

Verifikasi Tanda Tangan *Offline* Melalui Pencocokan Graf (*Graph Matching*)

Dody Dharma¹⁾

1) Jurusan Teknik Informatika ITB, Bandung 40132, email: dody-dharma@students.itb.ac.id

Abstrak – Makalah ini membahas sebuah metode verifikasi (uji keabsahan) tanda tangan yang sederhana dan efektif yang hanya bergantung pada intensitas pixel pada gambar raw binary dan menghindari penggunaan cara-cara rumit yang sering ditampilkan pada literatur-literatur. Metode ini memandang masalah verifikasi tanda tangan sebagai sebuah masalah pencocokan graf (*graph matching*).

Metode ini diuji dengan memanfaatkan tanda tangan asli dan tiruan/palsu yang dihasilkan oleh 5 orang subjek. Tingkat kesalahan sebesar 26,7% dicapai saat terdapat pemalsuan terampil (*skilled_forgeries*) dan 5,6% dicapai saat terdapat pemalsuan acak (*random_forgeries*). Sebuah nilai positif dari algoritma yang dipergunakan dalam metode ini adalah bahwa tingkat toleransi kesalahan dari pemalsuan random menjadi nol ketika Tingkat Kesalahan Penolakan (*false rejection rate*), *FRR*, sama dengan Tingkat Kesalahan Penyetujuan Tanda Tangan dari Pemalsuan Terampil (*skilled_forgeries*), *FAR_skilled*. Dengan mempertahankan ukuran kotak normalisasi pada 32 x 64 pixel membuat waktu verifikasi yang dibutuhkan hanya berkisar 2 detik.

Kata Kunci: Verifikasi Tanda Tangan (*Offline Signature Verification*), pencocokan graf (*graph matching*), Perampingan (*thinning*), Normalisasi (*Normalization*).

1. PENDAHULUAN

Kebutuhan untuk memastikan bahwa hanya orang yang tepat/benar yang berhak mendapat akses pada sistem dengan tingkat keamanan tinggi telah membawa kepada pengembangan dari sistem verifikasi personal otomatis (*automatic personal verification*). Tanda tangan, sidikjari, suara, dan tulisan tangan semuanya telah dipergunakan untuk menguji keabsahan identitas personal. Diantar semua itu, tanda tangan memiliki sebuah kelebihan yang sangat mendasar bahwa ia merupakan cara yang paling sering dipakai dalam mengidentifikasi seseorang dalam operasi sehari-hari seperti transaksi perbankan otomatis, transfer dana elektronik, analisis dokumen, dan kontrol akses.

Sebuah system verifikasi tanda tangan haruslah mampu mendeteksi pemalsuan dan pada waktu yang sama juga mampu mengurangi penolakan terhadap tanda

tangan asli. Permasalahan verifikasi tanda tangan dapat dikelompokkan kedalam kategori: offline dan online. Verifikasi tandatangan offline tidak mempergunakan informasi dinamik yang dipergunakan secara luas dalam sistem verifikasi tanda tangan online. Dalam makalah ini, yang akan diinvestigasi adalah masalah verifikasi tanda tangan secara offline. Untuk selanjutnya, agar lebih sederhana, verifikasi tanda tangan offline kita sebut saja verifikasi tanda tangan.

Persoalann verifikasi tanda tangan yang dihadapi menitik beratkan pada 3 tipe dari bentuk pemalsuan: Pemalsuan acak (*random_forgeries*), dihasilkan tnpa mengetahui nama penanda tangan maupun bentuk dari tanda tangannya; pemalsuan sederhana (*simple_forgeries*), dihasilkan oleh orang-orang yang mengetahui nama dari penanda tangan asli tanpa memiliki contoh dari tanda tangan slinya; Pemalsuan terampil (*skilled_forgeries*) dihasilkan oleh mereka yang melihat langsung tanda tangan aslinya, kemudian mencoba untuk mengimitasinya semirip mungkin. Dapat terlihat dengan mudah bahwa permasalahan verifikasi tanda tangan menjadi lebih dan lebih sulit ketika dihadapkan pada tahap kasus pemalsuanacak – pemalsuan sederhana, dan pemalsuan terampil, permasalahan semakin bertambah sulit saja ketika dihadapkan pada kasus dimana penanda tangan yang asli melakukan kesalahan dalam penanda tangan dalam beberapa kasus. Dalam kenyataannya, berlatih meniru sebuah tanda tangan sering membuat kita sulit sekali membedakan antara tanda tanga palsu dan yang asli, karena keduanya terlihat begitu identik; dalam banyak kasus proses pembedaan asli-palsu menjadi semakin sulit ketika beberapa penanda tangan asli memberikan variasi yang luas dalam menuliskan tanda tangan mereka.

