

Penggunaan Fungsi Hash pada Face Edge Recognition dalam Foto Database Penduduk untuk Autentikasi Diri

Made Edwin Wira Putra 13508010

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

if18010@students.if.itb.ac.id

Abstrak—Keaslian identitas diri seseorang dari kartu identitasnya belum dapat dijamin. Hal ini terbukti dengan adanya kasus-kasus penyalahgunaan kartu identitas untuk pemalsuan identitas diri yang menyimpang dari database asli penduduk. Dari masalah itu tercetus sebuah solusi dengan menggunakan perbandingan gambar foto yang terdapat pada kartu identitas dengan foto yang dimiliki server. Untuk mengurangi banyaknya perbedaan yang terjadi antara kedua foto tersebut, digunakan metode edge detection. Hasil dari edge detection tersebut akan didapatkan message digestnya. Pencocokan dilakukan dengan melakukan perbandingan terhadap message digest gambar foto kartu identitas dengan message digest yang tersimpan pada database kependudukan. Metode ini diharapkan dapat memudahkan dalam autentikasi identitas seseorang.

Kata Kunci— edge detection, fungsi hash, message digest, computer graphic

I. PENDAHULUAN

Teknologi saat ini sudah menjadi kebutuhan utama manusia di dunia. Teknologi juga kini menjadi penopang segala bentuk sistem informasi. Sistem informasi yang dulunya harus mencari data dalam buku catatan, kini tinggal mengetikkan apa yang ingin dicari dan tinggal melihat hasilnya. Internet sudah dapat diakses dengan mudah sehingga komunikasi dan pertukaran data dapat terjadi kapanpun dan dimanapun. Bahkan sekarang telah banyak dikembangkan berbagai teknologi dan sistem informasi yang mangkus untuk autentikasi identitas diri.

Identitas merupakan hal penting yang melekat pada seseorang dan menunjukkan diri sebenarnya orang tersebut. KTP (Kartu Tanda Penduduk) merupakan salah satu bentuk penanda identitas seseorang. Dengan menggunakan berbagai peralatan canggih, kita dapat dengan mudah mencari informasi mengenai seseorang. Namun sayangnya, tidak semua informasi tersebut valid atau dapat dipercaya. Semakin majunya teknologi juga berimplikasi kepada semakin canggihnya kelicikan orang. Kini orang dengan mudah membuat kartu identitas palsu. Kecurangan atau kejahatan yang terjadi dengan memanfaatkan pemalsuan identitas kini menjadi makin populer.

Oleh karena itu, diperlukan teknologi untuk

mengautentikasi identitas yang mudah dan murah untuk diterapkan. Salah satu caranya adalah dengan menggunakan metode face edge detection pada foto yang digunakan di kartu tanda penduduk. Dengan cara tersebut, dapat ditentukan identitas aslinya dengan membandingkan foto pada KTP dengan foto yang ada di Database Kependudukan. Penangkapan gambar pada foto dengan menggunakan kamera biasa tentu dapat menghasilkan perbedaan yang cukup besar ketika dibandingkan dengan yang ada di Database. Maka digunakan edge detection pada foto tersebut agar lebih kecil rentang perbedaannya.

Untuk mengefisiensikan dan memudahkan penyimpanan data foto di dalam database, kita dapat menggunakan fungsi hash sebagai pembandingnya. Data foto yang disimpan merupakan sebuah fungsi hash dari gambar edge detection foto yang telah diatur resolusinya ke ukuran tertentu.

II. DASAR TEORI

A. Edge Detection

Edge Detection merupakan perangkat mendasar dalam pengolahan citra dan *computer vision*. *Edge Detection* bertujuan untuk mengidentifikasi titik dalam sebuah citra digital di mana kecerahan citra berubah secara tajam atau disebut juga *discontinuities*.



Gambar 1 Canny Edge Detection pada Gambar Foto

Hasil penerapan *Edge Detection* untuk gambar dapat mengakibatkan satu set kurva tersambung yang menunjukkan batas-batas obyek. Sehingga, penerapan algoritma *Edge Detection* untuk gambar secara signifikan dapat mengurangi jumlah data yang diproses, sehingga dapat menyaring informasi yang kurang relevan, sambil menjaga sifat struktural yang penting dari suatu gambar.

