Edge Detection of Digital Image with Different Edge Types

M F V Ruslau, R A Pratama, E Meirista

Department of Mathematics Education, Faculty of Teacher Training and Education, Universitas Musamus, Merauke-Papua, Indonesia

Email: maria_fkip@unmus.ac.id

Abstract. In digital images processing, there are three types of edges based on intensity changes. Namely, step edges, ramp edges and edges noise. An edge is defined as a set of pixels where there is an abrupt change in colour intensity over distance. On-ramp edges where gray levels change slowly, the Gradient Method is able to detect better. On step edges where the intensity or gray levels changes very quickly the Laplace method is able to detect better than the Gradient Method. In this study, three images were used as samples and identified the type of edge of each image. Furthermore, edge detection is performed with the first derivative operator Canny and the second derivative operator Laplacian of Gaussian. The results indicate that for step edges LoG provides better results, whereas for ramp edges Canny detects better. However, by selecting the right threshold that matches the \( \sigma \) (standard deviation), Canny is also capable to provide good edge detection results. The greater the \( \sigma \) value, the threshold was chosen must be small so that the results obtained are good and easily interpreted. The Canny operator produces a thinner edge and a firmer boundary between objects and between objects on the given sigma = 1 value while the LoG operator corrects better, especially on the steep part of the value \( \sigma = 2 \) compared to the value \( \sigma = 1 \).

1. Introduction

The problem of calculating the derivative of functions appears quite a lot in the field of engineering. For example in image processing. In image processing, derivatives are applied to detect the edges of objects in an image [1,2]. "A picture is more than a thousand words" means that an image can provide more information than that information presented in a text [3].

An image is a collection of image elements (pixels) which a whole record a scene through visual sensing, which can then and then presented in a discrete form for the benefit of processing with computers [4]. One technique for obtaining information from images is edge detection. Edge is defined as a change in a large intensity in a short distance, which shows the boundary between two objects in one image. Based on changes in intensity there are three types of edges, namely, step edges, ramp edges or widths edge and edges noise. Mathematically, changes in large intensity over a short distance can be seen as a function that has a large slope. The slope measurement of a function is done by calculating the first derivative. Changes will reach a maximum when the first derivative value reaches the maximum or the second derivative value (2nd derivative) is 0 [5]. The change in intensity in the image is associated with a discontinuity called edge. Intensity changes occur quickly, suddenly and far from one gray value to another gray value. It can be seen as a large slope function. In practice, the imperfection of the image retrieval process will produce blurred edges. Blurred edges tend to be thicker, and sharp edges tend to be thinner. Through a visible observation, it can be seen whether the edges produced by a particular method have good quality or not [6]. Therefore, it is necessary to have...
the right knowledge to determine the edge detection operator in accordance with the condition of an image to be detected. The expectation is the image segmentation process can provide a shape that is close to the actual shape. Segmentation is one process in image processing whose purpose is to divide the image into several objects or several regions [7].

Based on [8,9] there are several ways to do edge detection but commonly used are the Gradient Method and Laplacian Method. The Gradient method is perfect for detecting ramp edges where pixels of gray level change very slowly. The Canny operator is known as one of the best first gradient edge detection operators compared to other first gradient detection operators. Using the Canny Operator, the edges appear clearer, the difference between the edges and the image background is obvious. For images where the gray level changes quickly from dark to bright, the gradient operation will produce a widening edge. To detect the edges of the image, the Laplacian Method can be used. The Laplacian method will detect zero crossings to determine the boundary between black and white which is in the 2nd derivative of the image. The zero-crossing property is quite useful for determining the center of a thick edge. The disadvantage of applying the Laplacian Method is very sensitive to noise. Therefore, it is often combined with a Gaussian function called Laplacian of Gaussian (LoG) and the results of edge detection with this method can be increased by applying thresholding [10].

2. Theoretical Review

2.1. Image
Mathematically, an image is a continuous function of the intensity of light in 2 dimensions \( f(x, y) \). The amplitude of \( f \) at any pair of coordinates \( (x, y) \) is called the intensity or gray level. The image is a light source illuminating the object and reflected back and captured by optical devices such as the human eye, camera, scanner, sensor, satellite and so on, which then the reflected reflection can be recorded. There are two types of images namely analog image and digital image. Analog images are images that are continuous and cannot be represented or processed on a computer directly whereas digital image is a numerical representation with discrete values that can be read and processed by a computer. Note that a digital image is composed of a finite number of elements, each of which has a particular location and value. These elements are referred to as picture or image elements or pixels [11].

