



TESIS KI142502

Multi-feature Fusion Menggunakan Fitur Scale Invariant Feature Transform dan Local Extensive Binary Pattern untuk Pengenalan Pembuluh Darah pada Jari

HARDIKA KHUSNULIAWATI
NRP. 5114201047

PEMBIMBING I
Dr. Eng. Chastine Fatichah, S.Kom, M.Kom
NIP. 197512202001122002

PEMBIMBING II
Rully Soelaiman, S.Kom, M.Kom
NIP. 197002131994021001

PROGRAM MAGISTER
JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2016

[Halaman ini sengaja dikosongkan]



THESIS KI142502

Multi-feature Fusion Using Integration Feature of Scale Invariant Feature Transform and Local Extensive Binary Pattern for Finger Vein Recognition

HARDIKA KHUSNULIAWATI
NRP. 5114201047

SUPERVISOR I
Dr. Eng. Chastine Fatichah, S.Kom, M.Kom
NIP. 197512202001122002

SUPERVISOR II
Rully Soelaiman, S.Kom, M.Kom
NIP. 197002131994021001

**MASTER PROGRAM
DEPARTEMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2016**

[Halaman ini sengaja dikosongkan]

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Komputer (M.Kom.)
di
Institut Teknologi Sepuluh Nopember Surabaya

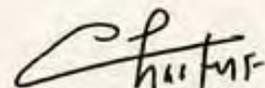
oleh:
HARDIKA KHUSNULIAWATI
Nrp. 5114201047

Dengan judul
*Multi-feature Fusion Menggunakan Fitur Scale Invariant Feature Transform
dan Local Extensive Binary Pattern untuk Pengenalan Pembuluh Darah pada
Jari*

Tanggal Ujian 22-6-2016
Periode Wisuda . 2015 Genap

Disetujui oleh.

Dr. Eng. Chastine Fatichah, S.Kom, M.Kom
NIP 197512202001122002



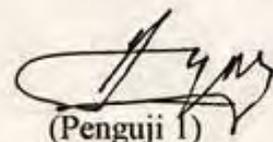
(Pembimbing 1)

Rully Soelaiman, S.Kom, M.Kom
NIP 197002191994021001



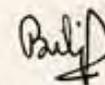
(Pembimbing 2)

Dr. Agus Zainal Arifin, S.Kom, M.Kom
NIP 197208091995121001



(Penguji 1)

Bilqis Amaliah, S.Kom, M.Kom
NIP 197509172001122002



(Penguji 2)

Anny Yuniarti, S.Kom., M.Comp.Sc
NIP 198106222005012002



(Penguji 3)

Direktur Program Pasca Sarjana,



Prof. Dr. Djauhar Manfaat, M.Sc., Ph.D.
NIP 196012021987011001

[Halaman ini sengaja dikosongkan]

Multi-feature Fusion Menggunakan Integrasi Fitur Scale Invariant Feature Transform dan Local Extensive Binary Pattern untuk Pengenalan Pembuluh Darah pada Jari

Nama Mahasiswa : Hardika Khusnuliawati

NRP : 5114201047

Pembimbing : 1. Dr. Eng. Chastine Fatichah, S.Kom, M.Kom
2. Rully Soelaiman, S.Kom, M.Kom

ABSTRAK

Pengenalan pembuluh darah jari merupakan salah satu area dalam bidang biometrika. Sehingga tahap-tahap dalam proses pengenalan pembuluh darah jari memiliki kesamaan dengan proses pengenalan menggunakan biometrika lain yaitu meliputi pengumpulan citra, praproses, ekstraksi fitur, dan pencocokan. Tingkat keberhasilan dari tahap pencocokan ditentukan oleh pemilihan fitur pembuluh darah jari yang digunakan. Kondisi citra pembuluh darah yang rentan terhadap perubahan skala, rotasi maupun translasi menyebabkan kebutuhan akan fitur yang tahan terhadap kondisi tersebut menjadi hal yang penting.

Fitur *Scale Invariant Feature Transform* (SIFT) adalah fitur yang telah cukup banyak digunakan untuk kasus pencocokan citra serta mampu tahan terhadap degradasi kondisi citra akibat perubahan skala, rotasi maupun translasi. Akan tetapi, fitur SIFT kurang memberikan hasil optimal jika diekstraksi dari citra dengan variasi tingkat keabuan seperti yang disebabkan oleh perbedaan intensitas pencahayaan. Fitur *Local Extensive Binary Pattern* (LEBP) merupakan fitur yang tahan terhadap variasi tingkat keabuan dengan informasi karakteristik lokal yang lebih kaya dan diskriminatif. Oleh karena itu digunakan teknik fusi untuk memperoleh informasi dari fitur SIFT dan fitur LEBP sehingga diperoleh fitur yang memiliki ketahanan terhadap degradasi kondisi citra akibat perubahan skala, rotasi, translasi, variasi tingkat keabuan seperti yang disebabkan oleh perbedaan intensitas pencahayaan.

