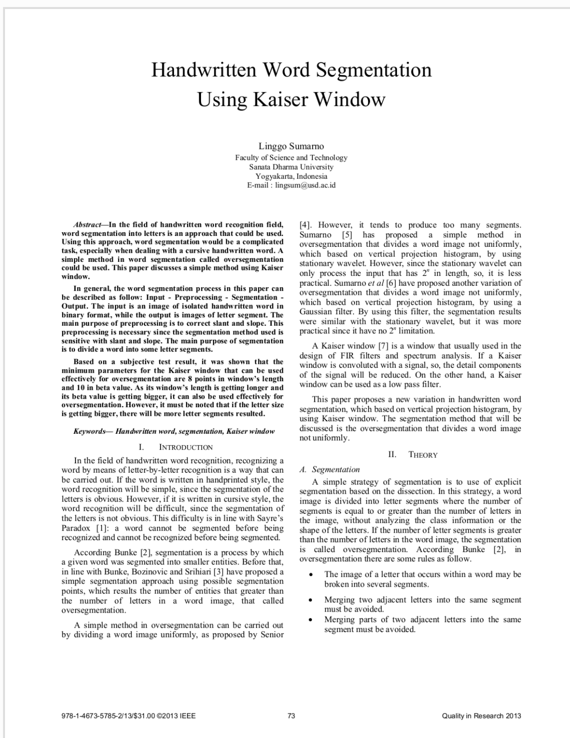
Digital Receipt

This receipt acknowledges that Turnitin received your paper. Below you will find the receipt information regarding your submission.

The first page of your submissions is displayed below.

Submission author:	Sumarno Linggo
Assignment title:	Periksa similarity
Submission title:	Handwritten Word Segmentation Using Kasier Window
File name:	2013_Juni_Handwritten_Word_Segmentation_Using_Kasier_W...
File size:	1.67M
Page count:	6
Word count:	3,503
Character count:	17,641
Submission date:	07-Feb-2018 02:29PM (UTC+0700)
Submission ID:	912401303



Handwritten Word Segmentation Using Kaiser Window

Linggo Sumarno
Faculty of Science and Technology
Sanata Dharma University
Yogyakarta, Indonesia
E-mail : lingsum@usd.ac.id

***Abstract*—In the field of handwritten word recognition field, word segmentation into letters is an approach that could be used. Using this approach, word segmentation would be a complicated task, especially when dealing with a cursive handwritten word. A simple method in word segmentation called oversegmentation could be used. This paper discusses a simple method using Kaiser window.**

In general, the word segmentation process in this paper can be described as follow: Input - Preprocessing - Segmentation - Output. The input is an image of isolated handwritten word in binary format, while the output is images of letter segment. The main purpose of preprocessing is to correct slant and slope. This preprocessing is necessary since the segmentation method used is sensitive with slant and slope. The main purpose of segmentation is to divide a word into some letter segments.

Based on a subjective test result, it was shown that the minimum parameters for the Kaiser window that can be used effectively for oversegmentation are 8 points in window's length and 10 in beta value. As its window's length is getting longer and its beta value is getting bigger, it can also be used effectively for oversegmentation. However, it must be noted that if the letter size is getting bigger, there will be more letter segments resulted.

***Keywords*— Handwritten word, segmentation, Kaiser window**

I. INTRODUCTION

In the field of handwritten word recognition, recognizing a word by means of letter-by-letter recognition is a way that can be carried out. If the word is written in handprinted style, the word recognition will be simple, since the segmentation of the letters is obvious. However, if it is written in cursive style, the word recognition will be difficult, since the segmentation of the letters is not obvious. This difficulty is in line with Sayre's Paradox [1]: a word cannot be segmented before being recognized and cannot be recognized before being segmented.

According Bunke [2], segmentation is a process by which a given word was segmented into smaller entities. Before that, in line with Bunke, Bozinovic and Srihiari [3] have proposed a simple segmentation approach using possible segmentation points, which results the number of entities that greater than the number of letters in a word image, that called oversegmentation.

