Image Segmentation Using Transition Region and K-Means Clustering

Ahmad Wahyu Rosyadi, Nanik Suciati, *Member, IAENG.*

***Abstract*— Several methods based on the transition region have been developed in image segmentation. Most are reported as effective methods with some limitations. Some are difficult to reduce background appearance, while others are difficult to find intact objects. A method which is capable of extracting a more intact transition region while reducing the background appearance is needed. In this study, we propose a novel method to extract a more intact transition region by combining the transition region with k-means clustering. First, a grayscale image is simplified to become a simple image by using k-means clustering. The simple image is a version of the original grayscale image with a lower gray level variation. Two transition regions are extracted: from the original image and the simple image. Then, the edge linking process is conducted based on the Quasi-Euclidean distance on the original image transition region by using the simple image transition region as a reference. Finally, region filling is done to get areas that are considered as objects. The proposed method is compared with two other transition region-based methods, and the experimental results show that the proposed method has the best performance in terms of image segmentation in general and the appearance of objects in the segmentation result.**

***Index Terms*— Transition region, K-means clustering, Edge linking, Quasi-euclidean, Region filling**

I. Introduction

Image segmentation is one of the most basic types of pre-processing in digital image processing and computer vision to extract objects from an image based on image characteristics such as color, gray level, and texture [1]. It can be used for further applications such as automatic measurement of object weight and size [2], object tracking [3]–[5], character identification, object recognition [6], [7] and image retrieval [8], [9]. There are several types of image segmentation techniques: thresholding based technique [10], boundary based technique [11], [12], region based technique [13], and hybrid based technique [14].

Thresholding-based image segmentation assumes objects and backgrounds have different values of gray level. This technique is conducted by determining a threshold value which can distinguish the gray level of objects and backgrounds. Pixel that has gray level value above the threshold is considered an object whereas pixel that has gray level value under the threshold is considered a background, or vice versa. Boundary-based image segmentation technique produces edge or boundary of an object. The region-based technique is conducted by grouping pixels with their neighbor pixels that have similar values [12], separating pixels that have different values, or combining both techniques [15]. Hybrid image segmentation technique is used to obtain a better segmentation result by combining two or more techniques [1].

Transition region based image segmentation is one of the hybrid techniques that combine region and thresholding based technique [14]. There are several studies of transition region-based image segmentation such as Zhang and Gerbrands (1991) that proposed the Effective Average Gradient (EAG) to generate transition regions in the image segmentation application [16]. This method is successfully conducted on an image that has drastic gray level changes. However, it is not suitable for the image that has a high variance of gray level. Then, a method that can solve this problem by using the local entropy for transition region extraction is proposed [17]. However, this method may generate false transitional pixels if there are many variations of gray level in the neighbor window of the pixel because its local entropy value will increase.

To overcome this, a method of transition region extraction using a modified local entropy method (MLE) is proposed [18]. It extracts the transition region by observing the degree and frequency of gray level changes. The extracted transition region of this method is better than LE's. However, it still needs some parameters that have to be set by the user: parameter to control the impact of local complexity and local variance, and parameter to control the amount of transition region pixels.

Then, a method that combines transition region and morphological operations (TRMO) to segment image that contains single or multiple objects is proposed [1]. This method extracts the transition region based on local variance value of each pixel. This method can separate objects from background well. However, this method may still generate false transition region if the gray level complexity of the image is quite high. The appearance of false transition region can make the segmentation result bad because some background pixels may be considered as the object. To solve this problem, a method to repair the extracted transition region (MRTR) is proposed [14]. This method is capable to reduce the appearance of false

A. W. Rosyadi is with the Department of Informatics, Universitas Qomaruddin Gresik, Indonesia (e-mail: wahyu.rosyad16@gmail.com).
N. Suciati is with the Department of Informatics, Institut Teknologi Sepuluh Nopember Surabaya, Indonesia (e- mail: nanik@if.its.ac.id).

transition region so that the segmentation result is better. But this method may result in a bad segmentation if the transition region is not fully connected. This transition region needs to be connected by the edge linking process. However, the edge linking process in this method cannot connect two endpoints that have long distance. So a method that can connect the endpoints of transition region that have close or long distance is very much needed. In this study, we propose a novel method to extract more intact transition region by combining the transition region with k-means clustering. The more intact transition region will make the segmentation result better because it can show more object pixels in the segmentation result.

II. RESEARCH METHOD

Image segmentation phase with the proposed method is shown in Fig. 1. A grayscale image is used as an input image that will be segmented. The image will be used by two types of processes: transition region extraction of the original image and transition region extraction of the simple image separately. Next, an edge linking process will be performed on the transition region of the original image based on the transition region of the simple image. Region filling is performed on the transition region from the edge linking process to produce an area that is considered an object. Then the segmentation process is carried out by displaying pixels that are in the area transition region. And the result of this system is a segmentation image that contains single or multiple objects.

