has been defined as where the location of edges might be. By analysing Figure 6, it can be seen that the vector field is perceived as thickening as it directs from area B towards the centre or highest point of the surface topographical map in area C. Therefore, this proves that the concept of surface feature boundary corresponds to what has been theoretically discussed as shown in Fig. 3. As already said, an approach with the aim of developing the definition of edges to identify its exact location manually is carried out in order to avoid those edges from being obtained accidentally. Therefore,

this paper focuses solely on manually identifying the edges based on theoretical concepts mentioned in section 2.2.

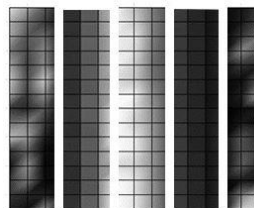


**Figure 5.** Contour plot of 3D simulated topography data



**Figure 6.** Contour plot of the vector field

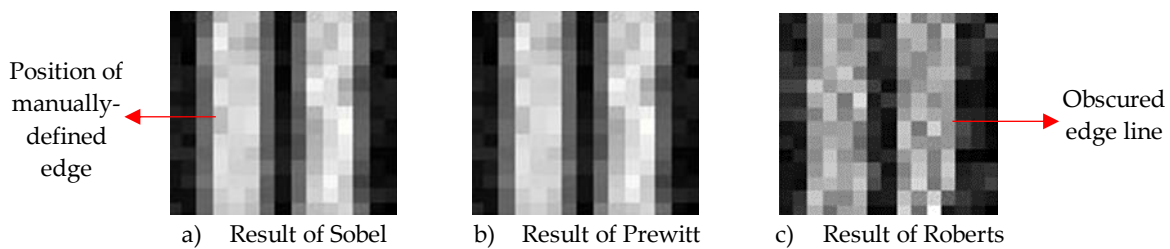
The second part of this section discusses on the implementation of gradient-based edge detection on defined surface feature boundary. This part describes how the step of implementing edge detection operators on the defined surface feature boundary is performed in MATLAB software. MATLAB software is chosen due to its ability to plot functions and data. After the preparation of simple 3D simulated surface topography data has been completed and the edge has been manually identified, the next step is to implement the gradient-based edge detection operators on the defined boundary. The data had gone through filtering and levelling processes using High Pass Gaussian Filter method where both processes were run simultaneously to remove an outlier presence on the surface and at the same time level the surface, producing an output for scale limited surface. Fig. 7 depicts the result of the defined surface feature boundary that has been manually identified, filtered, and levelled. Meanwhile, the white lines indicated the surface feature boundary.



**Figure 7.** Manually-defined surface feature boundary

After that, the task is proceeded with the implementation of gradient-based edge detection such as Sobel, Prewitt, and Roberts operators. The purpose of conducting this implementation is to investigate how the defined surface feature boundary is affected when the gradient-based edge detection is being implemented on it. Fig. 8 displays the results of the three gradient-based edge detection operators when dealing with the defined edge on the simulated 3D surface topography data. Traditionally, the gradient-based edge detection is done by finding the highest values or intensities in the gradient function as indicate by bright area in Fig. 8. On the other hand, as theoretically stated in section 2.3, the edge part is usually being declared when the brightness intensity of gray value rapidly changes. Hence, further analysis on the gradient function may enable the detection of the edges that comparable to the manually-defined edges of the original surface topography.

All edge detection operators have the same function which is to find the boundaries of objects by separating feature interior from its background to differentiate between significant and insignificant information. However, the results of performance for each operator are not really the same after being applied on the defined surface feature boundary. As seen in Fig. 8(a) and Figure 8(b), Sobel and Prewitt operators have clearer result of feature boundary compared to Roberts operator. Roberts cross kernels containing a set of 2X2 convolution masks are relatively small to compare with 3X3 convolution masks used by both Sobel and Prewitt operators. Therefore, based on the theoretical background of edge detectors, it can be proven that by using 3X3 convolution masks, Sobel and Prewitt operators are superior because of their ability to respond to the edges running vertically and horizontally at maximum performance and producing clear separation of feature boundary between significant and insignificant regions. Even though the edge detection process using Roberts operator is faster because of its small convolution masks size, it still has weaknesses where Roberts operator can result in highly susceptible to noise in the detection of the edges and their orientation, as well as producing very weak responses to genuine edges unless the presence of edges are very sharp. The results of the process when using Roberts operator can be seen in Fig. 8(c) where the magnitude of the edges is ultimately degraded. The uneven feature boundary line has caused the separation of feature interior and its background to be obscured. Hence, it can be concluded that the performance of Roberts operator is not up to the mark compared to the other two operators when it is being applied on a defined surface feature boundary.



**Figure 8.** The gradient function of the simulated surface topography by Sobel, Prewitt and Roberts

#### 4. Conclusion

To sum up this paper, an approach to develop the definition of the edges by manually identifying its location on the structured surface has been successfully carried out. The results show that the edges or feature boundary could be defined at the location where pixels have maximum brightness value representing the high gradient value where the position of the slope is very steep. Hence, proving the theoretical concept of changes in slope and changes in vector direction is acceptable to be referred. To compare with Roberts which has lots of noise detected on the pixels, the detection of feature boundary is clearer when Sobel and Prewitt are being used. The edge signified by the gradient-based operators are deviated from the manually-defined edges of the original surface data, but the gradient function surface has a characteristic that shows the possibility of positive result in the future study.