A simple method in oversegmentation can be carried out by dividing a word image uniformly, as proposed by Senior [4]. However, it tends to produce too many segments. Sumarno [5] has proposed a simple method in oversegmentation that divides a word image not uniformly, which based on vertical projection histogram, by using stationary wavelet. However, since the stationary wavelet can only process the input that has 2^n in length, so, it is less practical. Sumarno *et al* [6] have proposed another variation of oversegmentation that divides a word image not uniformly, which based on vertical projection histogram, by using a Gaussian filter. By using this filter, the segmentation results were similar with the stationary wavelet, but it was more practical since it have no 2^n limitation.

A Kaiser window [7] is a window that usually used in the design of FIR filters and spectrum analysis. If a Kaiser window is convoluted with a signal, so, the detail components of the signal will be reduced. On the other hand, a Kaiser window can be used as a low pass filter.

This paper proposes a new variation in handwritten word segmentation, which based on vertical projection histogram, by using Kaiser window. The segmentation method that will be discussed is the oversegmentation that divides a word image not uniformly.

II. THEORY

A. Segmentation

A simple strategy of segmentation is to use of explicit segmentation based on the dissection. In this strategy, a word image is divided into letter segments where the number of segments is equal to or greater than the number of letters in the image, without analyzing the class information or the shape of the letters. If the number of letter segments is greater than the number of letters in the word image, the segmentation is called oversegmentation. According Bunke [2], in oversegmentation there are some rules as follow.

- The image of a letter that occurs within a word may be broken into several segments.
- Merging two adjacent letters into the same segment must be avoided.
- Merging parts of two adjacent letters into the same segment must be avoided.

Handwritten Word Segmentation Using Kasier Window

by Linggo Sumarno

Handwritten Word Segmentation Using Kaiser Window

Linggo Sumarno
Faculty of Science and Technology
Sanata Dharma University
Yogyakarta, Indonesia
E-mail : lingsum@usd.ac.id

***Abstract*—In the field of handwritten word recognition field, word segmentation into letters is an approach that could be used. Using this approach, word segmentation would be a complicated task, especially when dealing with a cursive handwritten word. A simple method in word segmentation called oversegmentation could be used. This paper discusses a simple method using Kaiser window.**

In general, the word segmentation process in this paper can be described as follow: Input - Preprocessing - Segmentation - Output. The input is an image of isolated handwritten word in binary format, while the output is images of letter segment. The main purpose of preprocessing is to correct slant and slope. This preprocessing is necessary since the segmentation method used is sensitive with slant and slope. The main purpose of segmentation is to divide a word into some letter segments.

Based on a subjective test result, it was shown that the minimum parameters for the Kaiser window that can be used effectively for oversegmentation are 8 points in window's length and 10 in beta value. As its window's length is getting longer and its beta value is getting bigger, it can also be used effectively for oversegmentation. However, it must be noted that if the letter size is getting bigger, there will be more letter segments resulted.

Keywords— Handwritten word, segmentation, Kaiser window

I. Introduction

In the field of handwritten word recognition, recognizing a word by means of letter-by-letter recognition is a way that can be carried out. If the word is written in handprinted style, the word recognition will be simple, since the segmentation of the letters is obvious. However, if it is written in cursive style, the word recognition will be difficult, since the segmentation of the letters is not obvious. This difficulty is in line with Sayre's Paradox [1]: a word cannot be segmented before being recognized and cannot be recognized before being segmented.

According Bunke [2], segmentation is a process by which a given word was segmented into smaller entities. Before that, in line with Bunke, Bozinovic and Srihiari [3] have proposed a simple segmentation approach using possible segmentation points, which results the number of entities that greater than the number of letters in a word image, that called oversegmentation.

A simple method in oversegmentation can be carried out by dividing a word image uniformly, as proposed by Senior [4]. However, it tends to produce too many segments. Sumarno [5] has proposed a simple method in oversegmentation that divides a word image not uniformly, which based on vertical projection histogram, by using stationary wavelet. However, since the stationary wavelet can only process the input that has 2^n in length, so, it is less practical. Sumarno *et al* [6] have proposed another variation of oversegmentation that divides a word image not uniformly, which based on vertical projection histogram, by using a Gaussian filter. By using this filter, the segmentation results were similar with the stationary wavelet, but it was more practical since it have no 2^n limitation.