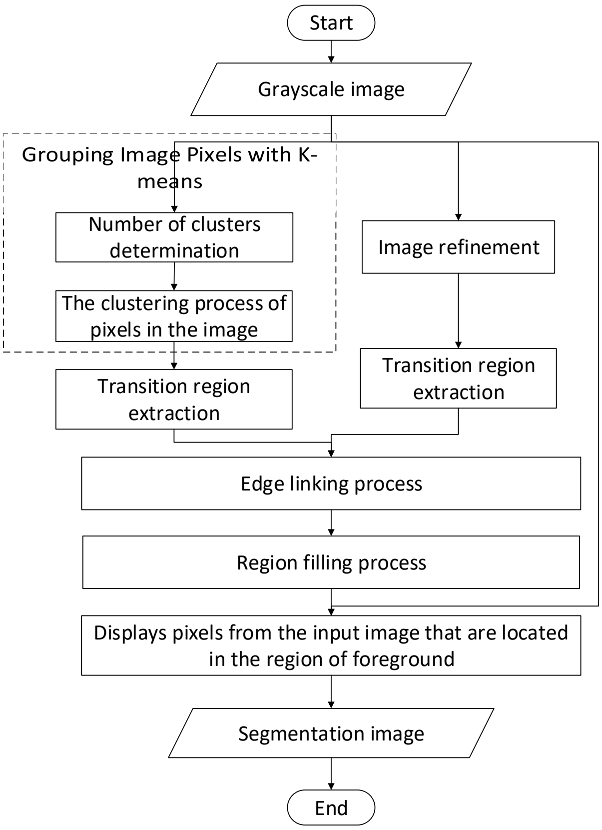


Fig. 1. Proposed method.

A. Transition Region

The transition region is a structure in an image that can be used to separate object and background. It has three characteristics: it has a width of several pixels near the edge, it should be between the object and the background and surrounds the object, there is a change of gray level at the transition region pixels.

Many descriptors have been proposed for transition region extraction [16], [17], [19]. In this study, the local variance is used for the extraction of the transition region. Regions that contain edges or not can be distinguished by local variance [1]. Edges of an object usually exist in regions with high local variance values. Local variance *LV* for each central pixels *p(i, j)* of *m x m* local neighborhood window can be calculated using Eq. (1).

$$LV(i,j) = \frac{1}{m^2-1}\sum_{x=1}^{m}\sum_{y=1}^{m}(f(x,y)-fm)^2 \tag{1}$$

with *f(x, y)* defines the gray level of a pixel in the local neighborhood window and *fm* defines the mean of gray level in the local neighborhood window. By moving the window from left to right and top to bottom, the local variance calculation is run to get the local variance value matrix. The values in the matrix will be compared with the global threshold value *Tg*. The *Tg* value can be calculated using Eq (2).

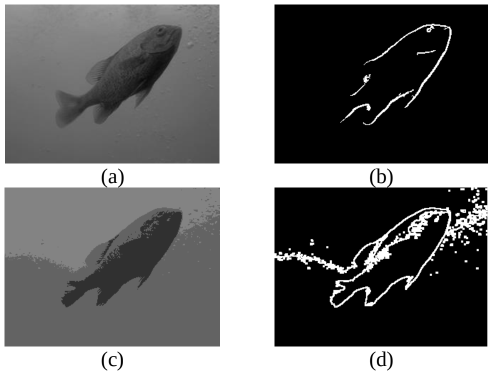


Fig. 2. (a) Grayscale image, (b) transition region of the original grayscale image, (c) simple image, and (d) transition region of the simple image.

$$T_g = \frac{1}{M \times N}\sum_{k=1}^{M}\sum_{l=1}^{N}F(k,l) \tag{2}$$

where *M* defines the height of the image, *N* defines the width of the image, *F(k, l)* is the gray level of the pixel *(k, l)* in the input image. Pixels that have a local variance value greater than or equal to *Tg* will be considered as the transitional pixel. Transition regions are arranged by grouping the transitional pixels that are connected and making each group as a different transition region as shown in Fig. 2(b).

This study uses two transition region images: transition region from the original image and a transition region from the image that its gray level has been simplified using k-means (simple image). These two transition regions have different strengths and weaknesses as shown in Table I.

TABLE I
STRENGTHS, WEAKNESSES, AND FUNCTIONS OF TWO TRANSITION REGION IMAGES

	Transition region from the original image	Transition region from a simple image
Weaknesses	It is difficult to produce a complete transition region in a complex gray level image.	It is difficult to determine the location of objects because they produce too many transition regions.
Strengths	It is easy to determine the position of an object because the majority of the transition region is extracted around the object.	It can produce a more complete transition region because the image has been simplified.
Function in this system	As a transition region that will be connected.	As a connection to the transition region from the original image that is broken.

The quality of the original image transition region is influenced by the complexity of the gray level of the image. In an image with a simple gray level complexity, the transition region can be extracted intact. However, the transition region may be extracted less intact or many rises in the background if the gray level of the image is quite complex as in Fig. 2(b). In contrast, transition region from a simple image is extracted intact because there is a drastic gray level change on the edge of each cluster so it can be used to repair the transition region from the original image as shown in Fig. 2(d).

(a) (b) (c)

Fig. 3. (a) Input image, (b) extracted transition region that contains many false transitional pixels, and (c) extracted transition region after image refinement.

The transition region of the original image is very dependent on the characteristic of the image background. This transition region can be generated well if the image contains a simple background. But, this transition region may be generated bad (false transitional pixels appear) if the background of the image is textured or contains high variations of gray levels. To solve this problem, some parts of the MRTR method are used to extract and repair this transition region. The extracted transition region sometimes still contains many small transition regions. The small transition region is removed because it is considered as noise by several existing methods [1], [14], [20].

