

TUGAS BESAR - SISTEM PENGENALAN SIDIK JARI

Rio Ariyanto Kusuma (13317011), Fajar Adi Nugroho (13317029), Yan Setiaji (13317077)

Dosen: Prof. Dr. Ir. Tati Latifah Erawati Rajab, Astri Handayani, S.T., M.T.

Tanggal Percobaan: 7/12/2020 s/d 21/12/2020

EL4125-Pengolahan Citra Digital

Pengolahan Citra Digital - Sekolah Teknik Elektro dan Informatika ITB

Abstrak

Dewasa ini pemanfaatan biometrik untuk keperluan pengenalan identitas telah banyak dilakukan. Salah satu biometrik yang umum digunakan adalah sidik jari. Analisis sidik jari dapat digunakan untuk keperluan identifikasi serta verifikasi. Namun, tidak jarang bahwa kualitas sidik jari yang terbaca pada sensor memiliki kualitas yang buruk. Untuk itu diperlukan sebuah algoritma untuk meningkatkan kualitas citra sidik jari yang rusak. Selain itu, diperlukan juga metode pencocokan citra sidik jari dengan sidik jari referensi yang mampu mencocokkan dengan kualitas yang baik. Percobaan ini dilakukan untuk memperoleh citra sidik jari laten dengan kualitas yang lebih baik menggunakan algoritma peningkatan kualitas citra serta memperoleh tingkat kecocokan yang tinggi menggunakan metode template matching.

Kata kunci: Sidik jari, peningkatan kualitas, template matching

1. PENDAHULUAN

1.1 LATAR BELAKANG

Manusia memiliki bagian-bagian tubuh yang dapat menjadi ciri yang membedakan seseorang dengan orang lain. Ciri tersebut bersifat spesifik dan dapat digunakan untuk mengenali identitas seseorang dengan menggunakan metode yang biasa disebut sebagai pengenalan biometrik. Meskipun demikian, tidak semua karakteristik dari bagian tubuh dapat digunakan sebagai biometrik. Terdapat beberapa syarat yang harus dipenuhi bagi suatu bagian tubuh untuk dapat dikategorikan sebagai biometrik, yaitu: keuniversalan (tiap orang harus memiliki ciri khas), ciri khas (terdapat perbedaan karakteristik antar individu), permanen (karakteristik harus invarian terhadap waktu), dan kolektibilitas (dapat diukur secara kuantitatif) [1].

Sejauh ini, terdapat beberapa bagian tubuh yang telah digunakan untuk analisis biometrik. Salah satu biometrik yang sangat umum digunakan adalah sidik jari. Sidik jari merupakan pola tonjolan epidermis pada jari tangan, telapak tangan, serta telapak kaki, yang terbentuk akibat ketegangan dari lapisan basal pada lembaran kulit [2]. Pola yang terbentuk pada sidik

jari sangat beragam. Namun, secara umum pola tersebut dapat dikelompokkan menjadi 3 yaitu pola lingkaran (*whorls*), simpul (*loops*), serta busur (*archs*) [2]. Ketiga kelompok pola tersebut merupakan pola inti dari sidik jari, yang artinya letak dari ketiga kelompok pola tersebut berada pada bagian tengah jari tangan atau kaki. Selain ketiga kelompok pola tersebut, yang menjadikan sidik jari bersifat unik dan spesifik adalah tekstur guratan pola pendukung di sekitar pola inti. Gabungan dari pola inti dan pola pendukung inilah yang kemudian dapat dianalisis untuk dapat menunjukkan identitas dari seseorang secara spesifik.