A digital image is a matrix where the row and column indices represent a point in the image and the matrix elements (called pixel/picture element/pels) state the gray level at that point. The digital image can be presented by matrix \( f \), which measures \( N \times M \) with the shape as shown below:

\[
f(x, y) \approx \begin{bmatrix}
f_{(0,0)} & f_{(0,1)} & \cdots & f_{(0,M-1)} \\
f_{(1,0)} & f_{(1,1)} & \cdots & f_{(1,M-1)} \\
\vdots & \vdots & & \vdots \\
f_{(N-1,0)} & f_{(N-1,1)} & \cdots & f_{(N-1,M-1)}
\end{bmatrix}
\]

where: \( N = \) number of rows ; \( 0 \leq y \leq N - 1 \)

\( M = \) number of column ; \( 0 \leq x \leq M - 1 \)

There are four types of images, namely binary image, intensity image, RGB image, and indexed image. Binary imagery, that is, an image where all pixels in an image are worth 0 (zero) or 1 (one). Intensity image is an image where the gray intensity or value has a possible color between black (minimum) and white (maximum). The RGB image is an image where each point or pixel represents a color which is a combination of three basic colors namely red, green and blue. While the indexed image is an image where each point on the image represents the index of an available color table [12,13].

2.2. Digital Image Processing
Image processing is an activity to improve image quality so that it is easily interpreted by humans/machines (computers). Digital image processing is a process that aims to manipulate and
analyze images with the help of a computer. Digital images processing can be grouped into two types of activities, that is:

1. Improve the quality of an image, so that it can be more easily interpreted by the human eye. An example is a noise removal operation in Lena’s image.

2. Processing information on an image for the purpose of object recognition automatically. An example is the edge detection operation in the image of Cameraman.

2.3. Edge Detection

background part. Similarly, if two objects overlap each other, if their intensity is not the same, it will leave a trail of edges so that one object is known to be in front of the other object or otherwise. Edge Detection is the most commonly used approach to detect gray level discontinuities. Edge is defined as a change in the intensity of a large distance, with $\alpha$ being the direction of the edge and this direction can vary depending on the change in intensity [14]:

Based on changes in intensity, there are three types of edges that exist in digital images, including the following:

1. The edge is step, where the edge has a very sharp intensity change with the edge 90o. The steep edge in the form of a digital image format and graph can be seen in Figure below.
2. A ramp or wide edge, i.e. an edge with a small edge angle, sloping can also be interpreted as a number of adjacent edges. The ramp edge in the form of a graphic and digital image format can be seen in the Figure below.

![Figure 5](image)

**Figure 5.** Ramp Edge in Matrices and Graphs.

3. The edge that contains noise. Edge noise in digital image format does not have a certain pattern. In plain view and in the form of a graph the noise can be seen in Figure 6. Noise can be caused by physical (optical) interference in the acquisition and intentional devices due to improper processing. The intensity of noise is random, so its presence can bring up other edges around the actual edge, besides that it can also shift the actual edge position.

![Figure 6](image)

**Figure 6.** Noisy Edge in Matrices and Graphs.

Visually it can be clearly seen at the focus of the object whether the gray color difference is different from dark to bright or not. Mathematically we have not been able to set a boundary or benchmark for gray values when an edge can be said to be step or ramp or contains noise, we can use the distribution of the gray degree value of the image as a reference.

According to [15], a histogram is a diagram used to calculate the number of occurrences of color in an image, in this case, is a gray color (0 - 255). The histogram assumes that the image consists of regions with different intensities, then classifies the parts of the image in the form of histograms into several peaks where each peak is called a region.

Values along with the x-axis state the degree of gray and the value along the y-axis represent the number of pixels that have a certain gray value. The histogram peak shows the intensity of the protruding pixels. The width of the peak shows the contrast range of the image. The valley width between the peaks of the histogram represents the threshold. Because the image used in this study has a gray degree of 256, from 0 to 255, the histogram will state the number of occurrences of a particular value or the contrast range between 0 - 255.