B. Image Refinement

Image refinement is used to correct the input image that produces too many false transitional pixels by smoothing them based on the percentage of false transitional pixels as shown in Fig. 3. False transitional pixel is a transitional pixel that often appears in the center of the foreground or background area as shown in Fig. 3(b). False transitional pixels are usually separated from the neighboring transitional pixels at very close distances (no more than 2 pixels) [14]. There are several steps to carry out the transition region refinement:

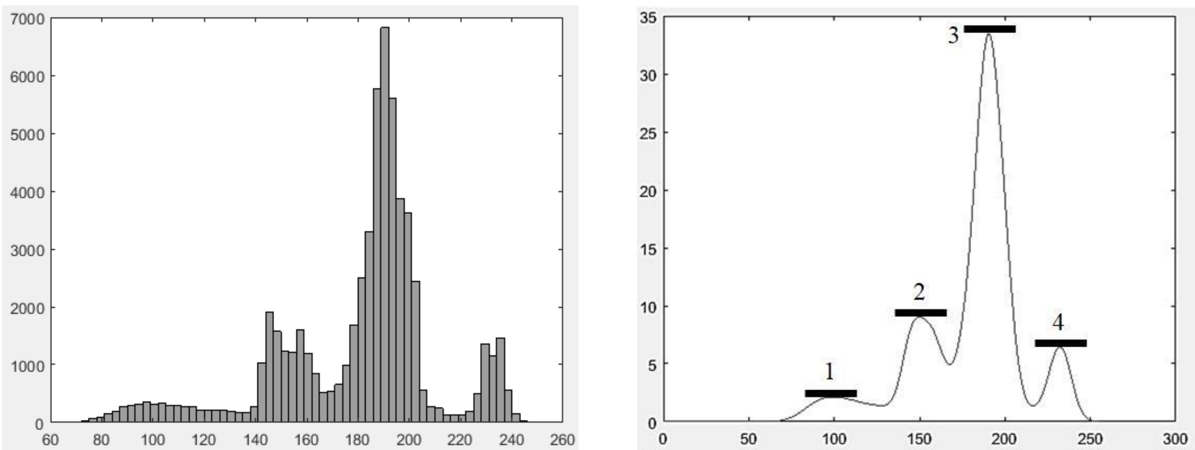
1. Transition region extraction
2. Calculation of the adjacent transition region percentage
3. The median filter is conducted to the input image if the percentage of the adjacent transitional pixel is greater than the normal proximity threshold. The value of a normal proximity threshold is 3 [14].

C. Grouping Image Pixels with K-Means

This phase aims to produce a simpler gray level image as shown in Fig. 2(c). The image will be used to extract a simple image transition region. The transition region of this image is expected to be more intact and can be used to select and connect the transition region from the original image that is less intact. This phase includes the process of determining the number of clusters and the clustering process of pixels in the image.

This study uses k-means clustering techniques with an adaptive number of clusters by the input image. There are several steps to determine the number of clusters:

1. The gray level histogram of the input image is first generated as shown in Fig. 4(a).
2. Probability density estimate function is calculated to each gray level in the histogram using Eq. (3). It is



(a) (b)

Fig. 4. Histogram (a) Input image and (b) Probability density estimate function.

used to generate the smoothed histogram. This histogram contains points that can be used as peaks or clusters as in Fig. 4(b).

$$pf(i) = \frac{1}{n}\sum_{x=1}^{M}\sum_{y=1}^{N} K\left(\frac{i - F(x,y)}{h}\right), \qquad i = 0, \ldots, 255 \tag{3}$$

with n stating the number of gray level in the histogram, h is the bandwidth, and the $K(.)$ represents the Gauss kernel [21].

3. Compare each point in the histogram to the two neighboring points. Points that have a pf value greater than the pf value of one point before and one point thereafter will be determined as peaks as shown in Eq. (4).

$$k = \sum_{j=0}^{n-1} \delta_j, \delta_j = \begin{cases} 1, pf(j-1) < pf(j)\ and \\ \quad pf(j+1) < pf(j) \\ 0, otherwise \end{cases}, \tag{4}$$

4. Calculate mean μ and standard deviation std using Eq. (5) and (6) respectively.

$$\mu = \frac{\sum_{i=0}^{255}(pf(i) * i)}{\sum_{i=0}^{255} pf(i)} \tag{5}$$

$$std = \sqrt{\frac{\sum_{i=0}^{255}(i-\mu)^2 * pf(i)}{\sum_{i=0}^{255} pf(i) - 1}} \tag{6}$$

5. Merge two adjacent peaks if the distance between them less than half of the standard deviation.

K-means clustering method is used to cluster pixels according to their gray level. The gray level of each pixel will be changed to the centroid gray level of the cluster. This process can make the gray level of the image simpler. The simple image is expected to produce a complete transition region according to the object's shape. The algorithm for k-means clustering:

1. Initialize the number of cluster k and cluster centroids based on the previous process.
2. Calculate the Euclidean distance d between the centroid c_k and each pixel $F(x, y)$ using Eq. (7):

$$d = \sqrt{(F(x,y) - c_k)^2} \tag{7}$$

3. Cluster each pixel into the closest cluster based on the distance d.
4. Recalculate the new position for the centroid using the Eq. (8):

$$c_k = \frac{1}{Nc_k}\sum_{i=1}^{Nc_k} F_i \tag{8}$$

where Nc_k is the number of cluster members k.

5. Repeat the process until there are no changes to the new centroid.

D. Edge Linking

The quality of the transition region depends on the gray level complexity of the input image. Transition region will be extracted well in a simple gray level image. However, in an image with a high gray level variation, the transition region may be difficult to be produced intact. So, the edge linking process is needed to connect the broken transition region and make it more complete.

