

DETEKSI TULISAN TANGAN MENGGUNAKAN METODE SEGMENTASI ADAPTIF DAN *HIDDEN MARKOV MODEL*

Akalily Mardhiyya¹, Bambang Hidayat², Suci Aulia³

^{1,2,3}Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom
^{1,2,3}Jl. Telekomunikasi No.1 Ters Buah Batu, Bandung 40257, Indonesia

¹akalilymardhiyya2@gmail.com ²bhidayat@telkomuniversity.ac.id

³suciaulia@telkomuniversity.ac.id

Abstract—In previous research, handwriting detection in [8] has been purposed for detecting handwriting letter and number using Hidden Markov Model (HMM) as classification method and Modified Direction Feature as extraction feature method and obtain accuration rate of 74,72%. The system is limited for only handwriting which each of character is sepearated. This research has purposed a system which can recognize handwriting with or without overlapping to each other but still separated. In this system, adaptive segmentation is used so as to separate the letters or numbers that overlap. The input of system is a scanned images of handwriting for training data and test data that contains letter and numbers. The early step of this system is pre-processing for training data and test data, then the data extracted by the MDF in order to get the characteristics of the image. After acquired traits, the image then classified by the Hidden Markov Model (HMM). System output as a text format in .txt. From the tests, the designed system can achive an average accuracy of 70,44% with an average computation time of 2,07 second. the system is still not perfect because there is still accuracy of 20% for character 0 and 0% for word of E22.

Keywords-handwriting, overlap, Modified Direction Feature, Hidden Markov Model, adaptive segmentation

Abstrak—Pada penelitian sebelumnya telah dilakukan deteksi tulisan tangan dengan akurasi mencapai 74,72%[8]. *Hidden Markov Model* (HMM) digunakan sebagai metode klasifikasi, sedangkan untuk metode ekstraksi cirinya digunakan *Modified Direction Feature* (MDF). Sistem terbatas pada tulisan tangan yang antarkarakter saling terpisah. Pada penelitian ini, dirancang suatu sistem yang dapat mengenali karakter huruf dan angka pada tulisan tangan dengan atau tanpa *overlapping* tetapi tidak bersentuhan antarkarakter. Pada sistem ini, digunakan segmentasi adaptif sehingga dapat memisahkan huruf maupun angka yang saling *overlap*. Masukan sistem merupakan hasil *scan* dari data latih dan data uji yang berisi karakter huruf ataupun angka pada tulisan tangan. Tahap awal pada sistem ini adalah *pre-processing* data latih dan data uji. Kemudian data tersebut diekstraksi cirinya dengan *Modified Direction Feature* (MDF) sehingga didapat ciri-ciri dari citra. Setelah ciri didapat, selanjutnya diklasifikasi dengan *Hidden Markov Model* (HMM). *Output* sistem berupa teks dengan format (.txt). Dari pengujian yang dilakukan, sistem yang dirancang menghasilkan akurasi rata-rata 70,44% dengan waktu komputasi rata-rata sebesar 2,07 detik. Sistem masih belum dikatakan sempurna karena masih terdapat akurasi 20% untuk karakter 0 dan 0% untuk kata E22.

Kata kunci-tulisan tangan, overlap, Modified Direction Feature, Hidden Markov Model, segmentasi adaptif