2.4. *Canny Edge Detector*
In a digital image, the magnitude or value of the first derivative can be used to detect the presence of edges at a point in the image. This property is used as a basis for using gradient operators as edge detectors. There are a number of first gradient edge detection operators that can be used to detect edges in the image, namely centralized Difference Gradient Operators, Sobel Operators, Prewitt Operators, Roberts Operators, and Canny Operators. The Canny operator detects edges more optimally than other first gradient edge detection methods. The Canny operator removes noise using a Gaussian filter.

\[ G_n(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \]  

(2)

The Canny method does not have a special operator, but consists of several special steps as follows [15]:

1. Smooth the image or get rid of noise by converting the input image with a mask that follows the Gaussian function (equation (2), as follows:

\[ h(x, y) = \sum_x \sum_y h_y \]  

(3)

Where,

\[ h_y(x, y) = e^{-\frac{x^2 + y^2}{2\sigma^2}} \]  

(4)

so that the obtained image has been smoothed \( G(x, y) \).

2. Find image gradients by converting each point in the smoothed image \( G(x, y) \) with the first derivative mask against \( x \) and \( y \), respectively as follows

\[
\begin{align*}
P(x, y) &= G(x, y) \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix} \\
Q(x, y) &= G(x, y) \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}
\end{align*}
\]

(5)  

(6)

3. Calculate the gradient magnitude and the gradient direction, where in the form of the magnitude gradient matrix \( M(x, y) \) and the gradient direction \( \theta(x, y) \) respectively as follows:

\[
\begin{align*}
M(x, y) &= \sqrt{P(x, y)^2 + Q(x, y)^2} \\
\theta(x, y) &= \tan^{-1}\left(\frac{Q(x, y)}{P(x, y)}\right)
\end{align*}
\]

(7)  

(8)

4. Applying maximum non-suppression, overriding pixels that have zero values that have nothing to do with edges.

![Figure 7. Gradient Direction Diagram in the Canny method](image)

The numbers outside the circle state the size of the angle and the numbers in the circle represent the angle code. The angles given code 0 include the angle 00, the angles coded 1 include the angle 450, the angles coded 2 include the angle 900 and the angles coded 3 include the angle 1350.

5. Detect edges with thresholding.
2.5. Laplacian of Gaussian Edge Detector

According to [2,14], in addition to the first gradient method, edges can also be determined using zero-crossing properties (imaginary straight lines connecting the positive and negative extreme values of the 2nd derivative will pass zero in the middle edge) of the second derivative Laplace. Compared to the first gradient method, the Laplacian method produces a fairly thin edge, while the Laplacian weakness is very sensitive to noise.

Laplacian of the 2-D function $f(x,y)$ is the second derivative defined as follows:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$  \hspace{1cm} (9)

This Laplace second derivative can be written in the form of an operator. The form of this operator in image processing can be written in neighboring coordinates from 8 adjacent points as shown in Figure below:

![Figure 8. 8 Point Neighbor Coordinate and Laplace Operator.](image)

Laplacian is generally not used in its original form for edge detection because it is very sensitive to noise, the magnitude of Laplacian produces multiple edges and cannot detect the direction of the edges. Therefore, often to find edges, Laplacian is combined with a Gaussian function. LoG function (Laplacian of Gaussian) as an operator or filter as follows:

$$\text{LoG} \triangleq G_\sigma(x,y) = \frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} e^{-\frac{(x^2 + y^2)}{2\sigma^2}}$$  \hspace{1cm} (10)

The steps for edge detection using LoG Operators are as follows:

1. Convolve the input image with the LoG filter, which is calculated by the formula below:

$$h(x,y) = \frac{\left(x^2 + y^2 - 2\sigma^2\right)h_t(x,y)}{2\pi\sigma^2 \sum_j h_t}$$  \hspace{1cm} (11)

where $h_t$ is obtained from the formula (5)

2. Determine zero-crossing by selecting a threshold.

3. Do thresholding to convert the output image into a binary image.

2.6. Thresholding

One way to tap objects from the background is to choose a threshold $T$. The thresholding technique can be put into three different types based on function $T$ and its associated parameters as given by $T = T[x,y,p(x,y),f(x,y)]$ where $f(x,y)$ is the gray level of the point $(x,y)$ and $p(x,y)$ states several local properties of that point (e.g the gray level average of the neighbor centered on $(x,y)$). Thresholding is the transformation of an input image $f(x,y)$ to an output (segmented) binary image $g(x,y)$ as follows [12,16]:

$$g(x,y) = \begin{cases} 1, & \text{if } f(x,y) \geq T \\ 0, & \text{if } f(x,y) < T \end{cases}$$  \hspace{1cm} (12)

pixels labeled 1 refer to objects while those labeled 0 are background objects.