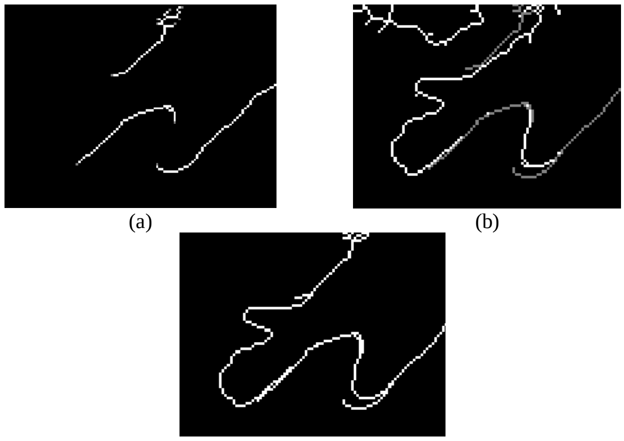


Fig. 5. (a) Not intact transition region, (b) simple image transition region, and (c) transition region after edge linking process.

The edge linking process in this study is slightly different from edge linking in general. Edge linking is generally used to connect two separated points by adding lines that are connected to both points such as curve-fitting edge linking and heuristic edge-linking methods [22]. These methods are not used in this study because the edge linking process in this study is more similar to the problem of pathfinding. A connection or path that can be used to connect two separated points of the transition region from the original image is available in the transition region from a simple image as shown in Fig. 5. So the problem that needs to be solved by edge linking in this study is to choose a suitable path that can connect two separated points of the transition region. In this study, Quasi-euclidean distance and Regional Minima are used to solve this problem.

(1) Quasi-euclidean distance

Quasi-euclidean distance Qe is used to calculate the geodesic distance transform of a binary image using Eq. (12) [23]. The binary image is obtained by combining the transition region from the original image and the transition region from the simple image. Transition regions pixels represent valid regions that can be traversed in the distance transform computation. Background regions represent constrained regions that cannot be traversed in the distance computation. For each pixel of valid regions, the geodesic distance transform assigns a number that is the constrained distance between that pixel and the nearest true pixel in the mask. The output of this function is a matrix contains geodesic distances. This matrix will be used by regional minima to find the route that connects two points or pixels with lower distance value.

(2) Regional Minima

A set of pixels *S* and a map *L* that allocates to every pixel *ps* in *S* form an image *I*. An image can be described as a directed graph that contains nodes (pixels of an image) and edges or arcs (pairs of pixel in adjacency relation *Aj* between pixels of *I*) [24].

A regional minimum is a maximal connected set $Rm \subseteq I$ such any edge $(pt, ps) \in Aj$ and $I_{(pt)} < I_{(ps)}$ for every $pt \in Rm$. Regional minima can be computed with the cost function in Eq. (9) [24].

$$fu_{ini}(\langle ps \rangle) = I(ps), for\ all\ ps \in S, \tag{9}$$

$$fu_{ini}(\pi.\langle pt, ps \rangle) = \begin{cases} fu_{ini}(\pi), & if\ I(pt) \leq I(ps), \\ +\infty, & otherwise. \end{cases}$$

Edge linking is conducted on the transition region of the original image that is not yet intact. Edge linking process is not done on the transition region that is intact to avoid unnecessary connections between some foregrounds. So that the intact transition region will be temporarily stored and will be combined with the transition region from the edge linking process. Determination of the transition region that has not been intact is done by using hole filling on the transition region of the original image. If the pixel number of the transition region is more than the number of empty pixels in the transition region's area, the transition region is considered intact. If the opposite is considered as not intact.

The edge linking process in the transition region that is not yet intact *ta* uses the transition region from a simple image *ts* as its link as shown in Fig. 5(a) and (b). Transition regions of original and simple images have different advantages, disadvantages, and functions described in Table I. Transition region ta_l will be labeled with the transition region ts_l that intersects with it. The transition region is then sorted according to the number of pixels. The length of each transition region pta_l is calculated using Eq. (10). The thinning process is done to attenuate and get the end of the transition region. The ends of the transition region will be connected according to the distance and visual similarity obtained from the transition region of a simple image.

$$pta_l = \sqrt{(bb_l - bt_l)^2 + (br_l - bl_l)^2} \tag{10}$$

The connection process at each endpoint of the transition region is carried out sequentially according to the sequence of transition regions obtained previously. The distance between the starting point (the end of the transition region) with the target candidate (the end of another transition region that has the same label) is calculated by reducing the pixel coordinates of the starting point *(xa, ya)* with the pixel coordinates of target candidate (xt_t, yt_t) as shown in Eq. (11). However, if there is no other transition region with the same label, the endpoint is marked as processed and continued with the other endpoint. The result of this process is more intact transition region as shown in Fig. 5(c).

Fig. 6. Some input images from MSRA and Weizmann dataset.

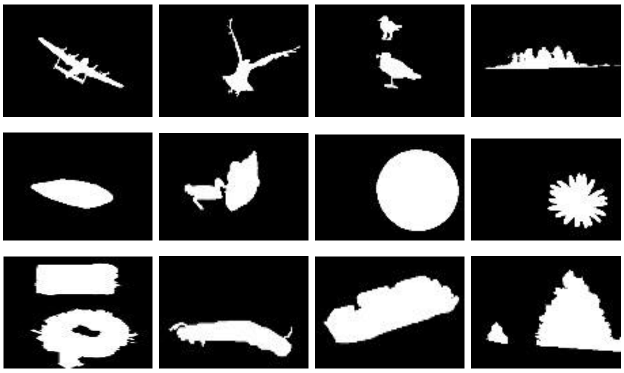


Fig. 7. Ground truths of input images from MSRA and Weizmann dataset.