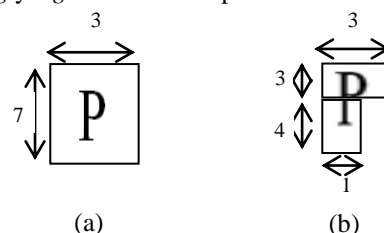
I. PENDAHULUAN

Pada penelitian sebelumnya telah dilakukan deteksi tulisan tangan dengan akurasi mencapai 74,72%[8]. *Hidden Markov Model* (HMM) digunakan sebagai metode klasifikasi, sedangkan untuk metode ekstraksi cirinya digunakan *Modified Direction Feature* (MDF). Sistem terbatas pada tulisan tangan yang antarkarakter terpisah satu sama lain. Pada penelitian ini, dirancang suatu sistem yang dapat mengenali karakter huruf maupun angka yang terdapat *overlap* antarkarakter, dengan mengacu pada penelitian sebelumnya. Metode ekstraksi ciri yang digunakan pada penelitian ini mengacu pada penelitian sebelumnya yaitu metode *Modified Direction Feature* (MDF), begitu pula dengan metode klasifikasi yang digunakan, yaitu metode *Hidden Markov Model* (HMM). Untuk proses segmentasinya digunakan segmentasi yang dapat memisahkan karakter yang saling *overlap* (dua karakter yang berada pada kolom citra yang sama).

II. DASAR TEORI

A. Segmentasi Adaptif

Segmentasi adaptif adalah metode pemisahan citra menjadi bagian-bagian pembentuknya dengan kemampuan dapat menyesuaikan kondisi citra, dalam penelitian ini kemampuan memisahkan dengan tepat setiap karakter pada citra tulisan tangan yang *overlap* antarkarakter. Metode yang digunakan adalah *labeling*. Pada penelitian ini dilakukan beberapa kali proses *labeling* yang diilustrasikan pada Gambar 1.






Gambar 1. (a) Citra huruf P dengan ukuran 3x7 (b) Pembagian citra untuk dilakukan *labeling*

Citra huruf P dengan ukuran 3x7 dibagi menjadi dua bagian, bagian pertama adalah citra bagian atas dengan ukuran 3x3 dan bagian kedua adalah citra bagian bawah dengan ukuran 1x4. *Labeling* pada huruf P ini dilakukan sebanyak tiga kali. *Labeling* pertama untuk citra bagian atas. *Labeling* kedua untuk citra bagian bawah. Setelah itu dilakukan pelabelan lagi untuk menggabungkan kedua *label* tersebut.

B. *Modified Direction Feature*^{[2] [5] [8]}

Modified Direction Feature merupakan metode ekstraksi ciri dengan menghitung nilai *Location transition* dan *Direction transition*. Nilai *Location transition* didapat dengan menghitung posisi transisi dan jumlah transisi pada bidang vertikal dan horizontal dari suatu citra. Transisi adalah posisi dimana terjadinya perubahan *pixel* dari *background* menjadi *foreground* tetapi tidak sebaliknya. Sedangkan nilai *Direction transition* didapat berdasarkan label arah dari sebuah *pixel*, setiap *pixel* bernilai 1 (*foreground*) pada citra memiliki arah tersendiri seperti pada Tabel I.

Tabel I. Nilai Label dan Arah pada DF

| Arah | Nilai | Bentuk |
|----------------|-------|-----------------------------------------------------------------------------------|
| Vertikal | 2 | |
| Diagonal Kanan | 3 |  |
| Horizontal | 4 |  |
| Diagonal Kiri | 5 |  |

Untuk melakukan pelabelan arah pada masing masing *pixel* dapat dilakukan dengan cara sebagai berikut:

1. Lakukan pemindaian dari kiri atas ke kanan bawah.
2. Apabila menemukan sebuah *pixel foreground* (*pixel* bernilai 1) maka periksa *pixel* tetangga dari *pixel* tersebut.
3. Pada Tabel II, X adalah *pixel* yang akan diperiksa, kemudian periksa a – h. Apabila pada posisi tetangga dari a sampai h ditemukan *pixel foreground* (*pixel* bernilai 1), maka nilai X memiliki nilai arah berdasarkan aturan dibawah ini:
 - Jika pada posisi a atau e maka nilai arah adalah 5
 - Jika pada posisi b atau f maka nilai arah adalah 2
 - Jika pada posisi c atau g maka nilai arah adalah 3
 - Jika pada posisi d atau h maka nilai arah adalah 4

Tabel II. Matrik Ketetangaan Dalam Penentuan Nilai Label

| | | |
|---|---|---|
| a | b | c |
| h | X | d |
| g | f | e |

Berikut ini adalah prosedur untuk mendapatkan ciri dengan MDF.