Dalam makalah ini, kami menawarkan sebuah cara verifikasi yang sangat sederhana yang hanya bergantung kepada intensitas pixel dari *raw binary*. Metode ini memandang persoalan verifikasi tanda tangan sebagai kasus pencocokan graf. Bagian-bagian selanjutnya diorganisir sebagai berikut. Bagian 2 berisi survei terhadap bidang-bidang yang berbasis pencocokan graf. Pemrosesan awal tanda tangan dijelaskan pada Bagian 3. Verifikasi tanda tangan dirumuskan pada bagian 4. Dan hasilnya dilaporkan pada bagian 5. Terakhir, simpulan makalah ini paparkan pada bagian 6.

2. APLIKASI-APLIKASI YANG BERBASIS PENCOCOKAN GRAF (*GRAPH MATCHING*)

Banyak sekali aplikasi dari pencocokan graf. Berikut ini adalah beberapa survei dari beberapa hal terkait. Misalnya aplikasi graf dalam proses Pengenalan Wajah (*face recognition*) melalui pencocokan graf dengan sebuah multi-lapisan model gramatikal wajah . kemudian juga ada sistem yang bertugas mengatasi masalah pencahayaan dalam proses pengenalan wajah dengan mempergunakan algoritma "*Retinex dan Color Constancy*". Pendekatan yang dilakukan oleh algoritma ini melalui suatu pendekatan yang disebut "*elastic bunc graph matching*". Dan masih banyak lagi bentuk implementasi dari teori graf dalam mempermudah operasi dalam kehidupan sehari-hari, yang mungkin tidak perlu kami sebutkan disini secara satu persatu.

Intinya adalah bahwa permasalahan pencocokan (*matching*) adalah sebuah graf-subgraf isomorfisma dan algoritma dalam mengeksplorasi informasi semantik tentang simpul-simpul tanpa memerlukan informasi tentang topologi dari graf yang hendak dicocokkan

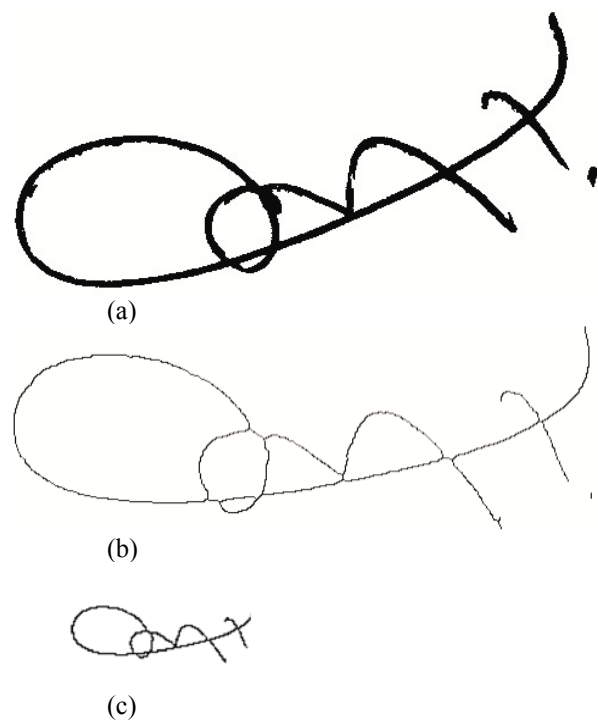
3. PEMROSESAN AWAL TANDA TANGAN (*SIGNATURE PREPROCESSING*)

Sebelum sebuah tanda tangan dapat dibandingkan dengan tanda tangan yang lain, perlu dilakukan beberapa operasi pemrosesan awal (*signature preprocessing*). Sebuah tanda tangan ditangkap sebagai sebuah gambar binari (*bunary image*), S . Kemudian, titik-titik *noise* (*pepper noise*) disekitar gambar dibuang, untuk memperoleh akurasi perhitungan yang lebih baik disekitar pusat area gambar. Kemudian Sudut, θ , dari bagian dari tanda tangan yang memiliki kecendrungan melengkung / berbelok terendah ditemukan . Sudut ini diukur berlawanan terhadap arah gerakan jarum jam dari axis-y. Tanda tangan diputar terhadap pusat area S sebesar θ derajat arah jarum jam. Rotasi ini mengeliminasi sudut pelerengan (*skew angle*) dari sebuah tanda tangan yang dibutuhkan untuk tanda tangan –tanda tangan dari subjek yang sama. Setelah rotasi, gambar tanda tangan tersebut diperhalus. Hal ini merupakan langkah diperlukan sebelum proses perampingan gambar (*thinning*). Untuk mengurangi data, gambar dirampingkan dengan mempergunakan *Zhang-Suen algorithm*. (tidak dibahas disini). Gambar yang telah dirampingkan memerlukan sebuah langkah normalisasi yang menjaga *aspect ratio* (proporsionalitas) dari tanda tangan tersebut. Himpunan pixel yang menyusun hasil akhir pemrosesan awal dari gambar S ini, dinotasikan sebagai himpunan noktah-noktah/simpul-simpul, X , yang merepresentasikan S . Langkah-langkah ini dirangkum sebagai berikut, dimana setelah setiap satu

pemrosesan, S digantikan oleh hasil pemrosesan tersebut.

