

Segmentasi Kata Tulisan Tangan Menggunakan Jendela *Blackman*

Lingo Sumarno

Fakultas Sains dan Teknologi

Universitas Sanata Dharma

Yogyakarta, Indonesia

E-mail : lingsum@usd.ac.id

Abstrak—Dalam pengenalan kata tulisan tangan, salah satu strateginya adalah mengenali huruf demi huruf yang menyusun kata tersebut. Dengan strategi ini, bila kata yang akan dikenali ditulis dengan ragam latin, pengenalan huruf demi hurufnya menjadi persoalan yang rumit, karena tidak jelasnya segmentasi antara huruf yang satu dengan huruf yang lain. Untuk mengatasi persoalan tersebut, dapat digunakan suatu metode segmentasi yang dinamakan segmentasi lebih. Tulisan ini membahas metode segmentasi lebih menggunakan jendela *Blackman*. Secara ringkas, proses segmentasi dalam tulisan ini sebagai berikut: Masukan – Pengolahan awal – Segmentasi – Keluaran. Masukan berupa sebuah citra kata terisolasi dalam format biner, serta keluaran berupa sejumlah citra segmen huruf. Pengolahan awal berfungsi untuk mengkoreksi slope dan slant dari citra masukan. Koreksi-koreksi ini diperlukan karena metode segmentasi yang digunakan sensitif terhadap *slope* dan *slant*. Segmentasi berfungsi untuk memecah citra kata menjadi sejumlah citra segmen huruf. Berdasarkan hasil pengujian secara subyektif terlihat bahwa, untuk semua jendela *Blackman* yang lebarnya 8, 12, dan 16 titik, dengan nilai alpha masing-masing mulai dari 0 ; 0,12 ; dan 0,40 dapat digunakan secara efektif untuk keperluan segmentasi. Secara umum, jendela *Blackman* dengan kelebaran mulai dari 8 titik, bila kelebaran dan nilai alpha-nya makin naik, dapat digunakan secara efektif untuk keperluan segmentasi.

Kata kunci—Kata tulisan tangan; segmentasi; jendela *Blackman*

I. PENDAHULUAN

Dewasa ini, dengan semakin berkembangnya teknologi komputer, dapat dibuat komputer yang dapat meniru kemampuan manusia. Salah satu kemampuan manusia yang dicoba untuk ditiru oleh komputer adalah kemampuan untuk mengenali kata tulisan tangan. Bagi manusia, untuk mengenali kata tulisan tangan merupakan hal yang sederhana. Namun tidak demikian halnya dengan komputer, karena komputer memerlukan serangkaian proses yang tertentu untuk melaksanakan hal tersebut. Salah satu strategi dalam perancangan serangkaian proses tersebut adalah dengan berdasarkan pengenalan huruf demi huruf. Jika kata yang akan dikenali dituliskan dengan ragam tegak (*handprinted*), maka proses pengenalannya menjadi sederhana, karena segmentasi huruf yang satu dengan huruf yang lain jelas. Akan tetapi, jika kata yang akan dikenali dituliskan dengan ragam latin (*cursive*) proses pengenalannya menjadi tidak sederhana, karena segmentasi huruf yang satu dengan huruf yang lain

tidak jelas. Hal ini terkait dengan Paradoks Sayre [1]: suatu kata tidak dapat disegmentasi sebelum dikenali dan tidak dapat dikenali sebelum disegmentasi.

Menurut Bunke [2], segmentasi merupakan suatu proses untuk mensegmentasi suatu kata menjadi entitas-entitas yang lebih kecil. Terkait dengan Bunke, sebelum itu Bozinovic dan Srihari [3] mengusulkan suatu cara segmentasi sederhana dengan menggunakan titik-titik segmentasi yang dimungkinkan, yang menghasilkan jumlah entitas yang melebihi jumlah karakter dalam suatu citra kata, yang dinamakan segmentasi lebih.