Penelitian ini mengusulkan *multi-feature fusion* menggunakan fitur SIFT dan LEBP untuk pengenalan pembuluh darah pada jari. Fitur hasil *fusion* diproses

dengan metode *Learning Vector Quantization* (LVQ) untuk menentukan apakah citra pembuluh darah jari yang diuji dapat dikenali atau tidak. Dengan menggunakan *multi-feature fusion* diharapkan mampu representasi fitur yang dapat meningkatkan akurasi dari proses pengenalan pembuluh darah jari meskipun fitur diambil dari citra yang mengalami degradasi.

Berdasarkan hasil uji coba diperoleh bahwa penggunaan *multi-feature fusion* dengan fitur SIFT dan LEBP memberikan hasil yang relatif lebih baik jika dibandingkan dengan hanya menggunakan fitur tunggal. Hal tersebut dapat dilihat dari peningkatan hasil kinerja sistem pada kondisi optimum dengan nilai akurasi sebesar 97,50%, TPR sebesar 0,9400 dan FPR sebesar 0,0128.

Kata kunci: pembuluh darah jari, *Scale Invariant Feature Transform*, *Local Extensive Binary Pattern*, *multi-feature fusion*, *Learning Vector Quantization*

**Multi-feature Fusion Using Integration Feature of Scale
Invariant Feature Transform and Local Extensive Binary
Pattern for Finger Vein Recognition**

Student's Name : Hardika Khusnuliawati

Student's ID : 5114201047

Advisors : 1. Dr. Eng. Chastine Fatichah, S.Kom, M.Kom
2. Rully Soelaiman, S.Kom, M.Kom

ABSTRACT

Finger vein recognition is one of the areas in the field of biometrics. The steps of finger vein recognition has in common with other biometric recognition process which include image acquisition, preprocessing, feature extraction and matching. The success rate of matching stage is determined by the selection of features. The conditions of finger vein images are susceptible to changes in scale, rotation and translation. The need for features that are resistant to these conditions becomes important.

Scale invariant Feature Transform (SIFT) feature is a feature that has been quite widely used for image matching case and be able to withstand degradation due to changes in the condition of the image scale, rotation and translation. However, SIFT feature provide less optimal results when extracted from the image with gray level variations such as those caused by differences in lighting intensity. *Local Extensive Binary Pattern* (LEBP) feature is a feature that has resistance to gray level variations with richer and discriminatory local characteristics information. Therefore the fusion technique is used to obtain information from SIFT feature and LEBP feature. So that, the feature that has been produced can resist degradation problems such as changes in the condition of the image scale, rotation, translation, and gray level variations which caused by differences in lighting intensity.

This study proposes a multi-feature fusion using SIFT and LEBP features for finger vein recognition. This fusion feature will be processed by Learning Vector Quantization (LVQ) method to determine whether the testing image can be

recognized or not. By using a multi-feature fusion, it is expected to get representations of features that can improve the accuracy of the finger vein recognition although the feature is taken from the degraded image.

Based on experiment results, finger vein recognition that use multi-feature fusion using integration feature of scale invariant feature transform and local extensive binary pattern provide a better result than only use a single feature. It can be seen from the increase of performance system in optimum condition. The accuracy value can achieve 97.50%, TPR at 0.9400 and FPR at 0.0128.

Keyword: finger vein, Scale Invariant Feature Transform, Local Extensive Binary Pattern, multi-feature fusion, Learning Vector Quantization

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Puji syukur kepada Allah SWT atas segala karunia dan rahmat-Nya penulis dapat menyelesaikan tesis yang berjudul “***Multi-feature Fusion Menggunakan Fitur Scale Invariant Feature Transform dan Local Extensive Binary Pattern untuk Pengenalan Pembuluh Darah pada Jari***”.

Harapan dari penulis semoga apa yang tertulis di dalam buku tesis ini dapat bermanfaat bagi pengembangan ilmu pengetahuan saat ini, serta dapat memberikan kontribusi yang nyata. Dalam pelaksanaan dan pembuatan tugas akhir ini tentunya sangat banyak bantuan yang penulis terima dari berbagai pihak, tanpa mengurangi rasa hormat penulis ingin mengucapkan terima kasih sebesar-besarnya kepada:

1. Bapak Rully Soelaiman, S.Kom., M.Kom. dan Ibu Dr. Eng. Chastine Fatichah, S.Kom, M.Kom selaku dosen pembimbing penulis yang telah memberikan banyak arahan, bantuan, nasihat, dan perhatian yang banyak sehingga membuat penulis dapat menyelesaikan tesis ini.
2. Bapak Dr.Eng Darlis Herumurti, S.Kom.,M.Kom.selaku ketua jurusan Teknik Informatika ITS, Bapak Waskitho Wibisono, S.Kom.,M.Eng.,Ph.D dan segenap dosen Teknik Informatika yang telah memberikan ilmunya selama penulis menempuh perkuliahan di Teknik Informatika ITS.
3. Para penulis artikel ilmiah yang karyanya direferensi oleh tesis ini sehingga membantu mempermudah penulisan Tesis ini.
4. Bapak, Ibu, Adik dan keluarga yang selalu memberikan dukungan moral, material, dan doa sehingga mendapat kelancaran untuk menyelesaikan tesis ini.
5. Teman-teman angkatan 2014 pascasarjana jurusan Teknik Informatika ITS yang telah menemani maupun memberikan saran, masukan, dan dukungan terhadap pengerjaan tesis ini.
6. Serta pihak-pihak lain yang tidak dapat disebutkan disini yang telah banyak membantu penulis dalam penyusunan tesis ini.