1. Singkirkan titik-titik pengotor (*pepper noise*) dari S .
2. Cari Sudut dari pelerengan (bagian pada tanda tangan yang memiliki kecendrungan melengkung / berbelok terendah)(*angle of least second momment*), θ , dari S .
3. Putar S sejauh θ searah jarum jam.
4. Perhalus S .
5. Rampingkan S .
6. Normalisasi S . Himpunan pixel-pixel yang menyusun hasil akhir setelah normalisasi dari gambar S ini, disebut sebagai himpunan noktah-noktah/simpul-simpul yang merepresentasikan tanda tangan S .

Berikut adalah contoh ilustrasi dari pemrosesan awal tanda tangan tersebut. Gambar (a) memperlihatkan sebuah gambar binari dari sebuah tanda tangan asli. Gambar (b) diperoleh setelah setelah proses pembersihan *pepper noise*, pencarian sudut pelerengan θ , rotasi gambar sejauh θ , penghalusan, dan perampingan . Gambar (c) diperoleh setelah normalisasi gambar yang telah dirampingkan. Himpunan pixel-pixel X , dari gambar 3 merepresentasikan tanda tangan asli dari gambar 1.



Gambar 1. Pemrosesan awal tanda tangan : (a) Gambar asal. (b) gambar setelah penyingkiran *pepper noise*, eliminasi sudut pelerengan, penghalusan, perampingan, dan (c) hasil akhir setelah normalisasi 64 x 128 px.

4. PERBANDINGAN DUA TANDA TANGAN SEBAGAI PERSOALAN PERBANDINGAN KESETARAAN (PENCOCKAN) GRAF

Berdasarkan teori graf, terdapat beberapa definisi sebagai berikut :

Definisi 1. *Let V be a finite nonempty set of vertices, and let E be a set of unordered pairs of elementstaken from V . The pair (V, E) is then called an undirected graph on V , where V is the set of vertices and E is its set of edges. We write $G = (V, E)$ to denote such a graph.*

Definisi 2. *A graph $G = (V, E)$ is called bipartite if $V = X \cup Y$ with $X \cap Y = \emptyset$, and every edge of G is of the form $\{x, y\}$ with $x \in X$ and $y \in Y$. If every vertex in X is joined with every vertex in Y , we have a complete bipartite graph. In this case, if $|X| = m$, and $|Y| = n$, the graph is denoted $K_{m,n}$.*

Definisi 3. *Let $G = (V, E)$ be a bipartite graph as defined above. A matching in G is a subset of E such that no two edges share a common vertex in X or Y . A complete matching of X into Y is a matching in G such that every $x \in X$ is the end point of an edge.*

Untuk graf bipartit $G = (V, E)$ dengan V dipartisi sebagai $X \cup Y$, complete matching X terhadap Y mensyaratkan $|X| \leq |Y|$. Jika $|X|$ besar, maka konstruksi dari sebuah pencocokan tidak dapat dicapai hanya dengan melalui observasi atau dengan cara mencoba-coba. Teori berikut ini, memaparkan sebuah syarat cukup perlu dan cukup untuk adanya sebuah kesetaraan(pencocokan).

Teorema 1. *Let $G = (V, E)$ be a bipartite graph with V partitioned as $X \cup Y$. A complete matching of X into Y exists if and only if for every subset A of X , $|A| \leq |R(A)|$, where $R(A)$ is the subset of Y consisting of vertices each of which is adjacent to at least one vertex in A .*

Misalkan S_1 dan S_2 adalah dua buah gambar tanda tangan yang akan dibandingkan. X dan Y adalah himpunan dari simpul-simpul (pixel) yang merepresentasikan S_1 dan S_2 . Jelasnya, X dan Y adalah himpunan yang tidak beririsan. Kita bangun sebuah graf bipartit $G = (V, E) = K_{m,n}$, dari X dan Y dimana $V = X \cup Y$, $|X| = m$, dan $|Y| = n$. Karena G adalah graf bipartit lengkap dan asumsikan bahwa tanda tangan terurut sebagai $|X| \leq |Y|$, berdasarkan Teorema 1, ini mengindikasikan terdapat sebuah complete matching X terhadap Y . Biasanya terdapat sangat banyak complete matching seperti ini. Tujuan kita adalah mencari harga minimum complete

matching dari X ke Y . ini merupakan beberapa bentuk Permasalahan Pemetaan (Assignment Problem, AP) yang terkenal dalam teori graf. Kita akan mempergunakan algoritma yang disebut sebagai Metode Hungaria, untuk menyelesaikan permasalahan pemetaan; sebagai contoh, carilah seberapa besar kesamaan S_1 dan S_2 .