a) Menentukan Nilai Transisi

Untuk mendapatkan nilai transisi dilakukan pemindaian dari kiri ke kanan, kanan ke kiri, atas ke bawah, dan bawah ke atas. Nilai transisi (TF) adalah nilai dari pembagian antara posisi terjadinya transisi dengan panjang atau lebar dari citra. Apabila pemindaian

dilakukan dari kiri ke kanan dan dari atas ke bawah, maka perhitungan nilai transisi sebagai berikut:

$$LT_i = 1 - \frac{x_i}{max} \tag{1}$$

Sebaliknya, apabila pemindaian dilakukan dari kanan ke kiri dan dari bawah ke atas, maka perhitungan nilai transisi sebagai berikut:

$$LT_i = \frac{x_i}{max} \tag{2}$$

Dengan x_i adalah posisi adanya transisi dan max adalah panjang citra jika pemindaian dilakukan dari atas ke bawah atau bawah ke atas, jika pemindaian dilakukan dari kanan ke kiri atau kiri ke kanan maka max adalah lebar citra.

b) Menentukan Nilai Arah

DF didapat dari pembagian label arah pada posisi terjadinya transisi dengan nilai pembagi. Pada penelitian ini nilai pembagi yang digunakan adalah 10 seperti yang tercantum dalam referensi. Apabila jumlah transisi yang ditemukan kurang dari jumlah transisi maksimal yang digunakan maka DF sisanya diberikan nilai 0 dan apabila jumlah transisi yang ditemukan lebih dari jumlah transisi maksimal yang digunakan maka tidak akan dihitung.

$$DF = \frac{\text{nilai arah}}{10} \tag{3}$$

Setelah semua nilai DF dan TF dari 4 arah dicari maka akan dilanjutkan dengan melakukan normalisasi vektor ciri yang didapat pada setiap arah pencarian dengan dimensi vektor ciri sebagai berikut:

$$\frac{\text{Jumlah_transisi} \times \text{jumlah_feature} \times \text{jumlah_vektor}}{\text{lebar_matriks_normalisasi}} \tag{4}$$

C. *Hidden Markov Model*^{[3] [8]}

HMM disebut *hidden* karena model pada HMM tersembunyi, tetapi parameter model diketahui. HMM memiliki kondisi tersembunyi yang tidak dapat diobservasi sehingga yang diobservasi adalah *output* pada setiap kondisi (*state*). Elemen-elemen yang terdapat pada HMM yaitu:

1. N, menyatakan jumlah *state*, dengan ruang *state* $S = \{S_1, S_2, \dots, S_N\}$ dan *state* pada waktu t dinyatakan dengan Q_t .
2. M, menyatakan jumlah pengamatan (observasi) tiap *state*, dengan simbol observasi $V = \{V_1, V_2, \dots, V_M\}$.
3. A, menyatakan distribusi probabilitas transisi antar *state*. $A = [a_{ij}]$ menyatakan probabilitas transisi dari *state* i ke *state* j dimana:

$$a_{ij} = P[Q_{t+1} = S_j | Q_t = S_i], 1 \leq i, j \leq N \tag{5}$$
4. B, menyatakan distribusi probabilitas simbol observasi dalam *state*. $B = \{b_j(k)\}$ menyatakan distribusi probabilitas simbol observasi ke- k pada *state* j dimana:

$$b_j(k) = P[V_k \text{ saat } t | q_t = S_j], 1 \leq k \leq M; j = 1, 2, 3, \dots, N \tag{6}$$
5. π , menyatakan distribusi probabilitas *initial state*, $\pi = \{\pi_i\}$ menyatakan distribusi probabilitas inisialisasi pada *state* i dimana:

$$\pi_i = P[Q_1 = S_i], 1 \leq i \leq N \tag{7}$$

a. Permasalahan pada HMM [4]

Agar HMM dapat diaplikasikan ke berbagai permasalahan yang nyata, terdapat tiga permasalahan pada HMM yang harus diselesaikan, yaitu:

1. *Evaluation problem*

Permasalahan pertama pada HMM yang harus diselesaikan adalah mencari $P(O|\lambda)$ atau peluang barisan observasi $O = \{O_1, O_2, \dots, O_T\}$ jika diberikan model HMM $\lambda = (A, B, \pi)$. Peluang ini dapat ditentukan secara induksi dengan algoritma *forward* dan algoritma *backward*.