Dalam membuat suatu sistem biometrik untuk menganalisis sidik jari, terdapat dua pilihan sistem yang biasa digunakan yaitu sistem verifikasi dan sistem identifikasi[3]. Sistem verifikasi merupakan sistem yang memvalidasi identitas seseorang dengan membandingkan data biometrik yang diambil dengan template biometrik orang tersebut yang telah tersimpan dalam database sistem. Adapun sistem identifikasi digunakan untuk mengenali individu dengan mencari template dari semua pengguna dalam database untuk menemukan kecocokan. Permasalahan yang sering timbul baik pada proses verifikasi atau identifikasi adalah kesalahan pemberian kesimpulan kemiripan akibat pembacaan tekstur sidik yang tidak lengkap atau berkualitas buruk. Untuk itu diperlukan proses peningkatan kualitas citra sidik jari berkualitas buruk yang terbaca pada sensor serta proses pencocokan yang memiliki performansi yang baik.

Pada percobaan ini, penulis akan menjelaskan mengenai algoritma peningkatan kualitas dan pencocokan citra sidik jari menggunakan metode *template matching*. Percobaan dilakukan dengan memanfaatkan dataset sidik jari dan mengimplementasikan beberapa program yang tersedia bebas di internet.

1.2 TUJUAN

Tujuan yang ingin dicapai oleh penulis melalui percobaan ini adalah sebagai berikut:

- a. Memperbaiki kualitas citra sidik jari
- b. Memverifikasi kepemilikan sidik jari dengan metode pencocokan

2. RUANG LINGKUP PERMASALAHAN

2.1 BATASAN DAN ASUMSI

Percobaan yang dilakukan penulis menggunakan batasan dan asumsi pada beberapa hal, diantaranya:

- Menggunakan 1 dataset dari FVC2002
- Hanya terdapat satu sidik jari pada setiap citra

2.2 KRITERIA PENYELESAIAN MASALAH

Penulis menganggap percobaan ini telah selesai apabila pada tahap analisis didapatkan luasan dibawah kurva ROC (*Reciever Operating Characteristic*) memperoleh nilai diatas 80%