A Kaiser window [7] is a window that usually used in the design of FIR filters and spectrum analysis. If a Kaiser window is convoluted with a signal, so, the detail components of the signal will be reduced. On the other hand, a Kaiser window can be used as a low pass filter.

This paper proposes a new variation in handwritten word segmentation, which based on vertical projection histogram, by using Kaiser window. The segmentation method that will be discussed is the oversegmentation that divides a word image not uniformly.

II. Theory

A. Segmentation

A simple strategy of segmentation is to use of explicit segmentation based on the dissection. In this strategy, a word image is divided into letter segments where the number of segments is equal to or greater than the number of letters in the image, without analyzing the class information or the shape of the letters. If the number of letter segments is greater than the number of letters in the word image, the segmentation is called oversegmentation. According Bunke [2], in oversegmentation there are some rules as follow.

- The image of a letter that occurs within a word may be broken into several segments.
- Merging two adjacent letters into the same segment must be avoided.
- Merging parts of two adjacent letters into the same segment must be avoided.

A strategy in explicit segmentation based on the dissection is to use vertical projection histogram of a word image. For example, Fig. 1(b) shows the shape of the histogram "hills" that represent the letter segments of the word image of Fig. 1(a). As a hypothesis, if the basic shape of the histogram hills can be made more obvious, the separation of letter segments can be made easier. The shape of the histogram hills in Fig. 1(b) still shows the rough shape (i.e. its basic shapes are less obvious), since there are details components shown. In order to make its basic shapes more obvious, a low pass filter is needed to remove the detail components. According Sumarno *et al* [6], a low pass filter for that purpose should have the following properties.

- It must be able to remove many detail components.
- It must be able to maintain histogram hills, which represent letter segments. On the other hand, it must have high localization property.

As shown in Fig. 1(c), the detail components have been removed by using a low pass filter. Fig. 1(d) shows the vertical lines that indicate the local minimum places. Finally, Fig. 1(e) shows the segmentation result where the segmentation places are based on the local minimum places.

B. Kaiser Window

Kaiser window $w(n)$ with the length N is defined by

$$w(n)=\begin{cases}\dfrac{I_0\left(\beta\sqrt{1-((n-0.5N)/(0.5N))^2}\right)}{I_0(\beta)}, & 0\le n\le N\\ 0, & \text{otherwise}\end{cases} \tag{1}$$

where $I_0(\cdot)$ is the zero-th order Modified Bessel function of the first kind, which can be computed using

$$I_0(x)=1+\sum_{k=1}^{\infty}\left[\frac{(0.5x)^k}{k!}\right]^2 \tag{2}$$

C. Convolution

As described above, the Kaiser window can be used as a low pass filter. Low pass filtering of the signal $x(n)$ with a Kaiser window $w(n)$ is the convolution process between $x(n)$ with $w(n)$. Since the signal $x(n)$ has finite length, there will be distortion at the edges of filtering results. In order to overcome that problem, Misiti *et al* [8] have proposed a method by lengthening the signal $x(n)$ by means of signal repetition which called periodization. For example $x(n) = \{x(1), x(2), \ldots, x(M)\}$ is the signal to be convoluted with Kaiser window $w(n)=\{w(1), w(2), \ldots, w(N)\}$, where N is an even number, and $M > N$, so the signal repetition by means of periodization will be

$$x_{per}(n)=\{x(M-n+1),\ldots,x(M),x(1),\ldots,x(M),x(1),\ldots,x(N-1)\} \tag{3}$$

The convolution of lengthen signal $x_{per}(n)$ with Kaiser window $w(n)$ will give output signal