Penulis telah berusaha sebaik-baiknya dalam menyusun tugas akhir ini, namun penulis mohon maaf apabila terdapat kekurangan, kesalahan maupun kelalaian yang telah penulis lakukan. Kritik dan saran yang membangun dapat disampaikan sebagai bahan perbaikan selanjutnya.

Surabaya, Juni 2016

Hardika Khusnuliawati

DAFTAR ISI

ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvii
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan masalah	5
1.3. Tujuan dan Manfaat Penelitian	6
1.4. Kontribusi	6
1.5. Batasan Masalah	6
BAB II DASAR TEORI DAN KAJIAN PUSTAKA	7
2.1. Citra Pembuluh Darah Jari	7
2.2. Praproses	8
2.2.1. <i>Contrast Limited Adaptive Histogram Equalization (CLAHE)</i>	8
2.2.2. Frangi Filter	9
2.2.3. Median Filter	10
2.2.4. Operasi Morfologi	10
2.3. Ekstraksi Fitur Pembuluh Darah Jari	12
2.3.1. <i>Fitur Scale-Invariant Feature Transform (SIFT)</i>	13
2.3.2. <i>Algoritma Bag Of Visual Words (BVW)</i>	15
2.3.3. <i>Fitur Local Extensive Binary Pattern (LEBP)</i>	16
2.4. <i>Learning Vector Quantization (LVQ)</i>	18
BAB III METODOLOGI PENELITIAN	21
3.1 Alur Metodologi Penelitian	21
3.1.1 Studi Literatur Ekstraksi Fitur SIFT	22
3.1.2 Studi Literatur Ekstraksi Fitur LEBP	22

3.1.3	Studi Literatur Metode Klasifikasi LVQ.....	22
3.1.4	Perancangan Ekstraksi Fitur SIFT-LEBP berbasis <i>Multi-Feature Fusion</i>	22
3.1.5	Perancangan Sistem Untuk Mendeteksi PDR Perancangan Sistem Pengenalan Pembuluh Darah Jari.....	23
3.1.6	Implementasi Rancangan Sistem Pencocokkan Citra Pembuluh Darah Jari	23
3.1.7	Uji Coba dan Analisis Hasil Uji Metode Pengenalan.....	23
3.2	Rancangan Sistem	25
3.2.1	Citra Retina Dataset Citra Pembuluh Darah Jari	26
3.2.2	Praproses.....	27
3.2.3	Ekstraksi Fitur SIFT	30
3.2.4	Ekstraksi Fitur LEBP	30
3.2.5	Fusion Fitur SIFT-LEBP	37
3.2.6	Pengenalan Pembuluh Darah Jari dengan LVQ	43
BAB IV HASIL UJI COBA DAN PEMBAHASAN		45
4.1	Lingkungan Uji Coba.....	45
4.2	Uji Coba Pengenalan Pembuluh Darah Jari dengan Fitur SIFT	45
4.3	Uji Coba Pengenalan Pembuluh Darah Jari dengan Fitur LEBP.....	48
4.4	Uji Coba Pengenalan Pembuluh Darah Jari dengan Fitur SIFT-LEBP	49
4.5	Analisis Uji Coba Sistem Pengenalan Pembuluh Darah Jari.....	54
BAB V KESIMPULAN DAN SARAN		59
5.1	Kesimpulan	59
5.2	Saran	60
DAFTAR PUSTAKA.....		61
LAMPIRAN A		65
LAMPIRAN B.....		103
BIODATA PENULIS.....		107