Penandaan tepi yang diekstraksi dari gambar dua

dimensinya suatu gambar tiga dimensi dapat diklasifikasikan sebagai *viewpoint dependent* atau *viewpoint independent*. Sebuah tepi yang tergolong *viewpoint independent* biasanya mencerminkan sifat-sifat yang melekat pada benda tiga dimensi, seperti *surface markings* atau *surface shape*. Tepi yang tergolong *viewpoint dependent* dapat berubah sejalan dengan perubahan sudut pandang, dan biasanya mencerminkan geometri suatu obyek.

Ada banyak metode untuk *Edge Detection*, namun secara umum dapat dikelompokkan menjadi dua, yaitu *search-based* dan *zero-crossing-based*. Metode *search-based* mendeteksi tepi dengan pertama mengkomputasikan ukuran ketebalan tepi, kemudian mencari *local directional maxima* dari *gradient magnitude* dengan menggunakan estimasi orientasi lokal tepi. Metode *zero-crossing-based* mencari *zero-crossing* dalam *second-order* komputasi ekspresi turunan sebuah citra untuk mencari tepi-tepinya.

B. Sobel Operator

Sobel Operator digunakan dalam pemrosesan citra untuk algoritma *edge detection*. Secara teknis, *sobel operator* merupakan operator diferensiasi diskrit, mengkomputasikan perkiraan gradien fungsi intensitas suatu gambar pada setiap titik dalam gambar, hasil dari *sobel operator* bisa merupakan vektor gradient yang bersesuaian ataupun vektor normal dari vektor tersebut. Secara sederhana, *sobel operator* menghitung gradien intensitas citra pada setiap titik, memberikan arah kemungkinan perubahan intensitas terbesar, dan tingkat perubahan intensitas pada arah tersebut.

Secara matematis, *sobel operator* menggunakan dua 3x3 kernel yang *convolved* dengan citra asli untuk menghitung perkiraan dari turunan - satu untuk perubahan *horizontal*, dan satu lagi untuk perubahan *vertical*. Jika kita mendefinisikan A sebagai sumber citra, dan (G_x, G_y) adalah dua citra yang setiap titiknya berisi perkiraan turunan horizontal dan vertikal, maka perhitungan komputasinya sebagai berikut :

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * A \quad \text{and} \quad G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * A$$

dimana * menunjukkan menunjukkan 2-dimensional *convolution operation*.

Koordinat x disini didefinisikan sebagai peningkatan ke arah “kanan”, dan koordinat y didefinisikan sebagai peningkatan ke arah “bawah”. Pada setiap titik di gambar, hasil perkiraan gradien dapat dikombinasikan untuk memberikan *gradient magnitude*, menggunakan :

$$G = \sqrt{G_x^2 + G_y^2}$$

Dari informasi ini, kita dapat menghitung arah gradien

$$\Theta = \arctan\left(\frac{G_y}{G_x}\right)$$

dimana, sebagai contoh, Θ bernilai 0 untuk tepi vertikal yang semakin gelap di sisi kirinya.

Sobel Operator terdiri atas 2 operasi terpisah, yaitu *smoothing* secara tegak lurus terhadap arah derivatif dengan filter segitiga : $h(-1) = 1, h(0) = 2, h(1) = 1$ dan pusat perbedaan sederhana di arah derivatif : $h'(-1) = 1, h'(0) = 0, h'(1) = -1$. Sobel filter untuk derivatif gambar dalam dimensi yang berbeda dengan

$$x, y, z, t \in (0, -1, 1)$$

$$1D: h'_x(x) = h'(x);$$

$$2D: h'_x(x,y) = h'(x)h(y)$$

$$3D: h'_x(x,y,z) = h'(x)h(y)h(z)$$

$$4D: h'_x(x,y,z,t) = h'(x)h(y)h(z)h(t)$$

Jadi, sebagai contoh 3D kernel Sobel di arah-z :

$$h'_z(:, :, -1) = \begin{bmatrix} +1 & +2 & +1 \\ +2 & +4 & +2 \\ +1 & +2 & +1 \end{bmatrix} \quad h'_z(:, :, 0) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad h'_z(:, :, 1) = \begin{bmatrix} -1 & -2 & -1 \\ -2 & -4 & -2 \\ -1 & -2 & -1 \end{bmatrix}$$