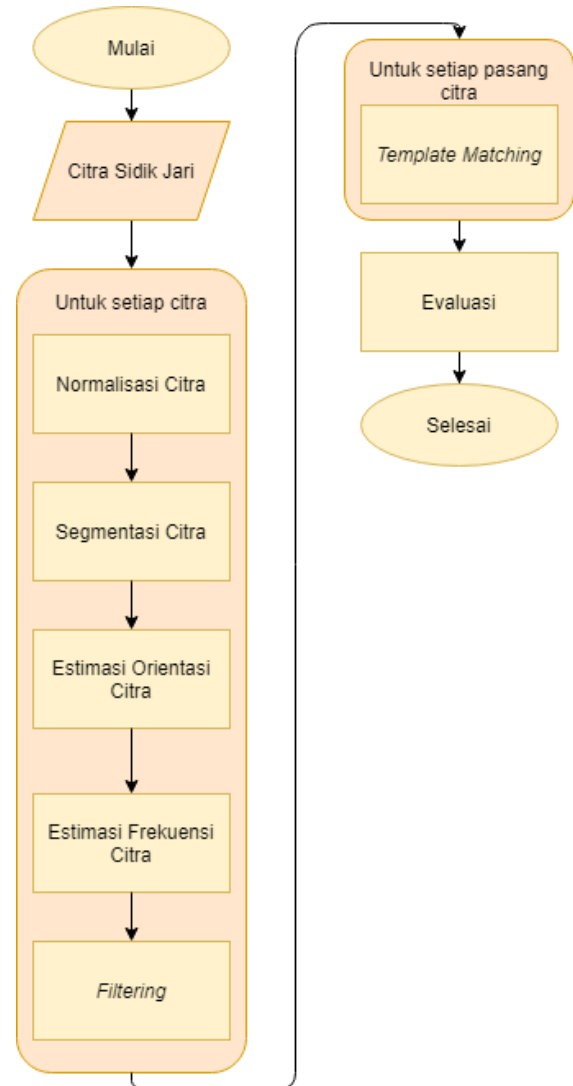
3. METODOLOGI

3.1 ALAT DAN BAHAN

Pada percobaan ini penulis menggunakan dataset sidik jari yang tersedia secara bebas dari Fingerprint Verification Competition (FVC) 2002 [4]. Data tersebut terdiri dari data sidik jari 10 subjek yang berbeda dengan masing-masing subjek terdiri dari 8 sidik jari untuk 1 jari tertentu. Adapun alat yang digunakan adalah bahasa pemrograman Python beserta *library* OpenCV, Numpy, Matplotlib, dan Sklearn. Penulis juga mengimplementasikan algoritma yang telah tersedia [5] dan memberikan beberapa perubahan untuk mendapatkan hasil yang baik.

3.2 RANCANGAN PERCOBAAN

Rancangan percobaan ini dapat dilihat pada diagram alir yang tertera pada Gambar 3.1.



Gambar 3-1 Diagram Alir Percobaan

3.2.1 Normalisasi Citra

Citra awal dimisalkan dengan notasi I , maka normalisasinya adalah

$$I_{norm}(x,y) = \frac{I(x,y) - I_{mean}}{I_{std}}$$

dengan I_{mean} dan I_{std} adalah nilai rata-rata dan standar deviasi dari citra awal.

3.2.2 Segmentasi Citra

Untuk citra ternormalisasi I_{norm} , daerah yang dianggap citra sidik jari adalah daerah yang memiliki nilai standar deviasi melebihi ambang batas tertentu. Pada percobaan ini, I_{norm} dibagi menjadi 16×16 daerah, lalu daerah yang memiliki standar deviasi lebih dari 0.1 dianggap sebagai daerah sidik jari. Setelah itu citra dinormalisasi lagi dengan rata-rata dan standar deviasi baru dari citra bagian sidik jari.

3.2.3 Estimasi Orientasi Sidik Jari

Untuk citra yang telah tersegmentasi I_{seg} , magnituda gradien sidik jari diestimasi menggunakan kernel gradien Gaussian untuk setiap piksel citra. Pada percobaan ini standar deviasi Gaussian adalah 1 dan ukuran kernel 6 kali dari standar deviasi.

Lalu untuk setiap daerah/blok, magnituda gradien sidik jari diestimasi menggunakan kernel Gaussian yang dikonvolusi dengan magnituda pada proses sebelumnya sebagai representasi penjumlahan yang terbobotkan. Pada percobaan ini standar deviasi Gaussian adalah 7 dan ukuran kernel 6 kali dari standar deviasi.

Karena citra bisa terdapat derau, sidik jari yang korup, dan adanya titik singular dan *minutiae*, estimasi orientasi dari proses sebelumnya bisa jadi kurang akurat. Oleh karena itu dilakukan *low pass filtering* lagi menggunakan kernel Gaussian pada *continuous vector field* yang didefinisikan sebagai

$$\Phi_x(i, j) = \frac{G_{xx}(i, j)}{\sqrt{G_{xy}^2(i, j) + (G_{xx}(i, j) - G_{yy}(i, j))^2}}$$

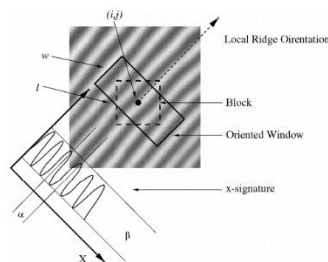
$$\Phi_y = \frac{G_{xx}(i, j) - G_{xy}(i, j)}{\sqrt{G_{xy}^2(i, j) + (G_{xx}(i, j) - G_{yy}(i, j))^2}}$$