Dalam segmentasi lebih, suatu cara yang sederhana adalah dengan membagi suatu citra kata dengan lebar segmen huruf yang seragam, sebagaimana diusulkan oleh Senior [4]. Namun dengan cara tersebut ada kecenderungan terlalu banyak segmen huruf yang dihasilkan. Sumarno [5] mengusulkan suatu cara segmentasi lebih menggunakan wavelet stasioner, yang dapat membagi citra kata dengan lebar segmen huruf yang tidak seragam. Dengan cara tersebut tidak ada kecenderungan terlalu banyak segmen huruf yang dihasilkan. Akan tetapi, karena wavelet stasioner hanya memproses masukan yang panjangnya 2^n , cara tersebut menjadi kurang praktis. Sumarno *et al* [6] mengusulkan penggunaan tapis Gaussian dalam melaksanakan segmentasi lebih. Unjuk kerja segmentasi dengan tapis Gaussian mirip dengan wavelet stasioner. Namun, penggunaan tapis Gaussian lebih praktis, karena tidak memerlukan masukan yang panjangnya harus 2^n .

Jendela *Blackman* adalah suatu jendela yang biasanya digunakan dalam pengolahan sinyal digital untuk meminimalkan diskontinuitas tepi-tepi sinyal [7]. Jika jendela *Blackman* dikonvolusikan dengan suatu sinyal, maka komponen-komponen detil dari sinyal tersebut akan berkurang. Ini berarti, telah terjadi penapisan komponen-komponen detil, atau dengan kata lain telah terjadi penapisan pelewat rendah. Oleh karena itu dapat dikatakan bahwa jendela *Blackman* dapat digunakan sebagai tapis pelewat rendah.

Tulisan ini membahas suatu cara baru dalam segmentasi menggunakan jendela *Blackman*. Segmentasi yang dimaksud adalah segmentasi lebih yang membagi citra suatu kata menjadi segmen-segmen huruf yang tidak seragam.

II. DASAR TEORI

A. Segmentasi

Tujuan utama segmentasi adalah memecah suatu kata menjadi huruf-huruf yang menyusunnya. Suatu strategi yang sederhana dalam segmentasi adalah segmentasi eksplisit yang didasarkan pada *dissection*. Pada strategi ini suatu kata dipecah menjadi sejumlah segmen huruf, dengan jumlah segmen yang sama atau lebih besar dari jumlah huruf yang menyusun kata tersebut, dengan tanpa menganalisa informasi kelas dan bentuk huruf. Jika jumlah segmennya melebihi jumlah huruf yang menyusun kata, maka segmentasinya disebut segmentasi lebih. Bunke [2] memberikan beberapa rambu segmentasi lebih sebagai berikut.

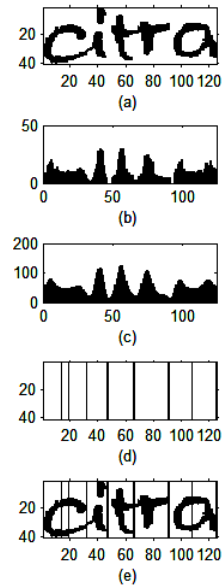
- Citra dari suatu karakter yang menyusun suatu kata dapat dipecah dalam beberapa segmen.
- Penggabungan dua karakter yang berdekatan dalam satu segmen harus dihindari.
- Penggabungan bagian-bagian dari dua karakter yang berdekatan dalam satu segmen harus dihindari.