2. *Decoding Problem*

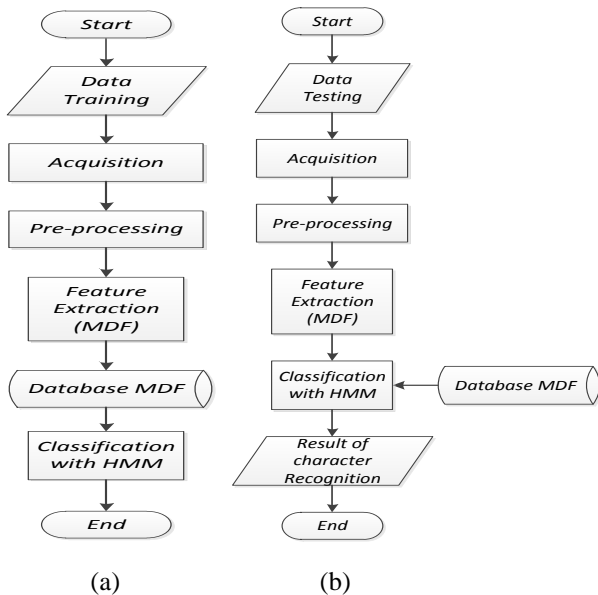
Permasalahan kedua pada HMM yaitu pencarian *state* yang optimal $Q^* = \{Q^*_1, Q^*_2, \dots, Q^*_T\}$ jika diberikan barisan observasi $O = \{O_1, O_2, \dots, O_T\}$ dan model $\lambda = (A, B, \pi)$. Untuk menyelesaikan *decoding problem* digunakan algoritma Viterbi.

3. *Learning problem*

Permasalahan ketiga pada HMM yaitu bagaimana mengatur parameter model $\lambda = (A, B, \pi)$ jika diberikan barisan observasi $O = O_1, O_2, \dots, O_T$ agar $P(O|\lambda)$ maksimum. Untuk menyelesaikan permasalahan ini digunakan algoritma *Baum-Welch*.

III. PEMBAHASAN

A. Perancangan Sistem



Gambar 2. Diagram Alir (a) Prose latihan dan (b) Proses uji

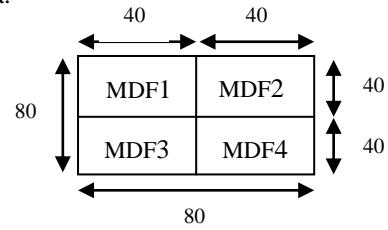
A. Analisis Hasil Simulasi Sistem

Pengujian terhadap sistem dilakukan dengan beberapa skenario yang telah ditentukan, yaitu mengubah parameter pada proses MDF, HMM, dan *pre-processing* untuk mendapatkan akurasi terbaik sistem.

a. Pengujian Parameter Proses Ekstraksi Ciri MDF

Pada penelitian ini proses ekstraksi ciri MDF dibagi menjadi 4 bagian, yaitu bagian atas kiri, bagian atas kanan, bagian bawah kiri, dan bagian bawah kanan dari citra berukuran 80x80 seperti pada Gambar 3, agar ciri

semakin mewakili karakter. Hal tersebut menyebabkan vektor ciri yang dihasilkan menjadi 4 kali dari sebelumnya.