2.7. Histograms
In image processing, histograms are used to analyze images. We can estimate the properties of an image just by looking at the details on the histogram. An image histogram is a type of histogram that acts as a graphical representation of the tonal distribution in a digital image. A grey level histogram indicates how many pixels of an image share the same grey level. The x-axis shows the grey levels (e.g. from 0 to 255), the y-axis shows their frequency in the image. This information can be used to calculate a threshold [17,18].

3. Method
Image samples used are images that have different levels of intensity change (images with steep edges, ramps, and images with edges containing noise). The images used in this study were taken intentionally and included in MATLAB. Then the test images are converted into intensity images or gray images and gray values are displayed in the histogram to determine the edge type of the images.

3.1. Edge Detection Algorithm
1. Downloading image. At this stage what is done is to read the image with the imread('image') command or from the file menu, import data.
2. Convert images into a grayscale image with rgb2gray(image) command.
3. Displays the histogram of image intensity with imhist(image) command.
4. Identify the type of image edge.
5. Intensification: Apply differentiation to enhance the quality of edges.
   - Detect edges using edge(Image,'canny') commands or edge(Image,'canny',thresh, sigma).
   - Detect edges using edge(Image,'log') commands or edge(Image,'log',thresh, sigma).
6. Threshold: Edge magnitude threshold is used to reject the noisy edge pixels and others should be confined.
7. Displaying the Image. Display the output image with the imshow(image) command. The image results from edge detection, then it will be interpreted for the sake of further analysis.

3.2. Edge Detection Algorithm
An edge detection method can be said to detect the most optimal edge if it meets the criteria, as follows:
- Clear response. There is only one response for each edge, so it is easily detected and does not cause confusion in subsequent image processing.
- Another test for edge detection is to use resistance to noise (noise). Interference in the input image can be used as a parameter that determines the level of appearance of several methods to track the edges of an object.
- Other parameters are by looking at the strength, smoothness, and thickness of the edges of the resulting image output, which can clearly distinguish objects from the background.

4. Finding and Result
Image as an object in this study is an RGB image. The experiment is carried out by generating an image and then converted to an image intensity (grayscale) with 256 gray level. Image intensity is a grayscale image that has the possibility of a color between black (minimum) and white (maximum) with intensity between 0 to 255. The experimental results are presented as follows:
To determine the edge type of a sample image, a grayscale image is presented in the histogram that contains the intensity distribution of an image. It is a plot with pixel values (ranging from 0 to 255) in x-axis and a corresponding number of pixels in the image on the y-axis.

In Figures 10 (a) and 10 (b), it can be seen that there are three histogram peaks where the difference in the number of pixels between levels of the histogram is very large. There is a very deep and wide valley between the three peaks which causes the histogram peak distance far enough and states that there is a very sharp change in the gray value composition or said the tree image and twilight image has a step edge. Figure 10 (c) shows that there are two peaks and two hills separated by a valley that is quite steep, but narrow and the two hills are quite wide. This shows that the distance or the difference in the number of pixels between the levels of the histogram is small so that it can be said that the change in the composition of gray values in rice image is not sharp or it has a ramp edge. The fact is that the image of nasi.jpg contains two types of edges, namely step edges in a small portion of the image (a pile of rice on the left side of the image and background) and a ramp edges on the right side of the image. Because the sloping part is more dominant than the steep part, the rice image is classified as an image with a ramp edge.
4.1. Edge Detection with Canny Edge Detector

The sample image is then detected using the Canny operator with \( \sigma = 1 \) (MATLAB default). The results are presented as follows:

![Image with step edge type](image1.png)
![Image with step edge type](image2.png)
![Image with ramp edge type](image3.png)

**Figure 11.** Edge detection results with Canny Operator (a) Image with step edge type (b) Image with step edge type, and (c) Image with ramp edge type.