DAFTAR GAMBAR

Gambar 2.1 Contoh citra pembuluh darah jari (Kumar & Zhou, 2012).....	7
Gambar 2.2 Ilustrasi operasi morfologi dilasi dan rotasi (a) Citra A merupakan citra asli yang akan diproses dengan operasi morfologi menggunakan structuring element B (b) citra hasil dilasi (c) citra hasil erosi	11
Gambar 2.3 Pembentukan deskriptor <i>keypoint</i> (Lowe, 2004).....	15
Gambar 2.4 Penggambaran piksel-piksel yang terlibat untuk memperoleh nilai dari fitur LmBP (Liu dkk., 2014)	17
Gambar 2.5 Penggambaran piksel-piksel yang terlibat untuk memperoleh nilai dari fitur LdBP (Liu dkk., 2014).....	18
Gambar 3.1 Alur Metodologi Penelitian	21
Gambar 3.2 Alur rancangan sistem pengenalan pembuluh darah jari.....	25
Gambar 3.3 Contoh citra pembuluh darah jari (a) jari telunjuk (b) jari tengah.....	26
Gambar 3.4 Citra hasil tahap praproses (a) Citra hasil CLAHE (b) Citra hasil Frangi Filter (c) Citra hasil Median Filter (d) Citra hasil Operasi Morfologi (e) Citra grayscale hasil CLAHE yang telah melalui tahap cropping dan normalisasi ukuran citra	28
Gambar 3.5 <i>Pseudocode</i> dari Tahap Praproses	30
Gambar 3.6 Kumpulan <i>keypoints</i> yang diekstraksi dari citra <i>grayscale</i>	30
Gambar 3.7 <i>Pseudocode</i> dari Tahap Ekstraksi Fitur SIFT	32
Gambar 3.8 <i>Pseudocode</i> dari Tahap Ekstraksi Fitur LEBP	36
Gambar 3.9 <i>Pseudocode</i> dari Tahap Penghitungan nilai LDC	37
Gambar 3.10 <i>Pseudocode</i> dari Tahap Ekstraksi Fitur LmBP	39
Gambar 3.11 <i>Pseudocode</i> dari Tahap Ekstraksi Fitur LdBP	39
Gambar 3.12 Diagram Alur dari Fusion Fitur SIFT-LEBP	40
Gambar 3.13 <i>Pseudocode</i> dari Fusion Fitur SIFT-LEBP	42
Gambar 3.14 <i>Pseudocode</i> dari Pengenalan Pembuluh Darah Jari dengan Metode LVQ44	
Gambar 4.1 Hasil pengenalan pembuluh darah jari dengan fitur SIFT yang meliputi (a) Nilai akurasi (b) Nilai TPR (c) Nilai FPR	46

Gambar 4.2 Contoh citra yang berhasil dikenali oleh sistem pengenalan pembuluh darah jari dengan fitur SIFT (a) Citra yang mengalami efek translasi (b) Citra yang mengalami efek rotasi (c) Citra yang mengalami efek rotasi dan translasi.....	47
Gambar 4.3 Contoh citra yang berhasil dikenali oleh sistem pengenalan pembuluh darah jari dengan fitur LEBP (a),(b) citra dengan perbedaan intensitas pencahayaan	49
Gambar 4.4 Hasil pengenalan pembuluh darah jari dengan fitur SIFT-LEBP dengan pengubahan nilai parameter jumlah hidden layer yang meliputi (a) Nilai akurasi (b) Nilai TPR (c) Nilai FPR	51
Gambar 4.5 Hasil pengenalan pembuluh darah jari dengan fitur SIFT-LEBP dengan pengubahan nilai parameter jumlah epochs yang meliputi (a) Nilai akurasi (b) Nilai TPR (c) Nilai FPR	52
Gambar 4.6 Hasil pengenalan pembuluh darah jari dengan fitur SIFT-LEBP dengan pengubahan nilai parameter learning rate yang meliputi (a) Nilai akurasi (b) Nilai TPR (c) Nilai FPR	53
Gambar 4.7 Contoh citra yang berhasil dikenali oleh sistem pengenalan pembuluh darah jari dengan fitur SIFT-LEBP (a),(b),(c) Citra dengan degradasi transformasi geometris dan perbedaan intensitas pencahayaan	57

DAFTAR TABEL

Tabel 3.1 <i>Confussion matrix</i> untuk memperoleh nilai FAR dan FRR	24
Tabel 3.2 Daftar Variabel yang Digunakan pada Tahap Praproses.....	28
Tabel 3.3 Daftar Fungsi yang Digunakan pada Tahap Ekstraksi Fitur SIFT	29
Tabel 3.4 Daftar Variabel yang Digunakan pada Tahap Ekstraksi Fitur SIFT	31
Tabel 3.5 Daftar Fungsi yang Digunakan pada Tahap Ekstraksi Fitur SIFT	32
Tabel 3.6 Daftar Variabel yang Digunakan pada Tahap Ekstraksi Fitur LEBP.....	33
Tabel 3.7 Daftar Fungsi yang Digunakan pada Tahap Ekstraksi Fitur LEBP.....	35
Tabel 3.8 Daftar Variabel yang Digunakan pada Tahap Fusion Fitur SIFT-LEBP	41
Tabel 3.9 Daftar Fungsi yang Digunakan pada Tahap Fusion Fitur SIFT-LEBP	42
Tabel 3.10 Daftar Variabel yang Digunakan pada Tahap Pengenalan pembuluh Darah Jari dengan Metode LVQ	43
Tabel 3.10 Daftar Variabel yang Digunakan pada Tahap Pengenalan pembuluh Darah Jari dengan Metode LVQ	44
Tabel 3.11 Daftar Fungsi yang Digunakan pada Tahap Pengenalan pembuluh Darah Jari dengan Metode LVQ	44
Tabel 4.1 Tetapan parameter uji coba pengenalan pembuluh darah jari dengan fitur SIFT.....	46
Tabel 4.2 Hasil uji coba pengenalan pembuluh darah jari dengan fitur LmBP, LdBP, dan LEBP.....	48
Tabel 4.3 Tetapan parameter uji coba pengenalan pembuluh darah jari dengan fitur SIFT-LEBP.....	50
Tabel 4.4 Hasil uji coba pengenalan pembuluh darah jari dengan fitur SIFT, LEBP, dan SIFT-LEBP.....	57

[Halaman ini sengaja dikosongkan]

BAB I

PENDAHULUAN

Bab ini membahas hal-hal yang mendasari pengerjaan tesis. Bahasan meliputi latar belakang, rumusan masalah, tujuan dan manfaat penelitian, kontribusi, dan batasan masalah penelitian.