Kunci utama dalam mempergunakan Metode Hungaria untuk menyelesaikan permasalahan AP adalah dengan mencari harga matrix C yang merupakan matrix $m \times n$ dengan baris-barisnya berkorespondensi dengan simpul-simpul X dan kolom-kolomnya berkorespondensi dengan simpul-simpul Y . karena simpul-simpul X dan Y berasal dari pixel-pixel yang diambil dari gambar tanda tangan S_1 dan S_2 , setiap simpul $x \in X$ atau $y \in Y$ memiliki koordinat x dan y -nya masing-masing (kartesius). Koordinat-koordinat ini dipergunakan untuk mencari jarak antara x dan y setelah pengaturan pusat area gambar dari himpunan X dan Y . Maka, c_{xy} elemen dari C , $x \in X$, $y \in Y$, setara dengan jarak Euclidean antara x dan y . Ini merupakan harga pencocokan titik x dari tanda tangan S_1 terhadap titik y dari tanda tangan S_2 . Setelah menghitung semua elemen dari C , maka rumusan AP (Assignment Problem) telah terpecahkan. Harga, $cost_{min}$, dari solusi akhir setara dengan jumlah seluruh elemen, c_{xy} , yang berkorespondensi dengan harga solusi minimum. Harga ini di normalisasi dengan membaginya dengan $|X|$ yang menghasilkan harga normalisasi minimum per pixel. Kemudian lebih jauh lagi, $cost_{min}$, dibagi dengan sebuah factor, f , yang mengandung kolom-kolom dari C yang tidak dapat dicocokkan. Factor f mengukur persentase dari simpul-simpul yang mencocokkan X dan Y secara lengkap.

Untuk memeriksa keaslian sebuah tanda tangan uji, S , dilakukan perbandingan S tersebut dengan sejumlah, p buah, dari prototipe tanda tangan asli dari subjek yang sama melalui langkah-langkah sebagai berikut :

1. Dilakukan pemrosesan awal terhadap sejumlah p prototipe tanda tangan sebagaimana yang dijelaskan pada bagian 3 makalah ini untuk menghasilkan himpunan-himpunan dari simpul-simpul, Y_1, Y_2, \dots, Y_p .
2. Juga Dilakukan pemrosesan awal terhadap S sebagaimana yang dijelaskan pada bagian 3 makalah ini untuk menghasilkan sebuah himpunan simpul-simpul X .
3. Ambil $d = \infty$, dimana d akan mengukur jarak minimum antara S dengan prototipe dari tanda tangan asli.

For every set of vertices $Y_i, i = 1, 2, \dots, p$, do
 {
 Let $f = \min (|X|, |Y_i|) / \max (|X|, |Y_i|)$.
 If $f \geq \alpha$, then
 {

Find the cost matrix C between X and Y_i as described earlier.

Rotate the matrix C so that there are at least as many rows as columns.

Let r = number of rows of C .

Compute the minimum cost matching, $cost_{min}$, of rows into columns considering the cost matrix C

Let $cost_{min} = cost_{min} / (f \times r)$.

If $cost_{min} < d$, then let $d = cost_{min}$

}
}

4. Jika d lebih kecil atau sama dengan beberapa beberap nilai-nilai awal, D , maka tanda tangan uji S diterima (dianggap absah), jika sebaliknya S ditolak.

Pada langkah 3 diatas, kadang-kadang, merotasikan C dibutuhkan oleh *Algoritma Pencocokan Hungarian (Hungarian Matching Algorithm)*, karena X dan Y_i akan berkorespondensi dari baris ke kolom, yang mensyaratkan bahwa jumlah baris harus lebih kecil atau sama dengan jumlah kolom. Lagi, faktor f , mengukur persentase dari simpul-simpul yang mecocokkan baris terhadap kolom secara lengkap. Kita melanjutkan pencarian sebuah kecocokan jika f lebih kecil dari nilai awal yang telah ditetapkan, α , karena kita bersumsi bahwa kemungkinan besar tanda tangan uji adalah palsu. Jika $f \geq \alpha$, maka harga ninimum kecocokan ditemukan. Bagaimanapun juga, kolom yang tidak dapat dicocokkan akan hakimi dengan membagi nilai harga minimum, $cost_{min}$, dengan f . harga ini dinormalisasi dengan membagi jumlah baris, r . Jarak titik awal, D , bergantung pada ukuran dari kotak normalisasi (*Normaliztion Box*), yang nilainya berbeda untuk subjek yang berbeda. Nilai D ini dihitung sebagai berikut :

$$D = B \times D_{max}$$

Dimana B adalah sebuah tetapan, dan untuk setiap subjek, D_{max} adalah jarak Euclidean maksimum antara 2 dari sejumlah p protipe tanda tangannya. D_{max} dihitung dengan langkah sebagai berikut :