Dimana G_{xx} dan G_{yy} adalah magnituda gradien daerah/blok pada arah x dan y serta G_{xy} adalah *low pass filtering* yang sama namun pada perkalian magnituda gradien piksel arah x dan y. *Low pass filtering* dilakukan pada daerah/blok yang sama pada proses sebelumnya. Sehingga orientasi akhirnya adalah

$$\theta(i, j) = \frac{1}{2} \tan^{-1} \frac{\Phi_x(i, j)}{\Phi_y(i, j)}$$

3.2.4 Estimasi Frekuensi Citra

Untuk setiap daerah/blok, frekuensi diestimasi dengan merotasi citra pada blok tsb sebesar negatif dari sudut orientasinya lalu menjumlahkan intensitas citra pada sumbu x dan mencari jarak puncak pertama dan terakhir. Ilustrasi dari tahap ini dapat dilihat pada Gambar 3.2



Gambar 3-2 Ilustrasi Estimasi Frekuensi [6]

3.2.5 Filtering

Filtering dilakukan menggunakan *filter* Gabor dengan parameter orientasi dan frekuensi lokal yang didapat pada 2 tahapan sebelumnya serta standar deviasi arah x dan y sebesar 0.65. *Filter* Gabor dirumuskan sebagai

$$h(x, y; \phi, f) = \exp \left\{ -\frac{1}{2} \left[\frac{x_\phi^2}{\delta_x^2} + \frac{y_\phi^2}{\delta_y^2} \right] \right\} \cos(2\pi f x_\phi)$$

$$x_\phi = x \cos \phi + y \sin \phi$$

$$y_\phi = -x \sin \phi + y \cos \phi$$

3.2.6 Template Matching

Template matching dilakukan menggunakan algoritma Oriented FAST dan Rotated BRIEF (ORB) [7]. Pada proses deteksi *keypoints*, ORB menggunakan algoritma FAST [8] pada deteksi pertama kali lalu dilanjutkan dengan algoritma Harris [9]. Karena FAST dan Harris tidak sensitif terhadap skala, maka ORB menggunakan *multiscale image pyramid*. Pada proses pembuatan deskriptor, ORB menggunakan rBRIEF, yaitu versi BRIEF [10] yang robas terhadap rotasi. Pada tahapan klasifikasi digunakan rata-rata jarak *keypoint* yang cocok sebagai ambang batas.

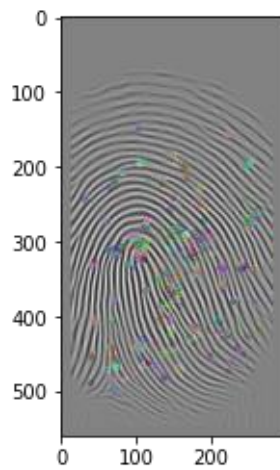
3.2.7 Evaluasi

Evaluasi dilakukan menggunakan *Area Under Curve* (AUC) dari *Region Operating Characteristic* (ROC) [11].

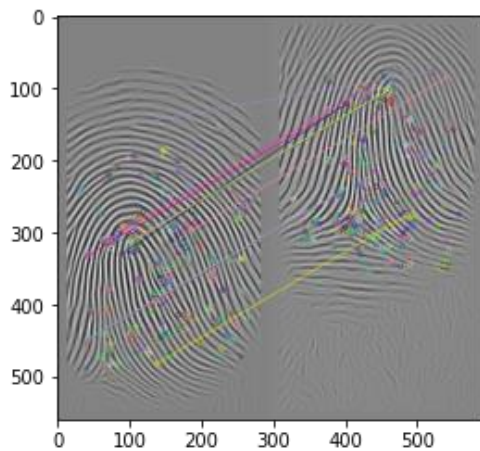
4. LANGKAH DAN HASIL PERCOBAAN



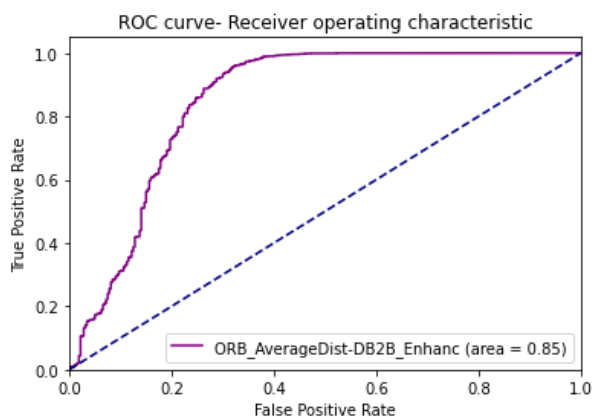
Gambar 4-1 Perbandingan Citra Awal dan Citra yang Mengalami Perbaikan