Suatu metode dalam segmentasi eksplisit yang didasarkan pada *dissection* adalah segmentasi yang menggunakan histogram proyeksi vertikal [6]. Gambar 1 memperlihatkan contoh penggunaan metode ini. Gambar 1(b) memperlihatkan “bukit-bukit” pada histogram proyeksi vertikal yang merepresentasikan segmen-segmen huruf pada Gambar 1(a). Sebagai hipotesis, jika bentuk dasar bukit-bukit tersebut dapat diperjelas, maka separasi antara bukit-bukit tersebut juga akan diperjelas, yang berarti pemisahan segmen-segmen huruf menjadi lebih sederhana. Bentuk bukit-bukit pada histogram proyeksi vertikal, pada Gambar 1(b) masih “kasar”, karena masih banyak terlihat komponen-komponen detail. Untuk mendapatkan bentuk dasar dari bukit-bukit tersebut, maka bukit-bukit tersebut dapat “dihaluskan” dengan penapisan pelewat rendah. Gambar 1(c) memperlihatkan bukit-bukit yang sudah dihaluskan tersebut. Gambar 1(d) memperlihatkan garis-garis vertikal, yang mengindikasikan tempat-tempat segmentasi, yang bersesuaian dengan minimum-minimum lokal pada “perbukitan” histogram proyeksi vertikal. Akhirnya, Gambar 1(e) memperlihatkan hasil segmentasi yang berdasarkan pada garis-garis vertikal pada Gambar 1(d).

B. Jendela Blackman

Jendela *Blackman* $w(n)$ dengan lebar N titik didefinisikan dengan

$$w(n) = a_0 - a_1 \cos\left(\frac{2\pi n}{N-1}\right) + a_2 \cos\left(\frac{4\pi n}{N-1}\right), \quad 0 \leq n \leq N-1 \quad (1)$$

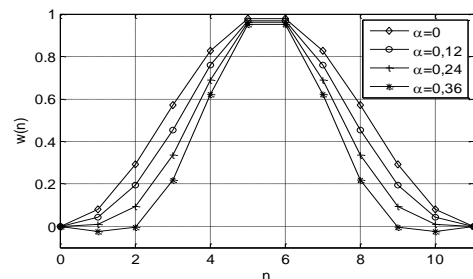


Gambar 1. Contoh Segmentasi Menggunakan Histogram Proyeksi Vertikal; (a) Citra Masukan; (b) Histogram Proyeksi Vertikal; Penapisan Histogram Proyeksi Vertikal Menggunakan Jendela *Blackman* 12 Titik dengan $\alpha = 0,12$; (d) Garis-Garis Vertikal yang Mengindikasikan Tempat-Tempat Minimum Lokal; (e) Hasil Segmentasi. (Catatan: Sumbu Tegak dan Horisontal Dinyatakan dalam Pikel)

dengan $a_0=(1-\alpha)/2$, $a_1=1/2$, dan $a_2=\alpha/2$. Gambar 2 memperlihatkan contoh jendela *Blackman* dengan lebar 12 titik pada berbagai nilai α .

C. Konvolusi

Sebagaimana telah dideskripsikan di atas, jendela *Blackman* dapat digunakan sebagai tapis pelewat rendah. Penapisan pelewat rendah sinyal $x(n)$ dengan jendela *Blackman* $w(n)$ merupakan suatu proses konvolusi antara $x(n)$ dengan $w(n)$. Bila sinyal $x(n)$ mempunyai panjang yang terbatas, dan juga ujung-ujung sinyalnya diskontinu, akan memunculkan distorsi pada ujung-ujung sinyal hasil konvolusinya. Untuk mengatasi hal tersebut, Misiti *et al* [8] telah memperkenalkan suatu metode konvolusi dengan memperpanjang sinyal $x(n)$ melalui pengulangan sinyal yang dinamakan periodisasi.



Gambar 2. Contoh Jendela *Blackman* 12 titik dengan Berbagai Nilai α .

Sebagai contoh sinyal $x(n) = \{x(1), x(2), \dots, x(M)\}$ yang akan dikonvolusi dengan jendela *Blackman* $w(n) = \{w(1), w(2), \dots, w(N)\}$, dengan N adalah bilangan genap, dan $M > N$.