1. Dilakukan pemrosesan awal terhadap sejumlah p prototype tanda tangan (sebagaimana yang dijelaskan dalam bagian 3 makalah ini) untuk menghasilkan himpunan Y_1, Y_2, \dots, Y_p .
2. Ambil $D_{max} = 0$.

3. For every set of vertices $Y_i, i = 1, 2, \dots, p-1$, do

For every set of vertices

$Y_i, i = i + 1, i + 2, \dots, p-1$, do

{

Let $f = \min (|Y_i|, |Y_j|) / \max (|Y_i|, |Y_j|)$.

If $f \geq \alpha$, then

{

Find the cost matrix C between Y_i and Y_j as described earlier.

Rotate the matrix C so that there are at least as many rows as columns.

Let r = number of rows of C .

Compute the minimum cost matching, $cost_{min}$, of rows into columns considering the cost matrix C .

Let $cost_{min} = cost_{min} / (f \times r)$.

If $cost_{min} < D_{max}$, then

let $D_{max} = cost_{min}$

}

}

4. HASIL PENGUJIAN

Himpunan Data (Datasets): Sebuah tabel 5 x 3 dicetak pada selembar kertas A4 yang berguna mengumpulkan tanda tangan-tanda tangan dari para subjek. Setiap sel pada tabel berukuran dengan lebar 6,3 cm dan tingi 4,5 cm. Ukuran ini bisanya cukup untuk menuliskan tanda tangan secara leluasa. Para subjek masing-masing diminta untuk menyediakan tanda tangan asli mereka pada selembar kertas A4 yang telah dipersapkan tadi. Para subjek juga diminta untuk menuliskan tanda tangannya dengan tidak mengenai tepi garis dari tiap sel pada tabel untuk mencegah terjadinya segmentasi tanda tangan. Pada pengujian kali ini, diperoleh 5 orang yang bersedia menyumbangkan masing-masing 15 buah tanda tagannya. Sehingga terkumpullah tanda tangan asli sebanyak $15 \times 5 = 75$ buah. Gambar 2. Memperlihatkan sebuah contoh halaman yang mengandung tanda tangan asli dari subjek I.

Jenis tabel yang sama juga dipergunakan untuk mengumpulkan tiruan dari tanda tangan-tanda tangan asli. Kemudian beberapa orang diminta untuk meniru tanda tangan –tanda tangan asli yang ada sebanyak 2 sampai 4 buah untuk setiap jenis tanda tangan asli. 15 tanda tanga tiruan terkumpul untuk setiap s =jenis

tanda tangan asli. Sehingga total tanda tangan tiruan yang berhasil diperoleh adalah sebanyak $15 \times 5 = 75$ buah. Gambar 3 memperlihatkan salah satu contoh halaman yang berisi 15 tanda tangan tiruan dari tanda tangan pada gambar 2.



Gambar 2. 15 Tanda tangan asli dari subjek I



Gambar 3. 15 tanda tangan tiruan dari tanda tangan Subjek I. Tanda tangan tiruan ini diperoleh dari 4 Subjek lainnya.

Setelah dikumpulkan, tanda tangan tersebut dipindai menjadi gambar binari dengan mempergunakan sebuah HP ScanJet 3400C scanner. Dengan resolusi 300 dpi. Algoritmanya dijalankan pada sebuah Pentium III PC, 850 MHz dengan 384 MB RAM. 3 jenis pengujian yang dilakukan:

1. **Uji Asli**, dimana tanda tangan-tanda tangan asli diuji dengan menetapkan 3 tanda tangan asli pertama dari setiap subjek sebagai prototipe, $p = 3$. sisa 12 tanda tangan lain dari setiap subjek diuji terhadap 3 prototipenya masing-masing. Sehingga jumlah total pengujian tanda tangan yang dilakukan sama dengan $12 \times 5 = 60$. Test ini dilakukan untuk menghitung *Tingkat Kesalahan Penolakan*, (*False Rejection Rate*), **FRR**.
2. **Uji Tiruan Random**, dimana untuk setiap subjek A, semua tanda tangan asli dari semua subjek selain A dianggap sebagai tanda tangan tiruan/ palsu. Hal ini memberikan jumlah pengujian sebanyak $5 \times (5-1) \times 15 = 300$ kali. Pengujian ini dilakukan untuk mengukur *Tingkat Kesalahan Penyetujuan dari sistem* (*False Acceptance Rate*), **FAR_{random}**.
3. **Uji Tiruan Terampil**, dimana untuk setiap subjek, 15 tanda tangan tiruan yang dianggap paling mirip dengan yang asli diuji, sehingga menghasilkan total pengujian sebanyak $15 \times 5 = 75$ kali. Pengujian ini juga untuk mengukur *Tingkat Kesalahan Penyetujuan yang lain* (*False Acceptance rate*) **FAR_{skilled}**.