The results of edge detection in images with steep edges (11 (a) and 11 (b)), indicate that edge detection using Canny Operators is quite good but still not optimal. This is evidenced by the presence of other edges beside the actual edge that Canny reads, which causes the image detection results to be less clean and clear. Likewise in the image with a sloping edge, the Canny Operator still cannot provide optimal results on steep parts.

To increase the results of edge detection, then the sample image will be detected by selecting several threshold values and different standard deviations (\( \sigma \)). Threshold values used are in the interval [0,1] because MATLAB only determines the threshold value in the interval [0,1]. The choice of threshold is done by trial and error to find out in what range the threshold results of edge detection in the two images will be close to the optimal results. In this study, the standard deviation (\( \sigma \)) used is \( \sigma = 1 \) and \( \sigma = 2 \).

![Threshold values](image4.png)

**Figure 12.** Optimum edge detection results using Canny Operators.

The results of edge detection using Canny Operators show images with curved edges (fig. 12(a)-12(h)), Canny can provide optimal results if the right threshold is used. The results agreed above are the results of estimates for the best threshold according to the size of \( \sigma \) used. When viewed carefully, it can be seen the results of edge detection in the image with a ramp edge, give the value of \( \sigma = 1 \) the edge result is thinner and the edge is more emphasized between objects and between objects and the rear screen. The results of edge detection show that for the tree image the results obtained will be close to optimal at the threshold range of 0.005 to 0.03 for \( \sigma = 1 \) and 0.009 to 0.2 for \( \sigma = 2 \). And for twilight
images, the results obtained are close to optimal in the threshold range 0.02 to 0.06 for \(\sigma=1\) and 0.009 to 0.03 for \(\sigma=2\).

The results of edge detection in rice images show that the greater the value of \(\sigma\) is used, the threshold was chosen must be small (Figure 12 (i) -12 (l)) so that the results obtained are good and easily interpreted further. Canny operators provide good edge detection results in threshold range between 0.19 to 0.25 for \(\sigma=1\) and between 0.009 to 0.02 for \(\sigma=2\). If observed carefully, for images with step edges, it can be seen that if the value of \(\sigma=1\) is used, then the resulting edge is thinner and gives a firmer edge between objects and between objects and background.

4.2. Edge Detection with LoG Edge Detector

The edges of the sample image are detected using the LoG operator with \(\sigma=2\) (default MATLAB). The edge detection results are presented as follows:

![Figure 13. Edge detection results with LoG Operator (a) Image with step edge type (b) Image with step edge type, and (c) Image with ramp edge type.](image)

From the results above, it can be seen that, in ramp edges, compared to the results of edge detection with canny operators, LoG operators provide better results. The resulting image is cleaner, the boundary between the object and the background (Figure 13 (a)) and, the boundary between one object and another with a clearer background (Figure 13 (b)). This is because at the point where the graph has a sharp angle, the vertical tangent or in the form of a jump (very bad sway) that we call the first step edge at that point does not exist. Whereas the edge will be found when the first derivative reaches the maximum. But we can look for zero crosses in the second derivative to find the edge of the image. LoG as a second derivative operator detects more accurately on step edges because when the second derivative is zero we can analyze how the tangent winding. So that LoG is able to detect edges precisely compared to Canny.

The results of edge detection using LoG with a threshold value and a different standard deviation (\(\sigma\)) and the Standard deviation (\(\sigma\)) used are \(\sigma = 1\) and \(\sigma = 2\), presented as follows:
The results of edge detection with LoG Operators show that for tree image the results obtained will be close to optimal in the threshold range of 0.029 to 0.05 for $\sigma = 1$ and 0.002 to 0.01 for $\sigma = 2$. And for twilight images, the results obtained are close to optimal in the threshold range of 0.028 to 0.03 for $\sigma = 1$ and 0.0015 to 0.003 for $\sigma = 2$. While the results of edge detection for rice images show that at the value of $\sigma = 2$, LoG corrects better, especially on steep parts than the value of $\sigma = 1$. LoG operators provide optimal results at the threshold range 0.09 to 0.1 for $\sigma = 1$ and 0.01 to 0.015 for $\sigma = 2$.