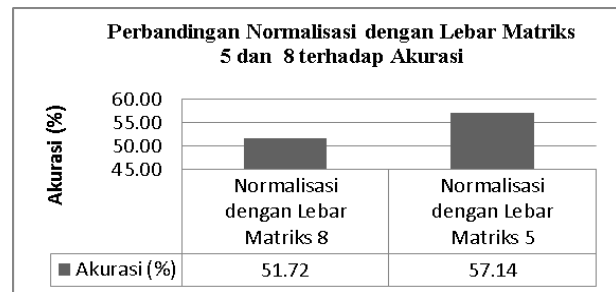


Gambar 3. Pembagian MDF Menjadi 4 Bagian

Pada skenario ini pengujian dilakukan terhadap 30 kata tulisan tangan. Parameter MDF yang diuji adalah lebar matriks normalisasi. Pengujian dilakukan dengan parameter HMM yang digunakan adalah jumlah *cluster* 128 dan jumlah *state* 50. Lebar matriks normalisasi MDF yang diuji yaitu 5 dan 8.

Tabel III. Pengaruh perubahan lebar matriks normalisasi pada MDF terhadap akurasi

| Normalisasi dengan lebar matriks 8 | | Normalisasi dengan lebar matriks 5 | |
|------------------------------------|-------------|------------------------------------|-------------|
| Karakter benar | Akurasi (%) | Karakter benar | Akurasi (%) |
| 105 | 51.72 | 116 | 57.14 |



Gambar 4 Pengaruh perubahan lebar matriks normalisasi pada MDF terhadap akurasi

Dapat dilihat pada Tabel III dan Gambar 4 bahwa akurasi terbaik diperoleh saat normalisasi dengan lebar matriks 5 yaitu sebesar 57.14%. Hal tersebut disebabkan saat normalisasi dengan lebar matriks 5, dihasilkan vektor ciri sebanyak 120, sesuai dengan persamaan (4) dengan jumlah_transisi= 3, jumlah_feature= 2 (LT dan DT), jumlah_vektor= 4 (kiri, atas, kanan, bawah) dan lebar_matriks_normalisasi= 5. Ciri yang didapat dari matriks normalisasi dengan lebar 5 sudah cukup mewakili citra karakter. Apabila jumlah vektor ciri yang dihasilkan terlalu sedikit, khawatir ciri kurang mewakili citra karakter.

b. Pengujian Parameter Proses Klasifikasi HMM

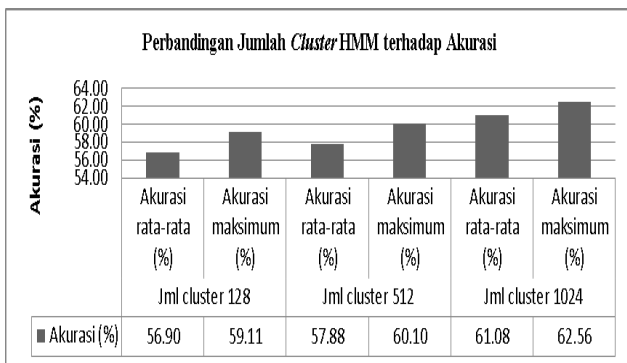
Setelah didapat parameter terbaik dari MDF (Normalisasi matriks dengan lebar 5), sistem dilakukan pengujian parameter HMM terhadap akurasi. Parameter HMM yang diuji antara lain jumlah *cluster* dan jumlah *state*.

1. Pengujian Parameter Jumlah Cluster pada Proses Klasifikasi HMM

Jumlah cluster yang akan diuji pada pengujian ini antara lain 128, 512, dan 1024. Untuk jumlah state yang digunakan adalah 50.