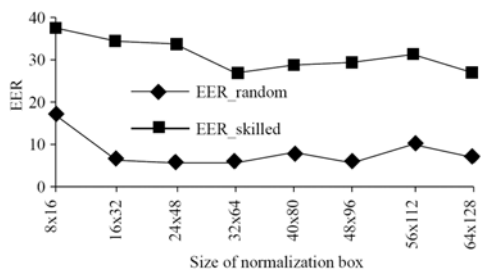
Algoritma dijalankan berkali-kali dengan mempergunakan ukuran kotak normalisasi (*Normalization Box*) : $8 \times 16, 16 \times 32, 24 \times 48, 32 \times 64, 40 \times 80, 48 \times 96, 56 \times 112, \text{ dan } 64 \times 128$. Untuk setiap Kotak Normalisasi tertentu, dilakukan sekali pemrosesan awal (preprocessed) terhadap 3 tanda tangan pertama dari setiap subjek, dan 3 tanda tangan yang telah diproses ini kemudian dipergunakan dalam semua pengujian. Rasio, α , dari jumlah baris terhadap jumlah kolom dari Matrix Harga (*cost Matrix*), C , ditetapkan 0. Faktor, B , dipergunakan untuk menghitung D_{max} disepakati bervariasi antar 0 hingga 3 tingkat penambahan (increments) sebesar 0,05.

Gambar 7 (a-h) memperlihatkan bagaimana *False Rejection Rate*, FRR, *False Acceptance Rate* untuk tiruan acak, **FAR_{random}**, dan *False Acceptance Rate* untuk tiruan yang sangat mirip, **FAR_{skilled}**, berubah terhadap B , untuk ukuran kotak normalisasi yang berbeda-beda. Semakin besar kotak normalisasi, FRR semakin mengecil dan kedua FAR semakin membesar. Secara alami terlihat bahwa kurva **FAR_{random}** lebih rendah dari pada kurva **FAR_{skilled}**, karena dalam tiruan acak, penanda tangan belum memiliki

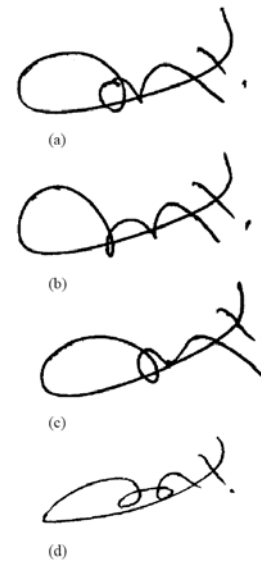
pengetahuan tentang dan atau melatih membuat tanda tangan yang hendak ditirunya. Ia hanya membuat tanda tangan yang dianggap telah mirip dengan tanda tangan yang hendak ditiru. Di lain pihak, dalam kasus tiruan terampil, pemalsu tidak akan membuat tiruan tanda tangan hingga ia mengetahui model dan berlatih meniru dari tanda tangan yang hendak dipalsukan. Kurva Keseimbangan Error (*Equal Error Rate*), yang didefinisikan sebagai tingkat dimana $FRR = FAR$, terhadap kotak normalisasi ditampilkan pada gambar 4. Perhatikan, bahwa ketika ukuran kotak meningkat, *EER* berkurang hingga ia mencapai 26,7% untuk tiruan terampil dan 5,6% untuk tiruan acak, pada ukuran 32×64 pixel, $B = 1,1655$. Setelah ukuran ini, *EER* mengalami fluktuasi (pasang-surut). Kembali ke gambar 7, kita perhatikan bahwa selain bagian (a), *FAR*-random menghilang secara mendadak pada titik dimana $FRR = FAR_{skilled}$, yang merupakan *nilai positif* dari Algoritma yang dipergunakan dalam uji ini.

Contoh dari tanda tangan asli yang ditolak dan tanda tangan palsu yang diterima oleh sistem diperlihatkan oleh gambar 5, ini merupakan kasus untuk subjek I, dengan ukuran kotak normalisasi 32×64 px, $B = 1,1655$, pada titik $FRR = FAR_{skilled}$. Untuk subjek ini, 3 dari 12 tanda tangan asli ditolak, dan hanya satu dari 15 tanda tangan palsu yang diterima oleh sistem. Semua tanda tangan tiruan acak ditolak pada kasus ini.