1.1. Latar Belakang

Pengenalan pembuluh darah jari merupakan wilayah baru dalam bidang biometrika. Saat ini, penelitian pada wilayah tersebut semakin memperoleh perhatian. Hal tersebut dikarenakan pembuluh darah jari dapat digunakan sebagai alternatif bagi identifikasi seseorang selain penggunaan sidik jari atau iris yang telah umum digunakan. Pembuluh darah jari memiliki karakteristik biometrika seperti bersifat universal, berbeda antara satu objek dengan objek yang lain, bersifat permanen, dan dapat diterima (Yang dkk., 2014). Selain itu, penggunaan pembuluh darah jari memiliki beberapa kelebihan jika dibandingkan dengan sidik jari atau wajah yaitu struktur pembuluh darah jari tersembunyi sehingga sulit untuk mencuri informasi pola pembuluh darah jari seseorang tanpa sepengetahuan orang tersebut. Sedangkan yang kedua, sulit untuk memalsukan pembuluh darah jari karena pembuluh darah jari hanya dapat ditangkap dari seorang yang hidup (Kumar & Zhou, 2012). Yang, dkk. (2014) menambahkan kelebihan lain dari penggunaan pola pembuluh darah yaitu minim rusak oleh faktor eksternal karena terlindung di dalam tubuh.

Tahap-tahap dalam proses pengenalan pembuluh darah jari memiliki kesamaan dengan proses pengenalan menggunakan biometrika lain yaitu meliputi pengumpulan citra, praproses, ekstraksi fitur, dan pencocokan. Dari serangkaian tahapan tersebut, tahap ekstraksi fitur menjadi tahap yang cukup banyak menangkap perhatian. Berbagai metode telah banyak diusulkan berkaitan dengan tahap ekstraksi fitur. Mulai dari ekstraksi fitur tekstur, pola, maupun *minutiae* dari citra pembuluh darah jari. Metode ekstraksi fitur berupa pola pembuluh darah jari telah dilakukan oleh Miura, dkk. (2004) yang menggunakan *line tracking* (Miura dkk., 2004). Kemudian, Miura, dkk. (2007) kembali mengusulkan metode untuk

mengekstraksi pola pembuluh darah jari dengan metode *maximum curvature* (Miura dkk., 2007). Sedangkan Song, dkk. (2011) mengusulkan penggunaan *mean curvature* untuk mengekstraksi pola pembuluh darah jari beserta bentuk geometri dari citra jari sebagai fitur (Song dkk., 2011). Metode ekstraksi fitur yang telah disebutkan ternyata belum berfokus pada penanganan degradasi kondisi citra akibat perubahan skala, rotasi maupun translasi. Padahal permasalahan tersebut sering ditemui dalam kasus pencocokan citra pembuluh darah jari (Liu dkk., 2014). Kondisi citra yang rentan terhadap perubahan skala, rotasi maupun translasi dapat mempersulit proses pencocokan citra dan mengurangi hasil keakuratan dari proses pencocokan tersebut. Sehingga penting untuk memperhatikan pemilihan fitur yang tahan terhadap kondisi tersebut.

Salah satu fitur yang telah cukup banyak digunakan untuk kasus pencocokan citra serta mampu tahan terhadap degradasi kondisi citra akibat perubahan skala, rotasi maupun translasi yaitu fitur *Scale Invariant Feature Transform* (SIFT) yang diusulkan pertama kali oleh Lowe (2004). Penggunaan fitur ini telah diimplementasikan pada kasus deteksi objek (Lowe, 2004), temu kembali citra (Yu dkk., 2013), dan pencocokan citra termasuk citra biometrika. Pada bidang biometrika, fitur SIFT telah digunakan antara lain untuk kasus pembuluh darah telapak tangan (Ladoux dkk., 2009), pembuluh darah jari (Pang dkk., 2012) dan pengenalan wajah (Xu & Zhao, 2015). Dari penelitian-penelitian pada bidang biometrika tersebut, fitur SIFT terbukti memberikan nilai kesalahan pengenalan yang rendah. Selain itu, fitur ini juga menunjukkan kinerja yang lebih baik jika dibandingkan penggunaan fitur lain pada kasus pengenalan citra dengan kondisi yang mengalami rotasi. Meskipun kemampuan fitur SIFT dapat diandalkan untuk mengatasi kesalahan pencocokan citra akibat perubahan skala, rotasi maupun translasi, fitur SIFT kurang memberikan hasil optimal jika diekstraksi dari citra dengan variasi tingkat keabuan seperti yang disebabkan oleh perbedaan intensitas pencahayaan (Kim dkk., 2012). Hal ini dikarenakan fitur SIFT sangat bergantung pada hasil penentuan *keypoint*. Sedangkan jumlah *keypoint* sebagai deskriptor fitur dapat dipengaruhi oleh intensitas pencahayaan. Padahal, citra pembuluh darah jari sangat rentan dengan perbedaan intensitas pencahayaan. Oleh karena itu, perlu fitur lain untuk memperkuat fitur SIFT

sehingga mendapatkan integrasi fitur yang tidak hanya *robust* terhadap perubahan geometri tetapi juga perbedaan intensitas pencahayaan.