$$ju_t = \sqrt{(xa - xt_t)^2 + (ya - yt_t)^2} \tag{11}$$

E. Region Filling & The Separation of Foreground and Background

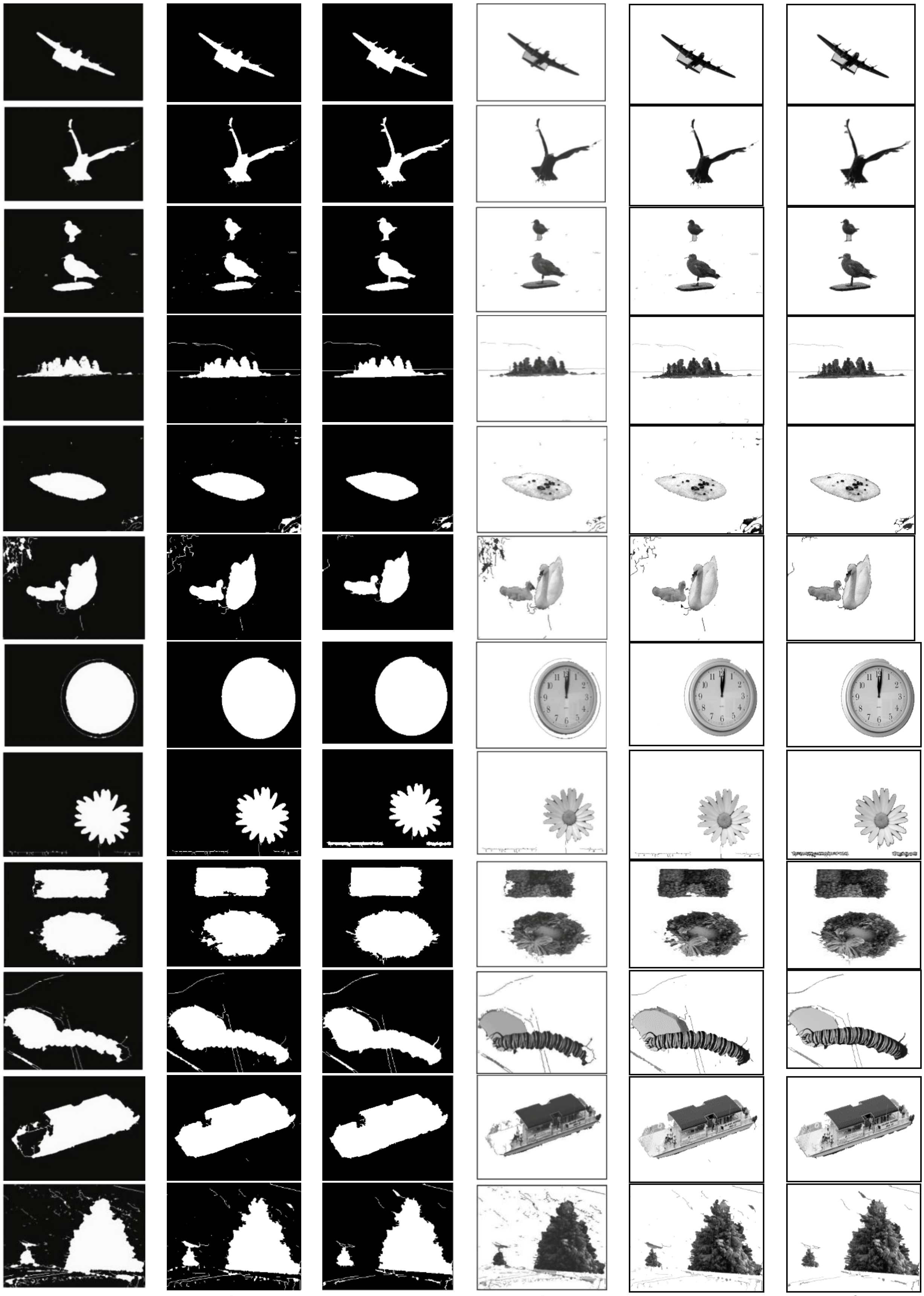
Transition regions that have gone through the process of edge linking and the intact transition regions are combined. This transition region will be processed with the region filling to fill the hole or area so that a binary image is obtained. Pixels from the grayscale image which intersect with the transition region area in the binary image will be displayed.

III. Results And Analysis

Some images from MSRA[25] and Weizmann[26] dataset were used in this study as shown in Fig. 6 and 7. Several stages are conducted to test this system. First, the grayscale image is inserted into the system. The image is

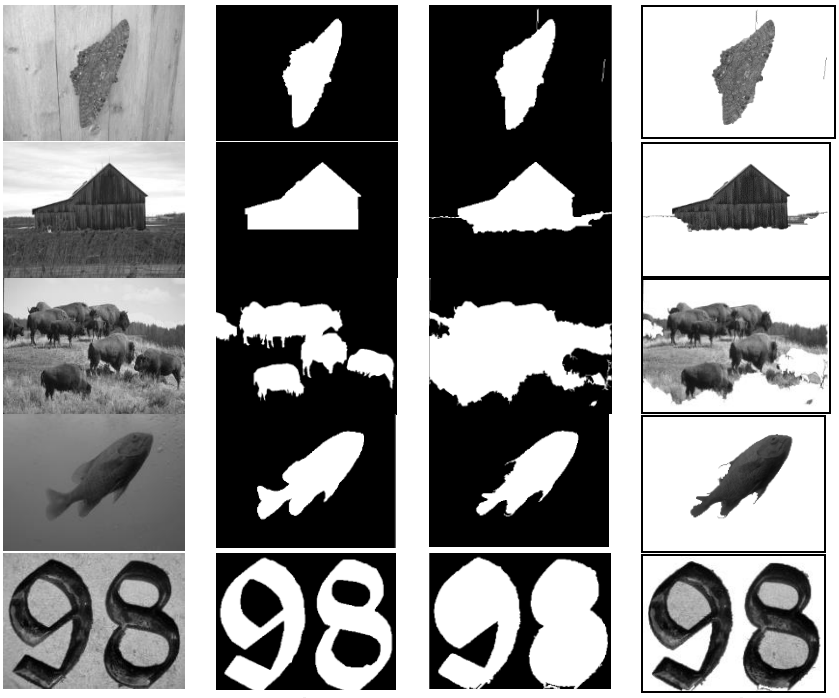
$$Qe[(x_1, y_1), (x_2, y_2)] = \begin{cases} |x_1 - x_2| + (\sqrt{2} - 1) \times |y_1 - y_2|, |x_1 - x_2| > |y_1 - y_2| \\ (\sqrt{2} - 1) \times |x_1 - x_2| + |y_1 - y_2|, otherwise \end{cases} \tag{12}$$