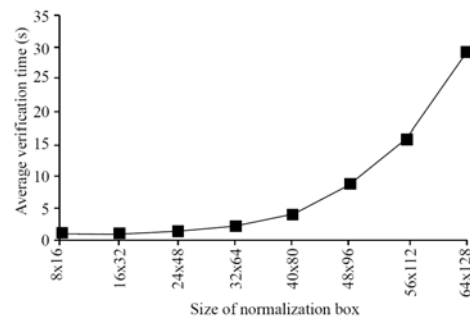
Gambar 6. Memperlihatkan rerata dari waktu proses yang dibutuhkan untuk menguji sebuah tanda tangan. Dapat dilihat bahwa, waktu yang dibutuhkan meningkat secara eksponensial seiring dengan meningkatnya ukuran kotak normalisasi, yang hal semacam ini memang perilaku khas yang terkenal dari *Hungarian Method* dalam *Complete Graph Matching*. Dengan mempertahankan ukuran kotak Normalisasi pada 32×64 px membuat waktu proses terletak dalam rentang 2 detik. Juga, dengan mempergunakan komputer yang memiliki spesifikasi/performa yang lebih tinggi seharusnya akan mampu mengurangi waktu prosesi.



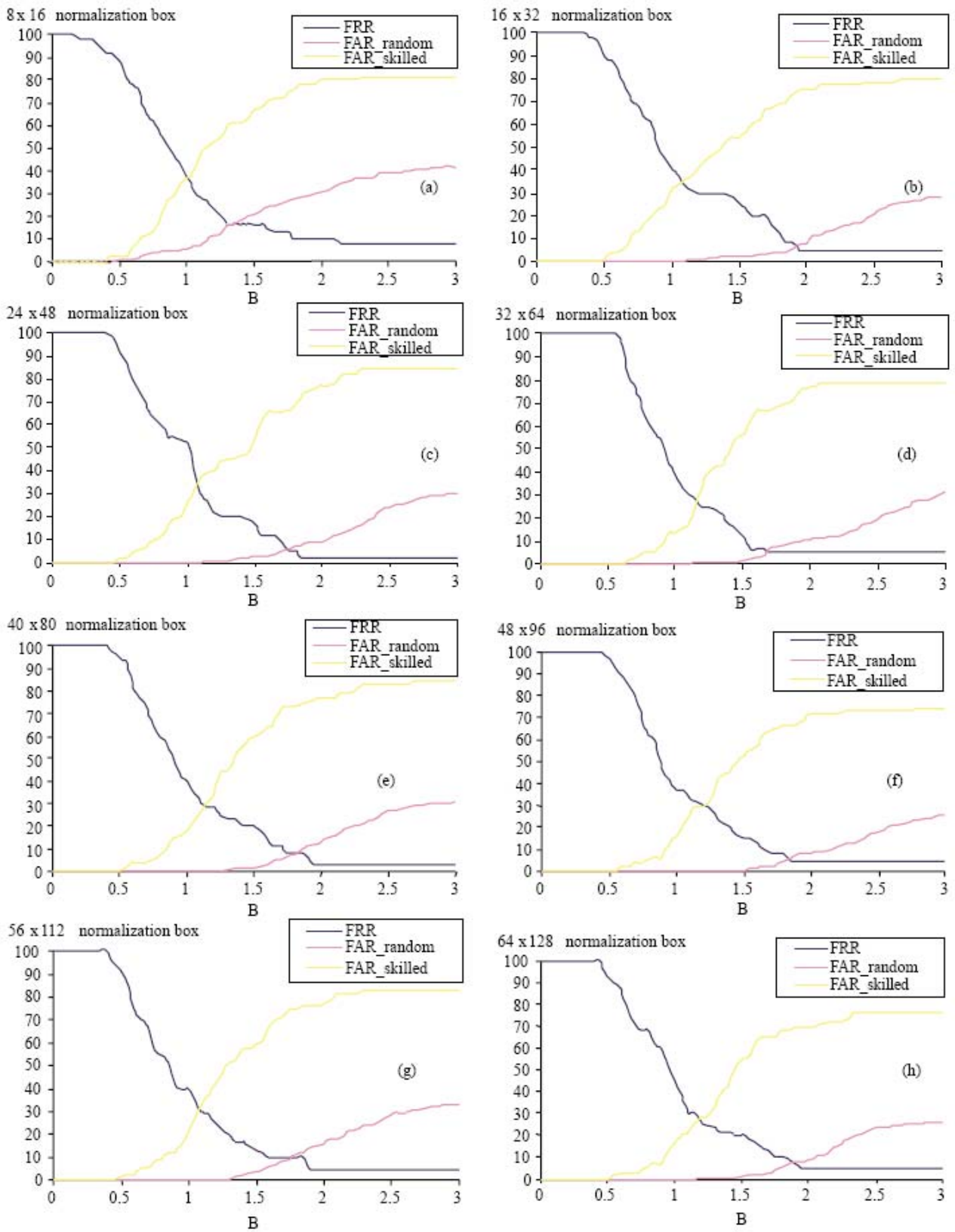
Gambar 4. *Equal Error Rate* (ERR) terhadap ukuran kotak normalisasi