Based on the experiments carried out, it was found that if the threshold given below the range would result in an indecisive edge even very bad in the sense of being unable to separate objects from the background properly. Whereas if the given threshold is greater than the range, there will be an undetected or missing edge so that a lot of information is lost from the image.

5. Finding and Result

Canny corrects very sharply on the sloping edge compared to LoG. With thresholding, LoG gives good results on steep parts, but on the part of the object that tends to slope, LoG causes a lot of missing edges. As for the Canny, the difference in objects and background becomes very clear and inside the object of rice with the sloping edge, Canny can detect very well. In the image with the ramp edge, it shows that for $\sigma$ large, the result will be close to optimal if the selected threshold value is small, both by using the Canny and LoG operators.

Acknowledgments
We thank Musamus University for facilities, supports, and publications.

6. Reference

[1] Kaur S 2016 Comparison between Edge Detection Techniques 145 15–8
[2] Xu W, Li J and Jia H 2019 The Applications of the Edge Detection on Medical Diagnosis of Lungs The Applications of the Edge Detection on Medical Diagnosis of Lungs J. Phys. Conf. Ser. 1187
[3] Ushma A, Scholar M and Shanavas P A R M 2014 Object Detection In Image Processing Using Edge Detection Techniques 04 10–3
[4] Hassaballah M, Abdelmegeid A A and Alshazly H A 2016 Image Features Detection, Description and Matching Image Features Detection, Description
[5] Alazzawi A, Alsaadi H, Shallal A H and Albawi S 2015 Edge Detection-Application of (First and Second) Order Derivative in Image Processing Diyala J. Eng. Sci. 430–40
[6] R M and Radha M 2011 Edge Detection Techniques for Image Segmentation Int. J. Comput. Sci. Inf. Technol. 3 259–67
[7] Sivakumar P and Meenakshi S 2016 A review on image segmentation techniques Int. J. Adv. Res. Comput. Eng. Technol. 5 641–7
[8] Avlash M and Kaur L 2013 Performances Analysis of Different Edge Detection Methods on Road Images Int. J. Adv. Res. Eng. Appl. Sci. 2 27–38
[9] Jabbar H M A, Hatem A J and Ameer I M A A 2018 Practical Study for the Properties of Hueckel Edge Detection Operator J. Phys. Conf. Ser. 1003
[10] Narendra V G and Harees K S 2011 Study and comparison of various image edge detection techniques used in quality inspection and evaluation of agricultural and food products by computer vision Int. J. Agric. Biol. Eng. 4 83–90

[11] Abdalla S H and Osman S E F 2016 Digital Image Processing Technology based on MATLAB Int. J. Adv. Res. Comput. Sci. 7 216–21

[12] Dwipayana M, Arnia F and Musliyana Z 2017 Histogram equalization smoothing for determining threshold accuracy on ancient document image binarization Histogram equalization smoothing for determining threshold accuracy on ancient document image binarization J. Phys. Conf. Ser. 1019

[13] Deligiannidis L and Arabnia R H 2015 Emerging Trends in Image Processing, Computer Vision, and Pattern Recognition Emerging Trends in Image Processing, Computer Vision, and Pattern Recognition (Waltham, USA: Elsevier Inc)

[14] Syst J E E and Mutneja V 2015 Methods of Image Edge Detection : A Review J. Electr. Electron. Syst. 4

[15] Sharma P, Singh G and Kaur A 2013 Different Techniques Of Edge Detection In Digital Image Processing Int. J. Eng. Res. Appl. 3 458–61

[16] Sahu T P and Jain Y K 2015 Improved Simplified Novel Method for Edge Detection in Grayscale Images Using Adaptive Thresholding J. Adv. Comput. Networks 3 157–61

[17] Kaur H and Sohi N 2017 A Study for Applications of Histogram in Image Enhancement Int. J. Eng. Sci. 2 59–63

[18] Heyduk A, Joostberens J, Nowrot A, Pielot J and Fabiańczyk K 2018 Image histogram as a tool for a coal stream homogeneity evaluation IOP Conf. Ser. Mater. Sci. Eng. 427