Hasil dari *sobel operator* adalah sebuah pemetaan 2-dimensi dari gradien setiap titik. Hal tersebut dapat diolah dan dilihat seolah-olah sebagai gambar, dengan area yang bergradien tinggi (dapat diinterpretasikan sebagai tepi) terlihat sebagai garis putih. Gambar berikut mengilustrasikan hal ini, dengan menunjukkan komputasi dari Sobel Operator pada gambar sederhana.



Gambar 2 Gambar tes Grayscale



Gambar 3 Normalisasi gradient magnitude dari sobel operator



Gambar 4 Normalisas gradien-x dari Sobel Operator



Gambar 5 Normalisasi gradien-y dari Sobel Operator

C. Fungsi Hash SHA-1

Fungsi hash adalah fungsi yang menerima masukan string yang panjangnya sembarang, lalu mentransformasikannya menjadi string keluaran yang panjangnya tetap dan umumnya berukuran lebih kecil daripada ukuran string semula. Suatu fungsi hash berlaku sebagai berikut:

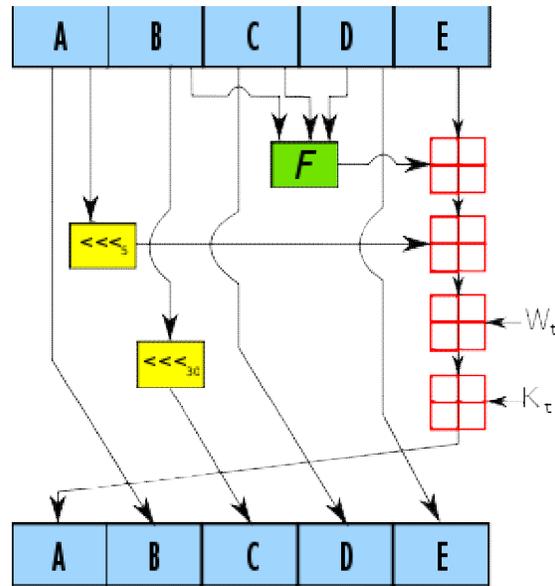
$$h = H(M)$$

Keluaran dari suatu fungsi hash disebut sebagai nilai hash atau message digest. Fungsi hash dapat menghasilkan dua message digest yang sama dari dua masukan yang berbeda. Keunikan message digest yang dihasilkan juga merupakan faktor penting dalam suatu fungsi hash dan hal ini masih terus menjadi bahan yang terus diteliti hingga sekarang.

Fungsi hash bersifat satu arah. Artinya kita dapat membentuk suatu message digest dari suatu masukan tetapi kita tidak dapat mendapatkan informasi masukan dari message digest. Hal ini dapat memberikan keuntungan dan kerugian. Keuntungannya adalah kita dapat menggunakan fungsi hash dengan aman tanpa harus merasa takut datanya diambil. Tetapi karena hal ini kita

tidak dapat bertukar informasi dengan menggunakan fungsi hash. Oleh karena inilah fungsi hash sering digunakan sebagai pengecek keaslian suatu dokumen digital, bukan untuk bertukar data.

Fungsi hash banyak digunakan dalam penyimpanan data base karena menghasilkan keluaran yang panjangnya sama sehingga mudah di simpan. Sebagai contoh adalah penyimpanan password, biasanya password disimpan dalam bentuk message digest. Hal ini memberikan keuntungan, yaitu kerahasiaan password tetap terjaga dan penyimpanan data menjadi mudah.