Pengulangan sinyal dengan periodisasi akan menghasilkan sinyal

$$x_{per}(n) = \{x(M-n+1), \dots, x(M), x(1), \dots, x(M), x(1), \dots, x(N-1)\} \quad (2)$$

Konvolusi sinyal yang diperpanjang $x_{per}(n)$ dengan jendela Blackman $w(n)$ akan menghasilkan

$$y(n) = x_{per}(n) * h(n) \quad (3)$$

atau

$$y(n) = \sum_j x_{per}(j) h(n+1-j) \quad (4)$$

dengan panjang $M+N-1$ lebih besar dari panjang $x(n)$. Agar supaya sinyal keluaran mempunyai panjang yang sama dengan sinyal masukannya, maka hanya bagian-bagian tertentu saja yang dipilih. Sebagai contoh, keluaran konvolusi di atas adalah $y(k) = \{y(1), y(2), \dots, y(L)\}$, dengan $L=M+N-1$, sehingga bagian-bagian $y(k)$ yang dipilih adalah

$$y_{keep}(k) = \{y(N+1), \dots, y(N+M)\} \quad (5)$$

III. METODOLOGI PENELITIAN

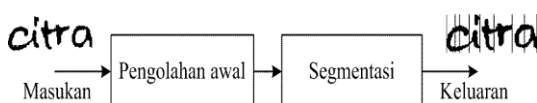
A. Bahan dan Alat Penelitian

Bahan penelitian berupa citra kata tulisan tangan yang terisolasi. Bahan penelitian ini berasal dari lembar pengambilan data yang di-scanned pada resolusi 300 dpi. Data diambil dari 10 penulis yang menuliskan kata "citra" sebagaimana biasanya mereka menulis, yaitu dengan ragam tulisan tegak, latin, atau campuran antara tegak dan latin. Kesepuluh penulis tersebut berasal dari berbagai tingkatan umur dan jenis kelamin. Peralatan penelitian berupa komputer dengan prosesor Intel Core2Duo E7500 (2,93GHz) dan 2GB RAM, yang dilengkapi dengan perangkat lunak MATLAB.

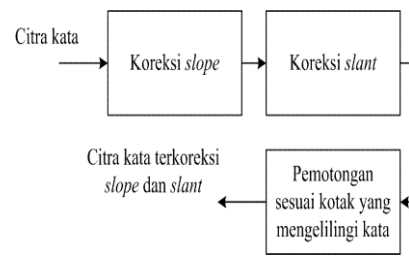
B. Pengembangan Sistem

Sistem yang dikembangkan untuk segmentasi kata tulisan tangan diperlihatkan pada Gambar 3. Masukan sistem berupa citra kata terisolasi dalam format biner, sedangkan keluarannya berupa sejumlah citra dari segmen-segmen huruf.

Pengolahan awal diperlukan untuk mengoreksi *slope* dan *slant* dari citra masukan. Pengolahan awal ini diperlukan karena sistem segmentasi yang digunakan, yang berdasarkan histogram proyeksi vertikal, sensitif terhadap adanya *slope* dan *slant*.



Gambar 3. Sistem Segmentasi Kata Tulisan.



Gambar 4. Langkah-Langkah Pengolahan Awal.

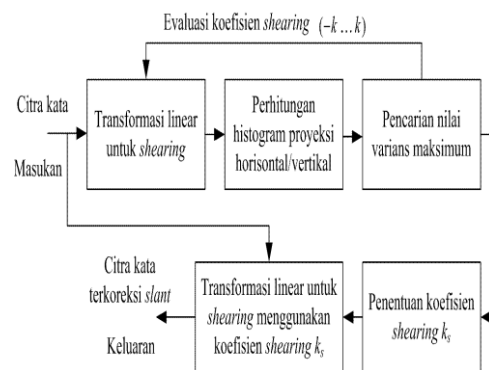
C. Pengolahan Awal

Pengolahan awal pada Gambar 3 diperlihatkan langkah-langkahnya pada Gambar 4 [6]. Koreksi *slope* dan *slant* dilaksanakan dengan transformasi linear untuk *shearing*. Halmos [9] menjelaskan secara detil transformasi linear ini.