then segmented by the system to obtain a segmented image containing single or multiple objects. The quality of the segmented image is evaluated by calculating misclassification error (ME), false positive rate (FPR), and false negative rate (FNR). ME is applied to calculate the level of classification errors in the image as shown in Eq. (13). FPR shows the capacity of background pixels that are classified as foreground pixels as shown in Eq. (14). FNR shows the capacity of foreground pixels that are classified as background pixels as shown in Eq. (15) [1].



(a) (b) (c) (d) (e) (f)

Fig. 8. Segmentation mask of (a) TRMO, (b) MRTR, (c) the proposed method, segmentation result of (d) TRMO, (e) MRTR, (f) the proposed method.



(a) (b) (c) (d)

Fig. 9. Image (a) Input, (b) Ground truth, (c) Segmentation mask of the proposed method, (d) Segmentation result of the proposed method.

TABLE II
THE PERFORMANCE OF SEVERAL METHODS IN IMAGE SEGMENTATION

Image type	Image	ME			FPR			FNR		
		Proposed	TRMO	MRTR	Proposed	TRMO	MRTR	Proposed	TRMO	MRTR
1) Simple background & simple foreground	Airplane	**0.0120**	0.0179	0.0166	0.0096	**0.0089**	**0.0089**	**0.0524**	0.1670	0.1457
	Eagle	0.0100	**0.0065**	**0.0065**	0.0105	**0.0047**	**0.0047**	**0.0002**	0.0378	0.0376
	Bird	**0.0238**	0.0274	0.0278	0.0198	**0.0192**	0.0194	**0.0925**	0.1684	0.1708
2) Textured background & simple foreground	Island	**0.0093**	0.0106	0.0142	**0.0068**	**0.0068**	0.0077	**0.0416**	0.1168	0.0979
	Iceberg	**0.0083**	0.0091	0.0167	0.0069	**0.0036**	0.0126	**0.0209**	0.0600	0.0550
	Duck	**0.0128**	0.0327	0.0202	**0.0051**	0.0239	0.0097	**0.0682**	0.0957	0.0950
3) Simple background & textured foreground	Clock	**0.0076**	0.0615	0.0082	0.0052	**0.0025**	0.0035	**0.0125**	0.1809	0.0177
	Flower	0.0260	**0.0078**	0.0080	0.0299	**0.0072**	0.0075	**0.0000**	0.0118	0.0118
	Walldécor	**0.0403**	0.0628	0.0584	0.0255	**0.0208**	0.0210	**0.0671**	0.1389	0.1261
4) Textured foreground & textured background	Caterpillar	**0.1010**	0.1052	0.1168	0.1098	**0.0974**	0.1181	**0.0438**	0.0948	0.1082
	Boat	**0.0124**	0.0622	0.0169	0.0030	0.0017	**0.0011**	**0.0342**	0.2023	0.0537
	Mountain-tree	**0.0348**	0.0726	0.0459	**0.0265**	0.0721	0.0364	**0.0588**	0.0741	0.0735
Average		**0.0248**	0.0397	0.0297	0.0215	0.0224	**0.0209**	**0.0410**	0.1124	0.0828

$$ME = 1 - \frac{|B_o \cap B_t| + |Fr_o \cap Fr_t|}{|B_o| + |Fr_o|} \tag{13}$$

where B_o defines the set of background pixels in the ground truth image, Fr_o defines the set of foreground pixels in the image of ground truth, B_t defines the set of background pixels in the segmented image by the system, and Fr_t defines the collection of foreground pixels in the segmented image by the system.

$$FPR = \frac{|B_o \cap Fr_t|}{|B_o|} \tag{14}$$

$$FNR = \frac{|Fr_o \cap B_t|}{|Fr_o|} \tag{15}$$

The values of FPR, FNR, and ME are used to compare the performance of the proposed method with the other transition region based image segmentation methods: TRMO and MRTR. The best values of FPR, FNR, and ME from each image are displayed in bold in Table II. The neighborhood size for transition region computation is 3x3. The reason behind choosing a window of size 3x3 is it provides a sharp transition region in comparison with a large window size i.e., 5x5 or 7x7[1].