Gambar 6 Iterasi SHA-1

Algorithm and variant	Output size (bits)	Internal state size (bits)	Block size (bits)	Max message size (bits)	Word size (bits)	Rounds	Operations	Collisions found?
SHA-0	160	160	512	$2^{64} - 1$	32	80	+,and,or,xor,rot	Yes
SHA-1								Theoretical attack (2^{51})
SHA-2	SHA-256/224	256/224	256	512	$2^{64} - 1$	32	+,and,or,xor,shr,rot	No
	SHA-512/384	512/384	512	1024	$2^{128} - 1$	64		

Tabel 1. Perbandingan Algoritma Hash yang dikeluarkan oleh National Institute of Standards and Technology

D. Konsep Penerapan Face Edge Detection dan Fungsi Hash untuk Autentikasi Identitas

Ide utama dari penerapan Face Edge Detection adalah untuk menghasilkan sebuah gambar yang memiliki tingkat kesamaan yang tinggi. Sehingga meskipun gambar dicetak dan diambil kembali gambar digital dari hasil cetakan tersebut, dapat dihasilkan gambar yang hampir sama setelah diekstraksi tepinya. Penggunaan fungsi hash adalah untuk membandingkan kedua gambar tersebut.

Jika transfer data dilakukan dengan mengirim gambar untuk dibandingkan dengan gambar yang ada di server Database, maka diperlukan akses koneksi yang aman dan cepat. Tentunya biaya yang diperlukan untuk pengadaan layanan tersebut tidak murah. Database-pun harus memiliki kapasitas yang besar untuk dapat menampung data seluruh gambar.

Dengan menggunakan edge detection pada foto kartu identitas, device yang digunakan sebagai sarana autentikasi menjadi lebih murah. Device yang diperlukan

secara umum dapat dimiliki oleh banyak orang, yang bila dibandingkan dengan device untuk mengidentifikasi sidik jari yang lebih mahal dan sulit untuk diperoleh. Proses request data dari server juga lebih cepat karena bentuk pertukaran data yang terjadi hanya berupa teks message digest SHA-1 dari gambar yang diekstraksi tepinya dengan data dari server.

III. PENGUJIAN AKURASI FUNGSI HASH PADA GAMBAR HASIL EDGE DETECTION

Pada bagian ini, akan dilakukan pengujian akurasi fungsi hash terhadap beberapa gambar yang telah diekstraksi fungsi hashnya. Algoritma Edge Detection yang digunakan berdasarkan prinsip Sobel Operator, dan algoritma ini disebut sebagai algoritma sobel edge detection. Algoritma fungsi hash yang digunakan adalah SHA-1. Pada pengujian kali ini, implementasi diterapkan dengan menggunakan bahasa C.

```

FILE          *bmpInput, *bmpOutput;
sImage        originalImage;
sImage        edgeImage;
unsigned int   X, Y;
int           I, J;
long          sumX, sumY;
int           nColors, SUM;
unsigned long  vectorSize;
unsigned long  fileSize;
int           GX[3][3];
int           GY[3][3];
unsigned char  *pChar, someChar;
unsigned int   row, col;

/* 3x3 GX Sobel mask. Ref: www.cee.hw.ac.uk/hipr/html/sobel.html */
GX[0][0] = -1; GX[0][1] = 0; GX[0][2] = 1;
GX[1][0] = -2; GX[1][1] = 0; GX[1][2] = 2;
GX[2][0] = -1; GX[2][1] = 0; GX[2][2] = 1;

/* 3x3 GY Sobel mask. Ref: www.cee.hw.ac.uk/hipr/html/sobel.html */
GY[0][0] = 1; GY[0][1] = 2; GY[0][2] = 1;
GY[1][0] = 0; GY[1][1] = 0; GY[1][2] = 0;
GY[2][0] = -1; GY[2][1] = -2; GY[2][2] = -1;

/*-----
      SOBEL ALGORITHM
-----*/
for(Y=0; Y<=(originalImage.rows-1); Y++) {
    for(X=0; X<=(originalImage.cols-1); X++) {
        sumX = 0;
        sumY = 0;

        /* image boundaries */
        if(Y==0 || Y==originalImage.rows-1)
            SUM = 0;
        else if(X==0 || X==originalImage.cols-1)
            SUM = 0;

        /* Convolution starts here */
        else {

            /*-----X GRADIENT APPROXIMATION-----*/
            for(I=-1; I<=1; I++) {
                for(J=-1; J<=1; J++) {
                    sumX = sumX + (int)( *(originalImage.data + X + I +
                        (Y + J)*originalImage.cols) * GX[I+1][J+1]);
                }
            }

            /*-----Y GRADIENT APPROXIMATION-----*/
            for(I=-1; I<=1; I++) {
                for(J=-1; J<=1; J++) {
                    sumY = sumY + (int)( *(originalImage.data + X + I +
                        (Y + J)*originalImage.cols) * GY[I+1][J+1]);
                }
            }

            /*---GRADIENT MAGNITUDE APPROXIMATION (Myler p.218)---*/
            SUM = abs(sumX) + abs(sumY);
        }
    }

    if(SUM>255) SUM=255;
}