Koreksi *slope* dan *slant* pada Gambar 4 dilaksanakan secara berurutan dengan menggunakan metode koreksi *slant* simultan yang dikenalkan oleh Slavik dan Govindaraju [10]. Gambar 5 [6] memperlihatkan langkah-langkah koreksi *slant* simultan ini. Pertama kali dilaksanakan koreksi *slant* menggunakan histogram proyeksi horisontal. Kemudian, dilaksanakan koreksi *slant* menggunakan histogram proyeksi vertikal. Hasilnya adalah citra kata yang sudah terkoreksi *slope* dan *slant*. Penentuan koefisien *shearing* k_s dilaksanakan dengan mencari nilai *varians* tertinggi pada histogram proyeksi horisontal/vertikal. Tabel 1 memperlihatkan parameter-parameter yang digunakan untuk koreksi *slant* dan *slope* ini.

D. Segmentasi

Gambar 6 [6] memperlihatkan langkah-langkah segmentasi yang dilaksanakan. Penapisan pelewat rendah merupakan suatu proses konvolusi dengan periodisasi seperti diperlihatkan pada persamaan (2) - (5). Pemotongan segmen-segmen huruf merupakan citra masukan menjadi sejumlah segmen huruf. Tempat-tempat pemotongannya berdasarkan minimum-minimum lokal yang didapat dari penapisan pelewat rendah dari histogram proyeksi vertikal. Gambar 1 memperlihatkan contoh dari langkah-langkah segmentasi ini.



Gambar 5. Langkah-langkah koreksi *slant*.

TABEL I. PARAMETER-PARAMETER UNTUK KOREKSI *SLANT* DAN *SLOPE*.

Koreksi	Parameter
<i>Slope</i>	Nilai-nilai koefisien shearing untuk evaluasi histogram proyeksi horisontal adalah {0,2 , -0,15 , ... , 0,2}. Dalam hal ini diasumsikan bahwa <i>slope</i> citra kata masukan adalah berada pada rentang nilai koefisien <i>shearing</i> -0,2 ... 0,2.
<i>Slant</i>	Nilai-nilai koefisien shearing untuk evaluasi histogram proyeksi vertikal adalah {0,4 , -0,35 , ... , 0,4}. Dalam hal ini diasumsikan bahwa <i>slant</i> citra kata masukan adalah berada pada rentang nilai koefisien <i>shearing</i> -0,4 ... 0,4.

IV. HASIL DAN PEMBAHASAN

A. Segmentasi pada Berbagai Ukuran Jendela Blackman

Pengujian sistem segmentasi pada Gambar 2 terhadap sebuah masukan yang berupa kata “citra”, yang secara subyektif mewakili ragam penulisan latin dan tegak, diperlihatkan hasilnya pada Gambar 7. Pengujian tersebut dilakukan untuk berbagai ukuran jendela *Blackman*. Sebagaimana terlihat pada Gambar 7, untuk semua jendela *Blackman* yang lebarnya 4 dan 20 titik, terlihat adanya entitas-entitas karakter “c” dan “i”, “i” dan “t”, atau “t” dan “r” yang ada dalam satu segmen. Ini merupakan indikasi terjadinya kesalahan segmentasi, karena segmen-segmen hasil segmentasinya tidak memenuhi semua rambu-rambu segmentasi.