There are 3 images: Airplane, Bird, and Eagle used to carry out the experiments on Type-1 images. In this type of image, there are not many drastic changes in the grayscale level in the middle of the background or foreground area so that the background and foreground are not textured. The segmentation process in this type of image should be able to separate the background and foreground properly. The proposed method has the best ME and FNR values on the Airplane and Birds images because it successfully produces good segmentation images. The proposed method displays many foreground pixels while minimizing the appearance of background pixels. Moreover, this method successfully displays the foreground more intact than the other methods as shown in Fig. 8 Airplane and Eagle because it still keeps the old transition region before the thinning process and combines it with the result from the edge linking process. Furthermore, the proposed method succeeds in eliminating many false transitional pixels spread throughout the image so that segmented images become cleaner as seen in Fig. 8 Airplane, Bird, and Eagle. However, this method has the highest FPR value because there are several background pixels attached to the object displayed in the segmented image like the Fig. 8 Eagle. This weakness may be overcome by using thinning techniques as in TRMO but with a kernel that can be flexible so that it does not delete the foreground pixels. Also, there is a background in the middle of the foreground that appears due to the hole filling process such as Fig. 8 Airplane. In cases like this, the system will immediately assume background pixels in the center of the foreground as foreground members, so that the pixels will also be displayed as foreground in the segmentation results. This happened because the region filling process on this system immediately filled the outer edge of the object without checking whether there was a background area or not. The other methods have better FPR values but their ME and FNR values are worse than the proposed method because the thinning process in them is too much to reduce the thickness of the transition region and even delete some parts of the foreground.

The next experiment was carried out on images with Type-2: textured background & simple foreground. The image has a textured background because there are many drastic changes in the grayscale level in the background area. The transition region extraction process in this type of image might generate some false transitional pixels in the background area. This experiment uses the image of Island, Iceberg, and Duck. Based on the results, the proposed method has the best results in almost all tests. The proposed method has the best ME and FNR values for each Type-2 image. This method also has the best FPR value for the image of Island and Duck. These results can be obtained because the proposed method can reduce connection errors between the background and foreground by deleting the small transition regions that can interfere with the edge linking process. Also, it managed to reduce the appearance of background pixels due to the false transitional pixels by using the image refinement process. This capability will make the process of hole filling better because foreground and background can be separated well.

The next experiment was carried out on Type-3 images: simple background & textured foreground. There are 3 images: Clock, Flower, and Wall decor used to carry out the experiments on Type-3 images. Images with these characteristics have a simple background so that the possibility of the emergence of the false transitional pixels in the background area is very small. The experimental results show that TRMO has the best FPR value in each image because it successfully reduces the appearance of the background in the segmented image. However, TRMO is too much to reduce the appearance of foreground pixels so that the FNR value is not too good and there are even missing parts of the foreground like the Clock and Wall décor image as we can see in Fig. 8. Unlike TRMO, the proposed method has the best FNR value for each image and the best ME value in almost all images. The proposed method succeeds in producing a more complete object than the TRMO and MRTR as seen in Fig. 8 Clock and Wall décor because it can connect two disconnected edges both near and far. The proposed method uses the size of the transition region that will be connected as the connection distance limit so that the distance between endpoints that will be connected becomes flexible. This method connects separated endpoints using the edge linking process based on the routes obtained from the simple image transition region. The connection route from the simple image transition region is needed because it is used to determine whether two endpoints of the transition region can be connected or not. If a route that can connect the two endpoints is found, the route or line is added to connect them. The edge linking process in this study is very dependent on the number of transition regions that need to be connected and the complexity of the connection route from the simple image transition region. The fewer transition region of the original image that needs to be connected or fewer transition region of the simple image will speed up the segmentation process.

Another experiment was conducted on Type-4 images: textured foreground & textured background. There are 3 images: Caterpillar, Boat and Mountain trees used to carry out the experiments. The image with this characteristic has a textured background and foreground so that there may be many false transitional pixels in the background and foreground area. Based on the experimental results, the proposed method has the best ME and FNR values for each image. It shows that in images with textured foreground & textured background, the proposed method can segment the image very well. Fig. 8 Mountain trees shows that the proposed method can prevent the merging of false transitional pixels and transition region because the proposed method also adds the ability of MRTR on the image refinement process. It shows that several false transitional pixels are still separated because it does not have a route to the transition regions. It makes the segmentation result better because the pixels in the false transitional pixels do not blend with the foreground pixels so that the appearance of background pixels can be minimized. Furthermore, the proposed method manages to reduce the appearance of false transitional pixels scattered in the middle or around the foreground area while still maintaining the integrity of the foreground shape as shown in Fig. 8 Mountain trees, Boats, and Caterpillar. However, sometimes this method is still difficult to produce good clusters (too many clusters). This will make the edge linking process worse because there are too many choices of routes that must be processed. This can prolong the processing time and may produce a less optimal foreground edge that can cause background pixels to be segmented into objects like Fig. 9 Buffaloes. So it is hoped that there will be further research that examines the improvement of the clustering process in this method.

IV. Conclusion

In this study, a novel method to extract more intact transition region by combining the transition region with k-means clustering for image segmentation has been presented. The proposed method has better performance to distinguish foreground and background than TRMO and MRTR. It happens because the proposed method can produce more complete objects while still reducing the appearance of the background. However, the proposed method sometimes displays too many background pixels as foreground due to the edge linking process which causes this method to have a worse average FPR value than MRTR.

References

[1] P. Parida and N. Bhoi, "Transition region based single and multiple object segmentation of gray scale images," *Eng. Sci. Technol. an Int. J.*, vol. 19, no. 3, pp. 1206–1215, 2016.

[2] S. Torres, C. Arroyo, and R. Serrano, "Automatic Measurement of Fish Weight and Size by Processing Underwater Hatchery Images," *Eng. Lett.*, vol. 26, no. 4, pp. 461–472, 2018.

[3] Z. Wang, B. Chen, J. Wu, and T. Yan, "Real-Time Image Tracking with An Adaptive Complementary Filter," *IAENG Int. J. Comput. Sci.*, vol. 45, no. 1, pp. 97–103, 2018.

[4] S. Rattarom, S. Uttama, and N. Aunsri, "Model Construction and Validation in Low-cost Interpolation-Based Gaze Tracking System," *Eng. Lett.*, vol. 27, no. 1, pp. 87–96, 2019.