Tabel IV Pengaruh jumlah cluster pada HMM terhadap akurasi

| Jml cluster 128 | | Jml cluster 512 | | Jml cluster 1024 | |
|-----------------------|----------------------|-----------------------|----------------------|-----------------------|----------------------|
| Akurasi rata-rata (%) | Akurasi maksimum (%) | Akurasi rata-rata (%) | Akurasi maksimum (%) | Akurasi rata-rata (%) | Akurasi maksimum (%) |
| 56.90 | 59.11 | 57.88 | 60.10 | 61.08 | 62.56 |



Gambar 5 Pengaruh perubahan jumlah cluster pada HMM terhadap akurasi

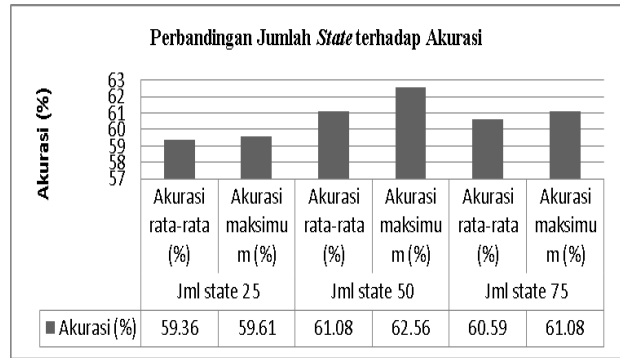
Dapat dilihat pada Tabel IV dan Gambar 5 bahwa dengan jumlah cluster sebanyak 1024 diperoleh akurasi terbaik, yaitu akurasi rata-rata sebesar 61,08% dan akurasi maksimum sebesar 62,56%. Semakin banyak jumlah cluster maka ciri yang terdapat pada suatu cluster hanya yang memiliki tingkat kemiripan yang besar sehingga akurasi yang didapat semakin besar.

2. Pengujian Parameter Jumlah State pada Proses Klasifikasi HMM

Pada pengujian ini akan diuji parameter jumlah state 25, 50 dan 75 terhadap akurasi sistem dengan parameter terbaik MDF (normalisasi matriks dengan lebar 5) dan jumlah cluster HMM yang menghasilkan akurasi terbaik(1024).

Tabel V Pengaruh Jumlah State pada HMM terhadap akurasi

| Jumlah state 25 | | Jumlah state 50 | | Jumlah state 75 | |
|-----------------------|----------------------|-----------------------|----------------------|-----------------------|----------------------|
| Akurasi rata-rata (%) | Akurasi maksimum (%) | Akurasi rata-rata (%) | Akurasi maksimum (%) | Akurasi rata-rata (%) | Akurasi maksimum (%) |
| 59.36 | 59.61 | 61.08 | 62.56 | 60.59 | 61.08 |



Gambar 6 Pengaruh perubahan jumlah state pada HMM terhadap akurasi

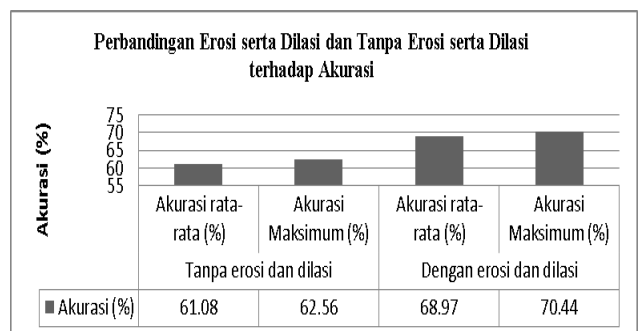
Dapat dilihat pada Tabel V dan Gambar 6 bahwa akurasi maksimum diperoleh saat jumlah state 50, yaitu sebesar 62,56%. Bertambahnya jumlah state mempengaruhi kondisi sistem yang akan dikenali.

c. Pengujian Parameter Proses Pre-processing

Pengujian dilakukan dengan membandingkan akurasi sistem yang dihasilkan saat dilakukan proses erosi serta dilasi dan tanpa proses erosi serta dilasi. Erosi adalah pengecilan atau penipisan objek citra biner, sedangkan dilasi adalah penebalan dalam citra biner. Pengujian dengan erosi dan dilasi pada pre-processing dilakukan sebelum tahap segmentasi. Digunakan parameter MDF dan HMM terbaik, yaitu lebar matriks normalisasi MDF 5, jumlah cluster HMM 1024 dan jumlah state HMM 50.