Apabila Gambar 7 dilanjutkan pengujiannya, yaitu untuk masing-masing lebar jendela diberikan nilai α yang makin naik yaitu $0,24 \leq \alpha \leq 0,44$, maka akan didapatkan indikasi-indikasi segmentasi seperti diperlihatkan pada Tabel 2. Sebagaimana terlihat pada Tabel 2, pada sebagian jendela *Blackman* yang lebarnya 12 dan 16 titik, yang masing-masing mempunyai nilai $0 \leq \alpha \leq 0,08$ dan $0 \leq \alpha \leq 0,36$, terlihat adanya indikasi kesalahan segmentasi, karena segmen-segmen hasil segmentasinya tidak memenuhi semua rambu-rambu segmentasi. Selain itu, pada Tabel 2 juga terlihat, untuk semua jendela *Blackman* yang lebarnya 8 titik, serta sebagian jendela *Blackman* yang lebarnya 12 dan 16 titik yang masing-masing mempunyai nilai $\alpha \geq 0,12$ dan $\alpha \geq 0,40$, tidak memperlihatkan adanya indikasi kesalahan segmentasi, karena segmen-segmen hasil segmentasinya memenuhi semua rambu-rambu segmentasi.

TABEL II. INDIKASI KEBERHASILAN SEGMENTASI PADA BERBAGAI UKURAN JENDELA *BLACKMAN*.

α	Lebar jendela (titik)				
	4	8	12	16	20
0	×	√	×	×	×
0,04	×	√	×	×	×
0,08	×	√	×	×	×
0,12	×	√	√	×	×
0,16	×	√	√	×	×
0,20	×	√	√	×	×
0,24	×	√	√	×	×
0,28	×	√	√	×	×
0,32	×	√	√	×	×
0,36	×	√	√	×	×
0,40	×	√	√	√	×
0,44	×	√	√	√	×

Catatan: a. “×” mengindikasikan ada kesalahan segmentasi.
 b. “√” mengindikasikan tidak ada kesalahan segmentasi.

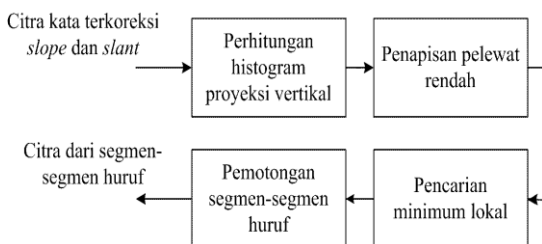
B. Segmentasi pada berbagai ragam penulisan

Pengujian sistem segmentasi pada Gambar 2 yang dilakukan terhadap 10 contoh ragam penulisan kata “citra”, diperlihatkan hasilnya pada Gambar 8. Pengujian untuk suatu contoh penggunaan jendela *Blackman* 12 titik dengan nilai $\alpha = 0,12$, diperlihatkan pada Gambar 8(a). Sebagai pembandingan, pengujian untuk suatu contoh penggunaan tapis Gaussian 16 titik dengan deviasi standar 4 [6] diperlihatkan pada Gambar 8(b). Sebagaimana terlihat pada Gambar 8(a) dan (b), baik penggunaan jendela *Blackman* mau pun tapis Gaussian, dapat digunakan secara efektif untuk keperluan segmentasi, karena tidak terlihat adanya kesalahan segmentasi.

C. Pembahasan

Pada Gambar 7, dimulai dari lebar jendela 8 titik, dengan nilai α yang sama, terlihat adanya kecenderungan jumlah segmen yang makin sedikit jika jendelanya makin lebar. Hal ini merupakan indikasi bahwa jika jendelanya makin lebar, kemampuan lokalisasi jendela makin tinggi karena makin banyak detil histogram proyeksi vertikal yang hilang. Akan tetapi, jika terlalu banyak detil histogram proyeksi yang hilang, entitas-entitas kecil karakter yang direpresentasikan oleh “bukit-bukit” kecil histogram juga hilang. Kejadian ini dapat menyebabkan entitas-entitas karakter akan makin kabur separasinya, yang akhirnya memunculkan adanya kesalahan segmentasi. Lihat contoh Gambar 7(y) yang memperlihatkan adanya entitas karakter ‘r’ dan ‘a’ yang berada pada satu segmen.