Fitur *Local Binary Pattern* (LBP) dikenalkan oleh Ojala, dkk. (2002) sebagai fitur yang memiliki ketahanan terhadap variasi tingkat keabuan citra (Ojala dkk., 2002). Fitur ini menyimpan pola tekstur ke dalam vektor biner dimana setiap nilai biner yang diperoleh merupakan hasil selisih piksel pusat dengan piksel-piksel tetangga. Hasil selisih ini tidak dipengaruhi oleh perubahan intensitas pencahayaan sehingga pola yang disimpan dapat dikatakan tahan terhadap variasi tingkat keabuan (Ojala dkk., 2002). Beberapa metode lain kemudian diusulkan untuk meningkatkan kinerja dari fitur LBP pada citra pembuluh darah jari. Zhang, dkk. (2010) mengusulkan metode ekstraksi fitur *Local derivative Pattern* (LDP) untuk mengurangi waktu komputasi dari metode LBP (Zhang dkk., 2010). Berikutnya, Rosdi, dkk. (2011) mengusulkan *Local Line Binary Pattern* (LLBP) yang berhasil menunjukkan hasil pengenalan citra pembuluh darah jari yang lebih baik dibandingkan fitur LBP maupun LDP (Rosdi dkk., 2011). Fitur lain diusulkan oleh Meng, dkk. (2012) yaitu *Local Directional Code* (LDC) dimana fitur ini berhasil meningkatkan hasil pengenalan citra pembuluh darah jari dengan fitur LLBP (Meng dkk., 2012). Kemudian, Liu, dkk. (2014) mengusulkan suatu fitur yang lebih kaya menyimpan informasi dan lebih diskriminatif jika dibandingkan metode ekstraksi fitur yang telah diusulkan sebelumnya (Liu dkk., 2014). Fitur yang diusulkan yaitu *Local Extensive Binary Pattern* (LEBP) yang merupakan hasil kombinasi dari dari fitur *Local Multilayer Binary Pattern* (LmBP) dan fitur *Local Directional Binary Pattern* (LdBP). Fitur LmBP adalah variasi dari bentuk fitur LBP sedangkan fitur LdBP adalah modifikasi dari fitur LDC. Fitur yang dihasilkan metode LEBP adalah fitur yang tahan terhadap variasi tingkat keabuan dengan informasi karakteristik lokal yang lebih kaya dan diskriminatif. Selanjutnya fitur LEBP ini akan dipadukan dengan fitur SIFT untuk memperoleh fitur *robust*.

Selain itu, tahap pencocokan adalah tahap yang juga perlu diperhatikan. Selama ini proses pencocokan dari citra pembuluh darah jari menggunakan penghitungan jarak seperti *euclidean distance* dan *hamming distance* dengan nilai ambang batas tertentu. Hal ini menyebabkan akurasi hasil pencocokan sangat

bergantung dengan nilai ambang batas yang ditetapkan. Sehingga diimplementasikan *machine learning* untuk sejumlah penelitian pada kasus pembuluh darah jari. Metode klasifikasi yang telah banyak diterapkan untuk kasus pengenalan pembuluh darah jari yaitu *Support Vector Machine* (SVM). Sebagai contoh Park (2012) menggunakan SVM untuk kasus pengenalan pembuluh darah jari dengan LBP dan Wavelet transform sebagai fitur (Park, 2012). Penelitian lainnya dilakukan oleh Khellat-kihel, dkk. (2014) yang juga menggunakan SVM dengan konsep *multi-class* serta *Gabor Filter* sebagai fitur (Khellat-kihel dkk., 2014). Alternatif metode lain sebagai pengganti SVM yang dapat digunakan yaitu *Learning Vector Quantization* (LVQ). Metode LVQ telah berhasil digunakan untuk kasus pengenalan ekspresi wajah seperti oleh Bashyal dan Venayagamoorthy (2008) yang menggunakan LVQ dengan LBP sebagai fitur (Bashyal & Venayagamoorthy, 2008) dan de Vries, dkk. (2015) yang menggunakan LVQ dan LBP sebagai fitur (de Vries dkk., 2015). Meskipun LVQ telah umum digunakan dalam bidang biometrika tetapi penerapannya untuk kasus pengenalan pembuluh darah jari belum banyak dipublikasikan. Metode LVQ sendiri memiliki kelebihan yaitu waktu tahap pelatihan yang cepat dan memungkinkan untuk melakukan pengkajian terhadap proses pelatihan (de Vries dkk., 2015). Untuk melakukan pengenalan pembuluh darah pada jari, data masukan dari metode LVQ yaitu fitur *robust* dengan menggunakan dua fitur yaitu fitur SIFT dan fitur LEBP. Akan tetapi, diperlukan suatu teknik tertentu untuk dapat menggunakan kedua fitur tersebut secara optimal.