Gambar 4-2 Deteksi Keypoint



Gambar 4-3 Hasil Template Matching



Gambar 4-4 Evaluasi ROC

Pada tahap pertama dilakukan pra-pemrosesan terhadap citra mentah sidik jari hasil dari pembacaan sensor TouchView II. Citra awal yang diperoleh masih memiliki beberapa kelemahan antara lain terdapat pori-pori, pengaburan, dan detail *ridge* yang kurang menonjol. Dilakukan *enhancement* terhadap citra asli dengan menggunakan filter gabor. Seperti yang telah diketahui bahwa dua parameter utama dalam filter gabor adalah sudut orientasi dan nilai frekuensi. Karena pada suatu sidik jari terdapat

berbagai pola dengan berbagai orientasi, berbeda dengan filter gabor pada umumnya yang hanya menerapkan satu set parameter untuk keseluruhan citra, parameter yang digunakan dalam filter gabor kali ini diperoleh melalui beberapa tahap. Tahap yang pertama adalah dilakukan perhitungan gradien dan orientasi lokal pada keseluruhan citra sidik jari. Dari informasi lokal tersebut dapat dilakukan pemfilteran gabor dengan parameter yang berbeda-beda untuk setiap bagian sidik jari (menyesuaikan dengan kondisi lokal) Tujuan dari *enhancement* kali ini adalah meningkatkan/menonjolkan detail *ridge* pada suatu citra seperti yang ditunjukkan pada Gambar 4-1.

Langkah selanjutnya adalah melakukan pendeteksian *keypoint* dengan menggunakan deskriptor ORB. Deskriptor ORB yang digunakan merupakan bawaan dari pustaka OpenCV (yang mana algoritma ORB juga dikembangkan oleh OpenCV sebagai alternatif dari SIFT dan SURF yang berlisensi). Dari informasi deskriptor antara dua citra sidik jari yang hendak diidentifikasi dilakukan proses *matching* dengan metode *brute force* (dicoba satu per satu) yang mana menghitung jarak Hamming dari tiap deskriptor yang dikalkulasi dan mengembalikan nilai terkecil dari keseluruhan nilai yang diperoleh.

Perhitungan nilai *match* dilakukan dengan mengatur ambang batas rata-rata jarak fitur yang dianggap *match* oleh algoritma, penentuan besar nilai ambang dapat diperoleh melalui analisis ROC yang mana memberikan nilai *True Positive Rate* dibandingkan dengan *False Positive Rate* pada beberapa nilai ambang batas yang telah ditentukan, untuk algoritma dan basis data tertentu akan memberikan karakteristik kurva ROC yang berbeda, dari kurva yang diperoleh dapat berguna untuk membantu mencari nilai ambang batas fitur yang sama yang dapat dikategorikan apakah dua citra sidik jari yang berbeda adalah identik atau tidak identik.