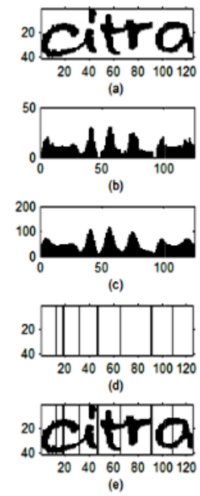


Fig. 1. An example of segmentation using vertical projection histogram; (a) Input image; (b) Vertical projection histogram; (c) Filtering result using Kaiser window 12 points with β value 10; (d) Vertical lines that indicate local minimum places; (e) Segmentation result. (Notes: Vertical and horizontal axes units are in pixels)

$$y(n)=x_{per}(n)*h(n) \tag{4}$$

or

$$y(n)=\sum_j x_{per}(j)\,h(n+1-j) \tag{5}$$

where the length of $M+N-1$ is greater than the length of $x(n)$. In order that the output signal has the same length with the input signal, so, only the certain parts of the output signal are selected. For example, the output of the convolution above is $y(n)=\{y(1), y(2), \ldots, y(L)\}$, where $L=M+N-1$, so, the selected certain parts of $y(n)$ are

$$y_{keep}(n)=\{y(N+1),\ldots,y(N+M)\} \tag{6}$$

III. Research Methodology

A. Materials and Equipments

Materials in this research were isolated handwritten word images in binary format. These materials came from the data acquisition sheets scanned at a resolution of 300 dpi. Data were taken from 10 writers from different ages and genders. They wrote the word "citra" as they usually write, i.e. handprinted, cursive, or mixed handprinted and cursive styles. Equipment in this research was a set of computer based on Intel processor Core2Duo E7500 (2,93GHz) and 2GB of RAM, which equipped with MATLAB software.

B. System Development

By using materials and equipments above, a handwritten word segmentation system has been developed (see Fig. 2). In this system, the input is an image of isolated handwritten word in binary format, while the output is images of letter segment.

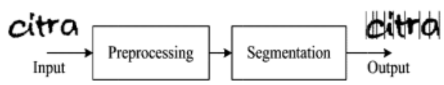


Fig. 2. A word segmentation system.

The main purpose of preprocessing is to correct slant and slope of the input image. This preprocessing is necessary since the segmentation process that uses vertical projection histogram is sensitive with slant and slope variation [6].

C. Preprocessing

Preprocessing steps in Fig. 2 are shown in Fig. 3 [6]. The slope and slant corrections are carried out by using the linear transform for shearing. See Halmos [9] for the detail of the linear transform. The slope and slant corrections will make the image template become bigger than its image. (Note: for the word image input, the image and its template have the same size). To overcome this, a cutting according to bounding box was applied.

The slope and slant correction in Fig. 3 are carried out sequentially by simultaneous slant correction method, as proposed by Slavik & Govindaraju [10]. Fig. 4 [6] shows the slant correction steps for the slope and slant correction. By using this method, first, slope correction using the horizontal projection histogram carried out. Then, followed by slant correction using the vertical projection histogram. Determining of shearing coefficient k_s is carried out by searching a highest variance value in the vertical/horizontal projection histogram. See TABLE I.

D. Segmentation

Segmentation steps are shown in Fig. 5 [6]. Low pass filtering is a convolution process between Kaiser window and vertical projection histogram, which has undergone periodization (see equations (3) to (6)). Letter segments cutting are a cutting of the input image in order to get letter segment images. The cutting places are based on the places of the local minimums in the vertical projection histogram that has undergone low pass filtering. An example of the segmentation steps is shown in Fig. 1.

IV. Results

A. Segmentation results on various Kaiser window parameters

Segmentation results on various Kaiser window parameters, for an example of the word "citra", shown in Fig. 6 and 7. It can be shown in Fig. 6 and 7, for all Kaiser

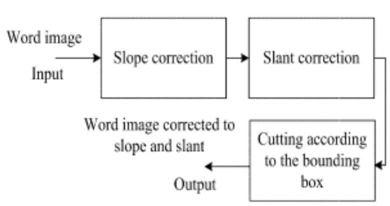


Fig. 3. Preprocessing steps.