Teknik fusi dapat didefinisikan sebagai suatu proses mengombinasikan berbagai sumber data untuk memperoleh informasi yang lebih baik (Castanedo, 2013). Dalam hal ini, informasi yang lebih baik memiliki arti lebih efisien, berkualitas tinggi, atau informasi yang dihasilkan lebih relevan. Jika melihat definisi dari teknik fusi dan manfaatnya, maka teknik fusi dapat digunakan untuk memperoleh informasi dari kedua fitur yang telah disebutkan sebelumnya yaitu fitur SIFT dan fitur LEBP. Fitur hasil fusi ini akan digunakan sebagai masukan dalam proses pengenalan pembuluh darah jari. Penggunaan teknik fusi untuk memperoleh representasi fitur baru dari pengolahan beberapa fitur telah dilakukan oleh Wang, dkk. (2014) dimana fitur yang dikenakan proses fusi yaitu fitur SIFT

dan fitur histogram warna. Fitur hasil fusi dari kedua fitur tersebut berhasil meningkatkan akurasi dari kasus pencocokan video (Wang dkk., 2014).

Oleh karena itu, penelitian ini mengusulkan *multi-feature fusion* menggunakan fitur SIFT dan LEBP untuk pengenalan pembuluh darah pada jari. Fitur hasil *fusion* kemudian diproses dengan metode *Learning Vector Quantization* (LVQ) untuk menentukan apakah citra pembuluh darah jari yang diuji dapat dikenali atau tidak. Penggunaan fitur SIFT dan LEBP berbasis *multi-feature fusion* diharapkan mampu menghasilkan fitur yang memiliki ketahanan terhadap variasi tingkat pencahayaan, variasi perubahan skala, rotasi, maupun translasi sehingga hasil akurasi dari pengenalan pembuluh darah jari dapat meningkat. Fitur SIFT digunakan untuk menghasilkan fitur yang tahan terhadap perubahan geometri seperti perubahan skala, rotasi maupun translasi. Sedangkan fitur LEBP digunakan untuk menghasilkan fitur yang tahan terhadap variasi tingkat keabuan. Hasil integrasi fitur SIFT dan LEBP ini kemudian menjadi fitur yang akan diproses oleh metode LVQ.

1.2. Rumusan masalah

Perumusan masalah yang terdapat pada penelitian ini yaitu sebagai berikut:

1. Bagaimana merancang *multi-feature fusion* menggunakan fitur SIFT dan fitur LEBP untuk ekstraksi fitur pada citra pembuluh darah jari.
2. Bagaimana mengimplementasikan sistem pengenalan pembuluh darah jari dengan *multi-feature fusion* menggunakan fitur SIFT dan fitur LEBP serta metode klasifikasi LVQ.
3. Bagaimana melakukan uji coba pengenalan pembuluh darah jari dengan *multi-feature fusion* menggunakan fitur SIFT dan fitur LEBP serta metode klasifikasi LVQ.
4. Bagaimana melakukan analisis hasil uji coba pengenalan pembuluh darah jari dengan *multi-feature fusion* menggunakan fitur SIFT dan fitur LEBP serta metode klasifikasi LVQ.

1.3. Tujuan dan Manfaat Penelitian

Tujuan dari penelitian ini adalah menghasilkan sistem untuk pengenalan pembuluh darah jari dengan *multi-feature fusion* menggunakan fitur SIFT dan fitur LEBP. Sedangkan manfaat dari penelitian ini adalah memberikan referensi metode identifikasi informasi seseorang dengan area biometrika lain selain iris, sidik jari, dan pembuluh darah tangan.

1.4. Kontribusi

Penelitian ini memiliki kontribusi yaitu *multi-feature fusion* menggunakan fitur SIFT dengan fitur LEBP yang digunakan pada kasus pengenalan pembuluh darah jari. Metode *multi-feature fusion* ditujukan untuk menghasilkan fitur yang memiliki ketahanan terhadap terhadap variasi tingkat pencahayaan, variasi perubahan skala, rotasi, maupun translasi.

1.5. Batasan Masalah

Permasalahan yang dibahas dalam penelitian ini memiliki beberapa batasan, yaitu:

1. Data yang digunakan untuk uji coba adalah data citra pembuluh darah jari dari Tsinghua University yaitu basis data THU-FVFDT2 : <http://www.sz.tsinghua.edu.cn/labs/vipl/thu-fvfdt.html> (Kumar & Zhou, 2012).
2. Dari 156 data individu yang tersedia, hanya digunakan data citra dari 50 individu saja dimana dari setiap individu diambil 6 citra pembuluh darah jari sehingga total data citra yang diperoleh sejumlah 300 data.
3. Implementasi menggunakan perangkat lunak MATLAB.