Sebagaimana terlihat pada Tabel 2, dimulai dari lebar jendela 8 titik, bila kelebaran dan nilai α -nya makin naik, tidak mengindikasikan adanya kesalahan segmentasi. Ini berarti, dengan jendela yang makin lebar dan juga nilai α yang makin tinggi, akan membuat kemampuan lokalisasi jendela akan tidak terlalu tinggi dan juga tidak terlalu rendah. Akibatnya, tidak akan terlalu banyak atau pun terlalu sedikit detil histogram proyeksi vertikal yang hilang. Hal ini menyebabkan dapat dilestarikannya bukit-bukit kecil histogram proyeksi vertikal, yang merepresentasikan entitas-entitas karakter. Dengan demikian, kesalahan segmentasi dapat dihindari.



Gambar 6. Langkah-langkah segmentasi.



Gambar 7. Hasil Segmentasi untuk Berbagai Ukuran Jendela *Blackman*; (a) – (f) Lebar Jendela 4 Titik dengan Nilai α Masing-Masing Adalah 0, 0,04 , 0,08 , 0,12 , 0,16 dan 0,20; (g) – (l) Lebar Jendela 8 Titik dengan Nilai α Masing-Masing Adalah 0, 0,04 , 0,08 , 0,12 , 0,16 dan 0,20; (m) – (r) Lebar Jendela 12 Titik dengan Nilai α Masing-Masing Adalah 0, 0,04 , 0,08 , 0,12 , 0,16 dan 0,20; (s) – (x) Lebar Jendela 16 Titik dengan Nilai α Masing-Masing Adalah 0, 0,04 , 0,08 , 0,12 , 0,16 dan 0,20; (y) – (ad) Lebar Jendela 20 Titik dengan Nilai α Masing-Masing Adalah 0, 0,04 , 0,08 , 0,12 , 0,16 dan 0,20. (Catatan: Sumbu Tegak dan Horizontal Dinyatakan dalam Piksel)

Indikasi keberhasilan segmentasi pada berbagai ukuran jendela *Blackman* pada Tabel II merupakan hasil yang didapatkan secara subyektif, karena hanya menggunakan sebuah ragam penulisan kata “citra”. Untuk mendapatkan hasil secara obyektif, maka perlu menggunakan lebih banyak lagi ragam penulisan kata “citra” dan juga untuk kata-kata yang lain selain “citra”. Namun, bila pengamatan indikasi keberhasilan segmentasi dilakukan secara visual, hal ini menjadi tidak praktis, karena harus mengamati satu demi satu indikasi keberhasilan segmentasinya. Untuk itu, supaya lebih praktis, hal tersebut tidak dilakukan secara visual namun dengan mengamati tingkat pengenalan dari keseluruhan pengenalan kata tulisan tangan. Dengan menggunakan cara ini diasumsikan, pada tingkat pengenalan yang tertinggi, kesalahan segmentasinya paling sedikit.

Sebagaimana terlihat pada Gambar 8(a), suatu contoh jendela *Blackman* 12 titik dengan nilai $\alpha = 0,12$ (yang pada Tabel 2 secara subyektif dinyatakan “bebas” dari kesalahan segmentasi), dapat juga digunakan secara efektif untuk mensegmentasi 10 ragam penulisan kata “citra”. Namun, ada suatu hal perlu diingat yaitu bahwa jendela *Blackman* tersebut masih perlu dievaluasi lebih lanjut keefektifannya dalam mensegmentasi lebih banyak lagi ragam penulisan kata “citra” dan juga untuk kata-kata yang lain.