Gambar 5. Contoh dari Kesalahan Penolakan (*False Rejection*) dan Kesalahan Penyetujuan (*false acceptance*) untuk kasus tanda tangan dari subjek I : (a), (b), dan (c) tanda tangan asli yang ditolak, dan (d) merupakan tanda tangan palsu yang disetujui oleh sistem.



Gambar 6: Rerata waktu verifikasi yang dibutuhkan versus ukuran Kotak Normalisasi.



Gambar 7 : False Rejection Rate (FRR) dan Acceptance Rate (FAR) terhadap B: dengan ukuran kotak Normalisasi yang berbeda-beda : (a) 8 x 16, (b) 16 x 32, (c) 24 x 48, (d) 32 x 64, (e) 40 x 80, (f) 48 x 96, (g) 56 x 112, dan (h) 64 x 128

5. Simpulan

Dalam makalah ini, masalah yang diinvestigasi adalah verifikasi tanda tangan *off-line*. Dari hasil pengamatan, dapat dengan mudah terlihat bahwa permasalahan dari verifikasi tanda tangan menjadi semakin sulit ketika yang diuji adalah mulai dari tanda tangan palsu yang bersifat acak ke tandat tangan palsu yang bersifat terampil, dan persoalan menjaidi lebih sulit ketika berhadapan dengan persoalan dimana pemilik tanda tangan yang asli membuat kesalahan dalam beberapa kasus.

Dalam makalah ini ditampilkan sebuah cara sederhana dan efektif dalam memverifikasi tanda tangan yang hanya bergantung pada intensitas pixel dari gambar raw binari dan menghindari cara-cara rumit yang bisanya sering dipergunakakn dalam literatur. Cara ini memandang persoalan verifikasi tanda tangan sebagai sebuah persoalan pencocokan graf, yang didalamnya *Hungarian Method* sebagai pemecah masalah.

Cara ini diuji dengan mempergunakan tanda tangan asli dan palsu yang dihasilkan oleh lima orang subjek. Tiga jenis pengujian yang dilakukan adalah : (1) Test asli (*genuine test*), dimana tanda tangan-tanda tangan asli diverifikasi, (2) Uji Tiruan Random, dimana untuk setiap subjek A, semua tanda tangan asli dari semua subjek selain A dianggap sebagai tanda tangan tiruan/palsu, (3) Uji Tiruan Terampil, dimana untuk setiap subjek, 15 tanda tangan tiruan yang dianggap paling mirip dengan yang asli diuji. Tingkat keseimbangan erro (*equal error rate*) sebesar 26,7% untuk pemalsuan terampil (*skilled_forgeries*) dan 5,6% untuk pemalsuan acak (*random_forgeries*) dicapai saat kotak normalisasi berukuran 32 x 64 px. Sebuah nilai positif dari algoritma yang dipergunakan disini adalah bahwa tingkat kesalahan penyetujuan (FAR) dari pemalsuan random menghilang secara mendadak pada titik dimana (*false rejection rate*) $FRR = FAR_{skilled}$ (*skilled forgery false acceptance rates*).

Mengenai waktu proses (*processing time*), terlihat bahwa ia mengalami peningkatan secara eksponensial seiring dengan meningkatnya ukuran kotak normalisasi. Dengan mempertahankan ukuran kotak Normalisasi pada 32 x 64 px membuat waktu proses terletak dalam rentang 2 detik. Juga, dengan mempergunakan komputer yang memiliki spesifikasi/performa yang lebih tinggi seharusnya akan mampu mengurangi waktu prosesi.

DAFTAR PUSTAKA

- [1] I. S. I. Abuhaiba, "Offline Signature Verification Using Graph Matching," vol.15, no.1., 2007
- [2] M. Ammar, Y. Yoshida, T. Fukumura, "Off-Line Preprocessing and Verification of Signatures,"

Int'l J. Pattern Recognition and Artificial Intelligence, vol. 2, no. 4., pp. 589-602, 1988.

- [3] J.K. Guo, D. Doermann, A. Rosenfeld, "Local correspondence for detecting random forgeries," Proceedings of the International Conference on Document Analysis and Recognition, pp. 319-323, 1997.