```

```

if (SUM<0) SUM=0;

*(edgeImage.data + X + Y*originalImage.cols) = 255 - (unsigned char)(SUM);
fwrite((edgeImage.data + X + Y*originalImage.cols), sizeof(char), 1, bmpOutput);
}
}

```

Berikut ini merupakan hasil uji terhadap gambar foto.



Gambar 7 (Bagian Kiri) foto yang digunakan untuk di sever database (bagian kanan) foto yang diambil dari kartu identitas

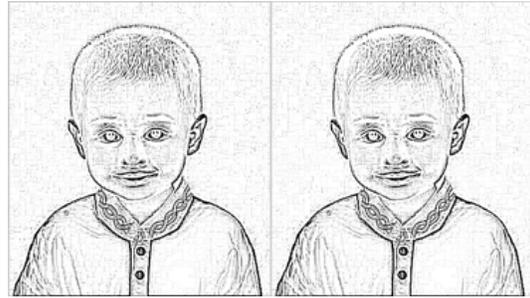


Gambar 8 (bagian Kiri)Foto yang digunakan pada server(Bagian kanan)foto yang diambil dari kartu identitas



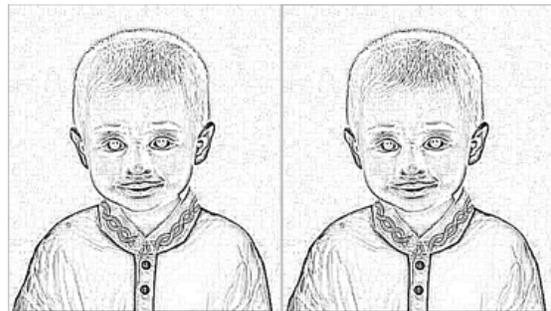
Gambar 9 (Bagian Kiri)Foto yang digunakan pada server(bagian kanan)foto yang diambil dari kartu identitas

Terlihat dari gambar-gambar tersebut terdapat perbedaan dari segi intensitas cahaya. Hasil ekstraksi tepi dari kedua gambar dan message digestnya adalah sebagai berikut :



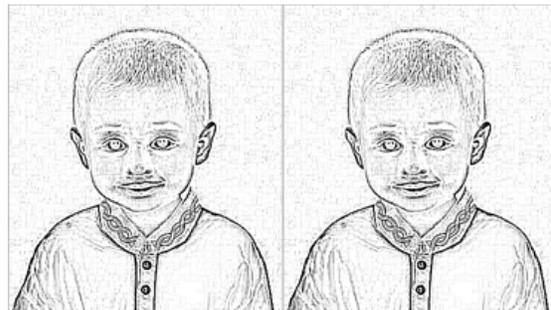
Gambar 10 (bagian kiri) dari server (bagian kanan) dari kartu identitas

Message Digest gambar server :
448622E2C2F0BDE7D71A13C1FEDCDC67D33F5636
Message Digest gambar kartu identitas :
026920E9194C99DA2A08DBE3CA74EEBB4413D70D