Tabel VI Pengaruh erosi dan dilasi terhadap akurasi sistem

| Tanpa erosi dan dilasi | | Dengan erosi dan dilasi | |
|------------------------|----------------------|-------------------------|----------------------|
| Akurasi rata-rata (%) | Akurasi Maksimum (%) | Akurasi rata-rata (%) | Akurasi Maksimum (%) |
| 61.08 | 62.56 | 68.97 | 70.44 |



Gambar 7 Pengaruh Erosi dan Dilasi pada Pre-processing terhadap Akurasi Sistem

Dapat dilihat pada Tabel VI dan Gambar 7 bahwa dengan proses erosi dan dilasi pada pre-processing dapat meningkatkan akurasi sistem. Hal tersebut disebabkan erosi berperan menyaring informasi dari citra dengan menghilangkan detail yang lebih kecil dari strel, sedangkan dilasi menyebabkan lubang-lubang yang terdapat di tengah objek akan mengecil sehingga dengan adanya erosi dan dilasi, informasi citra yang

diambil lebih sesuai dengan citra karakter yang diwakilkan.

d. Hasil Terbaik

Berdasarkan percobaan yang dilakukan terhadap semua kemungkinan dari kombinasi parameter yang diberikan, hasil terbaik diberikan oleh percobaan dengan parameter berikut :

- a. Ekstraksi ciri MDF : Normalisasi dengan lebar matriks 5
- b. Klasifikasi HMM :
 - i. Jumlah *cluster* : 1024
 - ii. Jumlah *state* : 50
- c. Pre-processing : dengan proses erosi dan dilasi

Akurasi terbaik pada data uji total menghasilkan nilai akurasi sebesar 70,44%. Selain akurasi, waktu komputasi dapat dijadikan tolak ukur performansi sistem. Waktu komputasi rata-rata saat dihasilkan akurasi terbaik adalah 2,07 detik.

IV. KESIMPULAN

Sistem deteksi tulisan tangan yang dibuat mampu mengenali karakter dengan akurasi terbaik sebesar 100% untuk karakter B/b, C/c, D/d, E/e, J/j, N/n, W/w, Y/y, dan Z/z serta mampu mengenali kata dengan akurasi terbaik sebesar 100% untuk kata LATIFAH dan AHMAD. Secara keseluruhan diperoleh akurasi terbaik sebesar 70.44% dengan parameter yang digunakan, yaitu dengan proses dilasi dan erosi pada pre-processsing, Normalisasi MDF dengan lebar matriks 5, jumlah *cluster* HMM sebanyak 1024, dan jumlah *state* sebanyak 50. Waktu

komputasi rata-rata secara keseluruhan dengan parameter yang memberikan akurasi terbaik adalah 2.07 detik.

DAFTAR PUSTAKA

- [1] (n.d.). Retrieved Maret 25, 2014, from www.willymanoppo.blogspot.com/2013/12/makalah-pengolahan-citra.html
- [2] Blumenstein, M., Liu, X. Y., & Verma, B. (n.d.). *A Modified Direction Feature for Cursive Character Recognition*. Australia.
- [3] Fink, G. A. (2007). *Markov Models for Pattern Recognition*. Dortmund: University of Dortmund.
- [4] Firdaniza, N. G., & Akmal. (2006). Hidden Markov Model. *Seminar Nasional Matematika dan Pendidikan Matematika*. Universitas Padjadjaran.
- [5] PERDANA, G. R. (2013). *Analisis dan Implementasi Optical Character Recognition Menggunakan*. Bandung: Institut Teknologi Telkom.
- [6] Purnomo, M. H., & Muntasa, A. (2010). *Konsep Pengolahan Citra Digital dan Ekstraksi Fitur*. Yogyakarta: Graha Ilmu.
- [7] Sutoyo, T., Mulyanto, E., Suhartono, V., & al, e. (2009). *Teori Pengolahan Citra Digital*. Yogyakarta: Andi.
- [8] Yuwitaning, E. F. (2014). *Implementasi Metode Hidden Markov Model (HMM) untuk Deteksi Tulisan Tangan*. Bandung: Telkom University.