#### Acknowledgement

The author of this research would like to acknowledge Universiti Teknologi MARA and Ministry of Higher Education Malaysia for financial support given as this work is fully supported under the Fundamental Research Grant Scheme (FRGS) with a Grant No: FRGS/1/2018/TK03/UITM/02/03.

#### References

- [1] X. Jiang, P. Scott, and L. Blunt. (2007). Paradigm shifts in surface metrology. Part I. Historical philosophy. *Proceedings The Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 463, pp. 2049–2070.
- [2] G. D. Macaulay, N. Senin, and C. L. Giusca. (2014). Comparison of segmentation techniques to determine

- the geometric parameters of structured surfaces. *Surface Topography: Metrology and Properties*, vol. 2, pp. 1–12.
- [3] G. D. Macaulay, N. Senin, C. L. Giusca, and R. K. Leach. (2015). Review of feature boundary identification techniques for the characterisation of tessellated surfaces. *Surface Topography: Metrology and Properties*, vol. 3, no. 1, pp. 1–21.
  - [4] J. Blanc, D. Grime, and F. Blateyron. (2011). Surface characterization based upon significant topographic features. *Journal of Physics: Conference Series*, vol. 311, no. 1, pp. 1–6.
  - [5] M. F. Ismail, T. R. Jaafar, N. Che Pin, and N. H. Zaini. (2015). Sobel operator for edges detection in surface texture analysis. *Jurnal Teknologi*, no. July 2016, pp. 71–74.
  - [6] E. J. Mohammad, M. Jawadkadhim, W. I. Hamad, and S. Y. Helyel. (2014). Study sobel edge detection effect on the image edges using MATLAB. *International Journal of Innovative Research in Science, Engineering and Technology*, vol. 3, no. 3, pp. 10408–10415.
  - [7] N. Senthilkumaran and R. Rajesh. (2007). Edge detection techniques for image segmentation – A survey of soft computing approaches. *International Journal of Recent Trends in Engineering*, vol. 1, pp. 250–254.
  - [8] M. Roushdy. (2006). Comparative study of edge detection algorithms applying on the grayscale noisy image using morphological filter. *GVIP Journal*, vol. 6.
  - [9] M. Sharifi, M. Fathy, and M. T. Mahmoudi. (2002). A classified and comparative study of edge detection algorithms. *Proceedings of International Conference on Information Technology: Coding and Computing*, pp. 117–120.
  - [10] Z. Jin-yu, C. Yan, and H. Xian-xiang. (2009). Edge detection of images based on improved sobel operator and genetic algorithms. *International Conference on Image Analysis and Signal Processing*, no. 3, pp. 31–35.
  - [11] N. Senin, L. A. Blunt, R. K. Leach, and S. Pini. (2013). Morphologic segmentation algorithms for extracting individual surface features from areal surface topography maps. *Surface Topography: Metrology and Properties*, vol. 015005, pp. 1–11.
  - [12] Y. Song and J. Shan. (2009). An adaptive approach to topographic feature extraction from Digital Terrain Models. *Photogrammetric Engineering and Remote Sensing*, vol. 75, March, pp. 281–290.
  - [13] J. Ma, X. Lu, G. Li, Z. Qu, and F. Qin. (2020). Toolpath topology design based on vector field of tool feeding direction in sub-regional processing for complex curved surface. *Journal of Manufacturing Processes*, vol. 52, pp. 44–57.
  - [14] S. Ghodrati, M. Mohseni, and S. G. Kandi. (2019). Application of image edge detection methods for precise estimation of the standard surface roughness parameters: Polypropylene / ethylene- propylene-diene-monomer blend as a case study. *Measurement*, vol. 138, pp. 80–90.
  - [15] Dharampal and V. Mutneja. (2015). Methods of Image Edge Detection : A Review. *Journal of Electrical and Electronic Systems*, vol. 4, no. 2, pp. 1–5.
  - [16] J. Kaur. (2012). A comparative analysis of thresholding and edge detection segmentation techniques. *International Journal of Computer Application*, vol. 39, no. February, pp. 29–34.
  - [17] Muthukrishnan.R and M.Radha. (2011). Edge detection techniques for image segmentation. *International Journal of Computer Science and Inforation Technology*, vol. 3, no. 6, pp. 259–267.
  - [18] E. P. Wibowo, M. Mustikasari, J. Harlan, S. Widiyanto, and D. Sundani. (2018) Panoramic X-Ray Analysis Using Edge Detection Method on Mandibular Cortex. 2018 2nd East Indonesia Conference on Computer and Information Technology (EIconCIT). pp. 245–249.



© 2021 by the author(s). Published by Annals of Emerging Technologies in Computing (AETiC), under the terms and conditions of the Creative Commons Attribution (CC BY) license which can be accessed at <http://creativecommons.org/licenses/by/4.0>.