[5] A. G. Sooai, K. Yoshimoto, H. Takahashi, S. Sumpeno, and M. H. Purnomo, "Dynamic Hand Gesture Recognition on 3D Virtual Cultural Heritage Ancient Collection Objects Using k -Nearest Neighbor," *Eng. Lett.*, vol. 26, no. 3, pp. 356–363, 2018.

[6] J. Wang, Y. Ruan, B. Zheng, and S. Gao, "Face Recognition Method Based on Improved Gabor Wavelet Transform Algorithm," *IAENG Int. J. Comput. Sci.*, vol. 46, no. 1, pp. 12–24, 2019.

[7] B. Zheng, J. Wang, Y. Ruan, and S. Gao, "Recognition Method Based on Gabor Wavelet Transform and Discrete Cosine Transform," *Eng. Lett.*, vol. 26, no. 2, pp. 228–235, 2018.

[8] A. W. Rosyadi, R. Danardono, S. S. Manek, and A. Z. Arifin, "A Flexible Sub-Block In Region Based Image Retrieval Based On Transition Region," *J. Ilmu Komput. dan Inf.*, vol. 11, no. 1, pp. 42–51, 2018.

[9] Q. Zheng, X. Tian, M. Yang, and H. Wang, "Differential Learning : A Powerful Tool for Interactive Content-Based Image Retrieval," *Eng. Lett.*, vol. 27, no. 1, pp. 202–215, 2019.

[10] F. A. Hermawati, H. Tjandrasa, and N. Suciati, "Hybrid Speckle Noise Reduction Method for Abdominal Circumference Segmentation of Fetal Ultrasound Images," *Int. J. Electr. Comput. Eng.*, vol. 8, no. 3, pp. 1747–1757, 2018.

[11] M. E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," *Proc. - 6th Indian Conf. Comput. Vision, Graph. Image Process. ICVGIP 2008*, pp. 722–729, 2008.

[12] Y. Ryu, Y. Park, J. Kim, and S. Lee, "Image Edge Detection using Fuzzy C-means and Three Directions Image Shift Method," *IAENG Int. J. Comput. Sci.*, vol. 45, no. 1, pp. 1–6, 2018.

[13] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," *Proc. Eighth IEEE Int. Conf. Comput. Vision. ICCV 2001*, vol. 1, no. July, pp. 105–112, 2001.

[14] A. W. Rosyadi and N. Suciati, "Median filter for transition region refinement in image segmentation," *JUTI J. Ilm. Teknol. Inf.*, vol. 16, 2018.

[15] K. Strasters and J. Gerbrands, "Three-dimensional image segmentation using a split, merge and group approach," *Pattern Recognit. Lett.*, vol. 12, no. May, pp. 307–325, 1991.

[16] Y. J. Zhang and J. J. Gerbrands, "Transition region determination based thresholding," *Pattern Recognit. Lett.*, vol. 12, no. 1, pp. 13–23, 1991.

[17] C. Yan, N. Sang, and T. Zhang, "Local entropy-based transition region extraction and thresholding," *Pattern Recognit. Lett.*, vol. 24, no. 16, pp. 2935–2941, 2003.

[18] Z. Li, D. Zhang, Y. Xu, and C. Liu, "Modified local entropy-based transition region extraction and thresholding," *Appl. Soft Comput. J.*, vol. 11, no. 8, pp. 5630–5638, 2011.

[19] Z. Li and C. Liu, "Gray level difference-based transition region extraction and thresholding," *Comput. Electr. Eng.*, vol. 35, no. 5, pp. 696–704, 2009.

[20] Z. Li, G. Liu, D. Zhang, and Y. Xu, "Robust single-object image segmentation based on salient transition region," *Pattern Recognit.*, vol. 52, pp. 317–331, 2016.

[21] M. Shakeri, M. H. Dezfoulian, H. Khotanlou, A. H. Barati, and Y. Masoumi, "Image contrast enhancement using fuzzy clustering with adaptive cluster parameter and sub-histogram equalization," *Digit. Signal Process. A Rev. J.*, vol. 62, pp. 224–237, 2017.

[22] W. K. Pratt, *DIGITAL IMAGE PROCESSING*, vol. 5. 2001.

[23] S. Jayaraman, S. Esakkirajan, and T. Veerakumar, *Digital Image Processing*. New Delhi: Tata McGraw-Hill Education, 2011.

[24] J. Stolfi and R. D. A. Lotufo, "The Image Foresting Transform : Theory , Algorithms , and Applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. June 2014, pp. 19–29, 2004.

[25] T. Liu, J. Sun, N.-N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," *IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 1–8, 2007.

[26] S. Alpert, M. Galun, A. Brandt, and R. Basri, "Image segmentation by probabilistic bottom-up aggregation and cue integration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, pp. 315–327, 2012.

Ahmad Wahyu Rosyadi received the bachelor degree in Computer Engineering from Universitas Islam Negeri Maulana Malik Ibrahim in 2014, the master degree in Computer Science from Insitut Teknologi Sepuluh Nopember in 2018. He became a lecturer in Department of Informatics, Universitas Qomaruddin since 2018. His currenct research interests include Image Processing, Computer Vision, Game, Computer Graphics, and Information Retrieval.

Nanik Suciati received the bachelor degree in Computer Engineering from Insitut Teknologi Sepuluh Nopember in 1994, the master degree in Computer Science from the University of Indonesia in 1998, and the doctoral degree in Information Engineering from the Hiroshima University in 2010. She became a lecturer in Department of Informatics, Institut Teknologi Sepuluh Nopember since 1994. Her currenct research interests include Computer Vision, Computer Graphics, and Computational Intelligence.