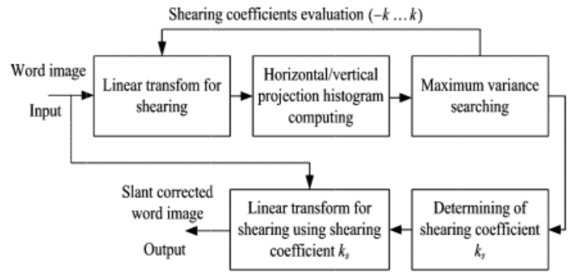


Fig. 4. Slant correction steps.

TABLE I. Parameters for slope and slant corrections.

Correction	Parameters
Slope	Shearing coefficient values for the evaluation of horizontal projection histogram at slope correction {-0.2, -0.15, …, 0.2}. In this case it is assumed that the slope of horizontal projection histogram is in the range of -0.2 … 0.2.
Slant	Shearing coefficient values for the evaluation of vertical projection histogram at slant correction {-0.4, -0.35, … , 0.4}. In this case, it is assumed that the slope of vertical projection histogram is in the range of -0.4 … 0.4.

windows with β value 0 and also with window's length 4 points, the segmentation results show parts of letter "c" and "i", or "i" and "t", or "r" and "a" are in the same segment. It means segmentation errors have been occurred, since they do not follow the oversegmentation rules [2].

As shown in Fig. 6 and 7, in order that oversegmentation can be effectively carried out, the minimum parameters of Kaiser window are 8 points in length and 10 in β value. As its window's length is getting longer and its β value is getting bigger, the oversegmentation can also be effectively carried out. In summary, the segmentation results from Fig. 6 and 7 can be seen in TABLE 2.

B. Segmentation results on various writing styles and sizes

The segmentation results on various writing styles and sizes of 10 words of "citra" which uses a sample of 12 points Kaiser window with β value 10 shown in Fig. 8(a). As a note, the 10 words came from 10 writers who wrote as they usually write. For comparison, segmentations results for the same 10 words of "citra", which uses 16 points Gaussian filter with standard deviation 4 which proposed by Sumarno *et al* [6] are shown in Fig. 8(b). As shown in Fig. 8(a) and 8(b), no segmentation errors occurred since they follow the oversegmentation rules.

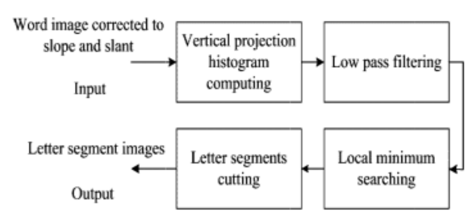


Fig. 5. Segmentation steps.

Fig. 6. Segmentation results on various parameters of Kaiser window; (a) – (f) Length: 4 points, β values: 0, 10, 20, 40, 80, and 160 respectively; (g) – (l) Length: 8 points, β values: 0, 10, 20, 40, 80, and 160 respectively; (m) – (r) Length: 12 points, β values: 0, 10, 20, 40, 80, and 160 respectively; (s) – (x) Length: 16 points, β values: 0, 10, 20, 40, 80, and 160 respectively. (Notes: Vertical and horizontal axes units are in pixels)

V. Discussions

As shown in Fig. 6 and 7 there are three tendencies that started from 8 points window's length. First, if β value is getting bigger, it will lower the occurrence of segmentation errors. As an example, from Fig. 7(s) – 7(x), if β value is getting bigger, the occurrence of segmentation error will be lower. This one indicates that the histogram hills localization is getting lower, since the number of detail components in the vertical projection histogram that removed also getting lower. Therefore, basic shapes of letter entities which represented by histogram hills become more unobvious (since too many component details), so that, the separation between letter entities become more obvious.

TABLE 2. Summary of the segmentation results from Fig. 6 and 7.

β	Window's length							
	4	8	12	16	20	24	28	32
0	×	×	×	×	×	×	×	×
10	×	√	√	×	×	×	×	×
20	×	√	√	√	×	×	×	×
40	×	√	√	√	√	√	×	×
80	×	√	√	√	√	√	√	√
160	×	√	√	√	√	√	√	√

Notes: a. "×" indicates there are segmentation errors.
b. "√" indicates there are no segmentation errors.