5. ANALISIS

Gambar 4-4 menunjukkan nilai ROC yang cukup bagus pada citra yang telah mengalami perbaikan (*enhancement*). Kurva ROC menunjukkan performa perbandingan True Positive Rate vs False Positive Rate pada nilai ambang batas yang berbeda-beda. Nilai True Positive Rate yang besar serta luasan di bawah kurva (AUC) yang besar menunjukkan performa yang baik. Model yang baik harus memiliki nilai performa AUC yang mendekati 1. Nilai AUC pada dataset yang sudah di-*enhance* diperoleh pada 0,85. Skor AUC diatas 0.8 tersebut tergolong sangat bagus [11]. Setidaknya, sistem ini

dapat digunakan untuk proses verifikasi sidik jari (penggolongan biner autokorelasi). Ambang batas pada sistem yang ideal hendaknya memberikan nilai TPR=1 dan FPR=0, namun pada sistem nyata hal tersebut jarang ditemui. Pemilihan nilai ambang batas berdasarkan analisis ROC dapat dipilih dengan memilih nilai ambang batas yang memberikan hasil pada kurva ROC sesuai dengan kriteria yang diinginkan (apakah mementingkan nilai TPR yang tinggi dengan berapapun nilai FPR atau dengan proporsi TPR/FPR yang paling tinggi). Pemilihan nilai ambang batas menyesuaikan dengan tujuan dibentuknya suatu sistem.

6. KESIMPULAN DAN SARAN

6.1 KESIMPULAN

Penulis pada percobaan ini berhasil melakukan proses peningkatan kualitas dari citra sidik jari yang semula memiliki kualitas buruk. Menggunakan *low pass filter* untuk mengestimasi orientasi serta filter gabor untuk mengestimasi frekuensi, kualitas dari citra dataset yang digunakan mampu untuk ditingkatkan

Penulis menggunakan *template matching* sebagai metode pencocokan sidik jari untuk keperluan verifikasi kepemilikan sidik jari. Dengan melakukan analisis pada kurva ROC, didapati bahwa nilai untuk luasan di bawah kurva ROC adalah sebesar 0.85 atau dapat dikategorikan sangat baik.

6.2 SARAN

Untuk dapat meningkatkan performa sistem, deteksi *keypoint* lebih bagus jika mengikutsertakan *minutiae*. Dapat pula ditambahkan metode pembelajaran mesin (jaringan saraf tiruan konvolusi) untuk meningkatkan performa sistem.

DAFTAR PUSTAKA

- [1] Jain, Anil K., Arun Ross, and Salil Prabhakar. "An introduction to biometric recognition." *IEEE Transactions on circuits and systems for video technology* 14.1 (2004): 4-20.
- [2] Kücken, Michael, and Alan C. Newell. "Fingerprint formation." *Journal of theoretical biology* 235.1 (2005): 71-83.
- [3] Maltoni, Davide, et al. *Handbook of fingerprint recognition*. Springer Science & Business Media, 2009.
- [4] <http://bias.csr.unibo.it/fvc2002/download.asp>. Diakses pada 18 Desember 2020 pukul 19.00
- [5] <https://github.com/mountzouri/fingerprint-recognition>. Diakses pada 19 Desember 2020 pukul 09.15
- [6] Hong, Lin, Yifei Wan, and Anil Jain. "Fingerprint image enhancement: algorithm and performance evaluation." *IEEE transactions on pattern analysis and machine intelligence* 20.8 (1998): 777-789.
- [7] Rublee, Ethan, et al. "ORB: An efficient alternative to SIFT or SURF." *2011 International conference on computer vision*. Ieee, 2011.
- [8] Rosten, Edward, and Tom Drummond. "Machine learning for high-speed corner detection." *European conference on computer vision*. Springer, Berlin, Heidelberg, 2006.
- [9] Harris, Christopher G., and Mike Stephens. "A combined corner and edge detector." *Alvey vision conference*. Vol. 15. No. 50. 1988.
- [10] Calonder, Michael, et al. "Brief: Binary robust independent elementary features." *European conference on computer vision*. Springer, Berlin, Heidelberg, 2010.
- [11] Hosmer Jr, David W., Stanley Lemeshow, and Rodney X. Sturdivant. *Applied logistic regression*. Vol. 398. John Wiley & Sons, 2013.
- [12] <https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5>. Diakses pada 19 Desember 2020 pukul 18.00