Gambar 11 (bagian kiri) dari server (bagian kanan) dari kartu identitas

Message Digest gambar server :
448622E2C2F0BDE7D71A13C1FEDCDC67D33F5636
Message Digest gambar kartu identitas :
EFEF502211B20C96EBC62BA2211420E0E494BB18



Gambar 12 (bagian kiri) dari server (bagian kanan) dari kartu identitas

Message Digest gambar server :
448622E2C2F0BDE7D71A13C1FEDCDC67D33F5636
Message Digest gambar kartu identitas :
448622E2C2F0BDE7D71A13C1FEDCDC67D33F5636

Dari hasil diatas didapati bahwa meski gambar hasil ekstraksi tepi terlihat sama, setelah di cek SHA-nya gambar tersebut masih menghasilkan message digest yang berbeda. Hal ini dikarenakan masih banyaknya noise yang muncul dari gambar hasil ekstraksi tepi. Metode ini akan menjadi lebih akurat bila diterapkan penghilang noise, sehingga didapat gambar yang murni hanya tepi saja.

IV. ANALISIS HASIL UJI

Berikut adalah analisis penulis dalam penggunaan fungsi hash pada *face edge detection* foto kartu identitas penduduk.

- Kelebihan :
Besar ruang penyimpanan yang dibutuhkan jauh lebih kecil. Sebagai perbandingan, penyimpanan dalam bentuk gambar menghabiskan ruang 2 kilo byte sedangkan dengan message digest membutuhkan ruang 40 karakter. Jika satu karakter menghabiskan 1 byte, maka ruang yang dibutuhkan hanya sekitar 40 byte.
- Kekurangan :
Hasil message digest dari gambar yang telah diekstraksi tepinya masih berbeda. Hasil ekstraksi tepi dari setiap gambar belum mereduksi noise hingga tingkat dimana gambar yang didapat hanya tepi yang menjadi *primary key* gambar.
- Permasalahan :
Permasalahan muncul ketika terdapat satu titik perbedaan pada gambar kartu identitas penduduk yang tidak dapat direduksi, sehingga hasil message digestnya tidak akan pernah menemui kesamaan dengan data server. Untuk mengatasi permasalahan ini, kualitas hasil pencetakan gambar foto pada kartu identitas harus baik, dan pereduksian noise dari gambar hasil ekstraksi adalah pokok bahasan yang harus diselesaikan untuk mendapatkan hasil message digest yang tepat.

V. KESIMPULAN

Berdasarkan hasil uji akurasi dan analisis mengenai penggunaan fungsi hash pada hasil ekstraksi tepi dari gambar foto kartu identitas diri, didapatkan kesimpulan sebagai berikut :

1. Pendeteksian tepi pada suatu foto dengan menggunakan algoritma sobel, belum dapat mereduksi noise yang timbul.
2. Algoritma SHA-1 menghasilkan message digest yang berbeda pada 2 gambar yang berbeda, meski hanya terdapat sedikit perbedaan.
3. Penggunaan fungsi hash dan *face edeg detection* untuk kartu identitas belum memenuhi ekspektasi yang diinginkan. Message digest foto server dan message digest foto pada kartu identitas masih berbeda.

DAFTAR PUSTAKA

<http://www.pages.drexel.edu/~weg22/edge.html>
<http://en.wikipedia.org/wiki/SHA-1>
http://en.wikipedia.org/wiki/Sobel_operator
http://en.wikipedia.org/wiki/Edge_detection
<http://www.gutgames.com/post/Sobel-Edge-Detection-and-Laplace-Edge-Detection-in-C.aspx>
<http://www.gutgames.com/post/Edge-detection-in-C.aspx>
<http://www.doc.ic.ac.uk/~akf/handel-c/cgi-bin/forum.cgi?msg=172>
<http://www.informatika.org/~rinaldi/Kriptografi/2010-2011/SHA.ppt>
<http://www.informatika.org/~rinaldi/Kriptografi/2010-2011/Fungsi%20Hash.ppt>
http://www.informatika.org/~rinaldi/Kriptografi/2009-2010/Makalah2/Makalah2_IF3058_2010_065.pdf

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 9 Mei 2011

ttd

Made Edwin Wira Putra
13508010