Second, if window's length is getting longer, it will raise the occurrence of segmentation errors. As an example, from Fig. 6(h), 6(n), 6(t), and 7(b), if window's length is getting longer, the occurrence of segmentation error will be raise. This one indicates that the histogram hills localization is getting higher, since the number of detail components in the vertical projection histogram that removed also getting higher. Therefore, basic shapes of letter entities which represented by histogram hills become more unobvious, so that, the separation between letter entities become more unobvious also.

Third, if window's length is getting longer and β value is getting bigger, there is no segmentation errors occurred. As an example, from Fig. 6(n), 6(u), 7(d), and 7(k), if window's length is getting longer and β value is getting bigger, there is no occurrence of segmentation errors. This one indicates that the histogram hills localization can be maintained (not too high nor too low), since the number of detail components in the vertical projection histogram that removed can also be maintained. Therefore, basic shapes of letter entities which represented by histogram hills can be maintained obvious, so that, the separation between letter entities can also be maintained obvious.

Fig. 7. The Segmentation results on various parameters of Kaiser window; (a) – (f) Length: 20 points, β values: 0, 10, 20, 40, 80, and 160 respectively; (g) – (l) Length: 24 points, β values: 0, 10, 20, 40, 80, and 160 respectively; (m) – (r) Length: 28 points, β values: 0, 10, 20, 40, 80, and 160 respectively; (s) – (x) Length: 32 points, β values: 0, 10, 20, 40, 80, and 160 respectively. (Notes: Vertical and horizontal axes units are in pixels)

A summary of the segmentation results in TABLE 2 are come from a subjective test results, since they only use one writing variety of the word “citra”. In order to get a more objective test results, the segmentation needs to be tested on even more writing variety of the word “citra”. However, if the segmentation results were observed visually, it would be impractical, since we have to observe it one by one. Therefore, in order to make it more practical, it is not done visually, but by observing the results of the recognition rate of the whole handwritten word recognition. It can be assumed that at the highest recognition rate, the number of segmentation errors is the lowest. Milewski [11] has shown an example of recognizing a word by means of character segment by character segment, which made use of a graph.

Fig. 8(a) and (b) show the segmentation results of the 12 points Kaiser window with β value 10 and the 16 points Gaussian filter with standard deviation 4 respectively. They can be used effectively to segment nine various writing styles and sizes of the word "citra", since there are no segmentation errors. However, Fig. 8(a5), (a8), and (a10) show one additional segment compared to Fig. 8(b5), (b8), and (b10) respectively, which indicates that 12 points Kaiser window with β value 10 a bit worse than 16 points Gaussian filter with standard deviation 4. This case shows that if it is compared with Kaiser window, the Gaussian filter is a bit more able to conserve “small hills” at vertical projection histogram, which represent character entities.

In general, Fig. 8 shows that if the letter size is getting bigger there will be more letter segments resulted. For example, at Fig. 8(a9), (a10), (b9), and (b10), a low pass filtering that made use of the 12 points Kaiser window with β value 10 and also the 16 points Gaussian filter with standard deviation 4 were not “strong” enough to smooth the “big hills” at projection histogram which represent character entities, so that, more letter segments are resulted. Therefore, in order to deal with the letter sizes it is suggested to introduce some size correction steps in the segmentation system.

VI. Conclusion and Suggestion

Based on what has been described above, it can be concluded the following.

- Based on the subjective test result, it was shown that the minimum parameters for the Kaiser window that can be used for oversegmentation effectively are 8 points in window’s length and 10 in β value.
- For the Kaiser window, as the window’s length is getting longer and its β value is getting bigger, it can also be used for oversegmentation effectively.
- If the letter size is getting bigger, there will be more letter segments resulted.

Here are some suggestions to further develop the segmentation system.