Bilamana Gambar 8(a) dan 8(b) dibandingkan, terlihat adanya unjuk kerja jendela *Blackman* 12 titik dengan $\alpha = 0,12$ yang sedikit lebih baik daripada tapis Gaussian 16 titik dengan deviasi standar 4 [6], karena memberikan jumlah segmen yang sedikit lebih kecil. Pada Gambar 8, perbandingan gambar (a9) dengan (b9) serta (a10) dengan (b10), memperlihatkan bahwa penggunaan jendela *Blackman* masing-masing memberikan satu dan dua segmen lebih sedikit daripada tapis Gaussian. Ini berarti jendela *Blackman* tersebut lebih mampu melestarikan “bukit-bukit” kecil pada histogram proyeksi vertikal, yang merepresentasikan entitas-entitas karakternya.

V. SIMPULAN DAN SARAN

Berdasarkan hal-hal yang telah disampaikan di atas, dapat disimpulkan hal-hal berikut.

- Berdasar pengujian secara subyektif terlihat bahwa, semua jendela *Blackman* yang lebarnya 8, 12, dan 16 titik dengan nilai α masing-masing $\alpha \geq 0$, $\alpha \geq 0,12$, dan $\alpha \geq 0,40$ dapat digunakan secara efektif untuk keperluan segmentasi lebih.
- Secara umum, jendela *Blackman* dengan kelebaran mulai dari 8 titik, bila kelebaran dan nilai α -nya makin naik, dapat digunakan secara efektif untuk keperluan segmentasi lebih.



Gambar 8. Hasil-Hasil Segmentasi untuk Beragam Penulisan Kata “citra”; (a) Menggunakan Jendela *Blackman* 12 Titik dengan Nilai $\alpha=0,12$; (b) Menggunakan Tapis Gaussian 16 Titik dengan Deviasi Standar 4 [6]. (Catatan: Sumbu Tegak dan Horizontal Dinyatakan dalam Piksel)

- Berdasar pengujian secara subyektif, jendela *Blackman* 12 titik dengan nilai $\alpha = 0,12$ sedikit lebih baik daripada tapis Gaussian 16 titik dengan deviasi standar 4 [6], karena dapat memberikan hingga sejumlah 2 segmen lebih sedikit.

Berikut ini beberapa saran untuk mengevaluasi lebih jauh sistem segmentasi lebih yang menggunakan jendela *Blackman* ini.

- Untuk mendapatkan hasil yang lebih obyektif, perlu dilakukan pengujian terhadap lebih banyak lagi ragam penulisan kata “citra” dan juga untuk kata-kata yang lain.
- Pengamatan keberhasilan segmentasi pada lebih banyak ragam penulisan akan lebih praktis, bila dilaksanakan dengan cara mengamati hasil tingkat pengenalan dari keseluruhan pengenalan kata tulisan tangan.

DAFTAR PUSTAKA

[1] K. Sayre, “Machine Recognition of Handwritten Words: A Project Report”, *Pattern Recognition*, vol. 5, no. 3, hal. 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*, hal. 448-459, 2003.
 [3] R. M. Bozinovic dan S.N. Srihari, “Off-Line Cursive Word Recognition”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, hal. 68 – 83, 1989

[4] A. W. Senior, *Off-line Cursive Handwriting Recognition using Recurrent Neural Networks*, PhD Thesis, Cambridge: Cambridge University Engineering Department, 1994.
 [5] L. Sumarno, “Segmentasi Kata Tulisan Tangan Menggunakan Wavelet”, *Compile*, vol. 1, no. 1, hal. 80-87, 2008.
 [6] L. Sumarno, A. Susanto, T. Sriwidodo, dan J.E. Istiyanto, “Segmentasi Kata Tulisan Tangan Menggunakan Tapis Gaussian 1D”, *Sigma*, vol. 12, no. 2, hal. 179-189, 2009.
 [7] A.V. Oppenheim, dan R.W. Schaffer, *Discrete-Time Signal Processing*. Upper Saddle River, NJ: Prentice-Hall, 1999, hal. 468-471.
 [8] M. Misiti, Y. Misiti, G. Oppenheim, dan 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, hal. 323-326, 2001.