(a) (b)

Fig. 8. Segmentation results for 10 writing styles and sizes of "citra"; (a) Using 12 points Kaiser window with β value 10; (b) Using 16 points Gaussian filter with standard deviation 4. (Notes: Vertical and horizontal axes units are in pixels)

- In order to obtain a more objective segmentation results, it is necessary to test more writing varieties and sizes of the word "citra" and for other words.
- In order to be more practical, it is suggested that the segmentation results are observed by observing the results of the recognition rate of the handwritten word recognition, which is the next process after the process of segmentation.
- In order to deal with the letter sizes it is suggested to introduce some size correction steps in the segmentation system.

REFERENCES

[1] K. Sayre, "Machine Recognition of Handwritten Words: A Project Report", *Pattern Recognition*, vol. 5, no. 3, pp. 213-228, 1973.

[2] H. Bunke, "Recognition of Cursive Roman Handwriting – Past, Present and Future", *Proceeding of 7th International Conference on Document Analysis and Recognition,* pp. 448-459, 2003.

[3] R. M. Bozinovic and S.N. Srihari, "Off-Line Cursive Word Recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 68 – 83, 1989

[4] A. W. Senior, *Off-line Cursive Handwriting Recognition using Recurrent Neural Networks*, PhD Thesis, Cambridge: Cambridge University, 1994, unpublish.

[5] L. Sumarno, "Handwritten word segmentation using wavelet" (in Indonesian), *Compile*, vol. 1, no. 1, pp. 80-87, 2008.

[6] L. Sumarno, A. Susanto, and J.E. Istiyanto, "Handwritten word segmentation using 1D Gaussian filter" (in Indonesian), *Sigma*, vol. 12, no. 2 , pp. 179-189, 2009.

[7] J. F. Kaiser, "Nonrecursive Digital Filter Design Using the Io-sinh Window Function", *Proceeding of IEEE Symposium of Circuits and Systems*, pp.20-23, 1974.

[8] M. Misiti, Y. Misiti, G. Oppenheim, J.-M. Poggi, 2005, *Wavelet Toolbox: For Use with MATLAB, Version 3,* Massachusetts: The Mathworks Inc., 2005.

[9] P. R. Halmos, *Finite-Dimensional Vector Spaces*. Springer-Verlag, 1993.

[10] V. Slavik, and V. Govindaraju, "Equivalence of Different Methods for Slant and Skew Corrections in Word Recognition Applications", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 3, no. 3, pp. 323-326, 2001.

[11] R. J. Milewski, *Automatic Recognition of Handwritten Medical Forms for Search Engines*, PhD Thesis, New York: University of New York, 2006, unpublish.

Handwritten Word Segmentation Using Kasier Window

ORIGINALITY REPORT

9%	6%	2%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	citee2015.jteti.ft.ugm.ac.id Internet Source	3%
2	media.neliti.com Internet Source	2%
3	Submitted to Higher Education Commission Pakistan Student Paper	1%
4	Aljahdali, Sultan, Mostafa Nofal, and Yasser Albagory. "A modified array processing technique based on Kaiser window for concentric circular arrays", 2012 International Conference on Multimedia Computing and Systems, 2012. Publication	<1%
5	publications.idiap.ch Internet Source	<1%
6	Submitted to Universiti Teknologi Malaysia Student Paper	<1%
7	Seong-Whan Lee, Chang-Hun Kim, Hong Ma, Yuan Y. Tang. "Multiresolution recognition of	<1%

unconstrained handwritten numerals with wavelet transform and multilayer cluster neural network", Pattern Recognition, 1996
Publication

8	Submitted to Middle East College of Information Technology Student Paper	<1 %
9	inba.info Internet Source	<1 %
10	es.scribd.com Internet Source	<1 %
11	www.yumpu.com Internet Source	<1 %
12	Datar, A.. "Design of Kaiser window based optimized prototype filter for cosine modulated filter banks", Signal Processing, 201005 Publication	<1 %
13	www.scribd.com Internet Source	<1 %

Exclude quotes On
Exclude bibliography On
Exclude matches < 5 words