BAB II

DASAR TEORI DAN KAJIAN PUSTAKA

Bab ini berisi pembahasan tentang kajian dan dasar teori yang digunakan sebagai teori pendukung penelitian. Adapun pembahasannya meliputi citra pembuluh darah jari, metode-metode dari tahap praproses, metode-metode ekstraksi fitur, dan metode klasifikasi LVQ.

2.1. Citra Pembuluh Darah Jari

Citra dari pembuluh darah jari merupakan salah satu citra yang dapat digunakan sebagai alternatif dalam proses pengenalan seseorang. Saat ini, penelitian dari citra pembuluh darah jari semakin mendapat perhatian dalam bidang biometrika. Sebagai salah satu karakteristik biometrika, citra pembuluh darah jari memiliki sifat-sifat antara lain bersifat universal, berbeda antara satu objek dengan objek yang lain, bersifat permanen, dan dapat diterima. Jika dibandingkan dengan karakteristik biometrika lain seperti sidik jari atau wajah, pembuluh darah jari memiliki dua kelebihan yaitu citra pembuluh darah jari hanya dapat diambil dari tubuh makhluk hidup dan cukup sulit untuk dikopi atau dijiplak karena terlindung di dalam tubuh sehingga minim rusak oleh faktor eksternal (Yang dkk., 2014).

Untuk mengambil citra pembuluh darah jari, digunakan sinar infra merah jarak dekat dengan panjang gelombang antara 700 hingga 1000 nanometer. Sinar infra merah jarak dekat dapat menyerap secara intensif kandungan hemoglobin dalam darah, akan tetapi jaringan lain secara mudah akan ikut terambil sehingga pola dari pembuluh darah akan ditangkap sebagai bayangan (Yang dkk., 2014). Gambar 2.1 menunjukkan hasil pengambilan citra pembuluh darah jari.



Gambar 2.1 Contoh citra pembuluh darah jari (Kumar & Zhou, 2012)

2.2. Praproses

Citra pembuluh darah jari dapat menjadi alternatif untuk alat identifikasi individu dengan memanfaatkan keunikan pola pembuluh darah jari dari masing-masing individu tersebut. Akan tetapi, karena citra pembuluh darah jari diambil dengan sinar infra merah, maka jaringan lain secara mudah akan ikut terambil sehingga pola dari pembuluh darah akan ditangkap sebagai bayangan sehingga tampak samar. Hal tersebut menyebabkan susah untuk membedakan antara pola pembuluh darah sebagai *foreground* dengan *background*.

Oleh karena itu, diperlukan tahap praproses sehingga citra yang akan diproses merupakan citra dengan kualitas baik yaitu citra dengan pola pembuluh darah jari terlihat jelas. Beberapa metode praproses yang perlu dilewati untuk dapat memperoleh citra dengan kualitas yang lebih baik yaitu metode CLAHE untuk memperjelas kontras citra sehingga perbedaan *foreground* dengan *background* dapat tampak lebih jelas, Frangi Filter untuk menonjolkan pola pembuluh darah, median filter untuk mengurangi derau dari citra, serta operasi morfologi untuk mendapatkan pola pembuluh darah dalam representasi citra biner.

2.2.1. Contrast Limited Adaptive Histogram Equalization (CLAHE)

Metode CLAHE merupakan pengembangan dari metode Histogram Equalization (HE) (Yakno dkk., 2011). Apabila HE memproses citra secara keseluruhan maka CLAHE akan membagi terlebih dahulu citra ke dalam sub-sub bagian citra. Setelah itu, metode HE diberlakukan pada setiap sub bagian tersebut. Langkah pertama yang dilakukan adalah menghitung nilai *cumulative density function* (CDF) untuk setiap sub citra dengan ukuran $w_x \times w_y$. Nilai CDF yang digunakan untuk menransformasi histogram dari setiap sub citra mengikuti Persamaan 2.1 (Yakno dkk., 2011).

$$i_{enh(i)} = \frac{cdf(i) - cdf_{min}}{(w_x \times w_y) - cdf_{min}} \times (L - 1) \quad (2.1)$$

dimana $i_{enh(i)}$ merupakan nilai intensitas hasil citra yang diperbaiki, $cdf(i)$ merupakan *cumulative density function* dari intensitas i , cdf_{min} adalah nilai *cumulative density function* minimum dari keseluruhan citra, sedangkan nilai L

jumlah tingkat keabuan dari citra. Citra hasil CLAHE diperoleh dari hasil transformasi intensitas berdasarkan Persamaan 2.1 pada setiap sub citra.

2.2.2. Frangi Filter

Frangi filter merupakan algoritma *filtering* untuk mencari struktur geometris yang dianggap berbentuk tubular, sebagai contoh struktur dari pembuluh darah (Frangi dkk., 1998). Deteksi struktur geometris tersebut menggunakan nilai *eigenvalues* dari *Hessian Matrix*. Oleh karena pembuluh darah memiliki diameter yang berbeda-beda maka penting untuk mempelajari fitur-fiturnya dalam *multiscale framework*.

Hessian Matrix dapat diperoleh dari hasil penghitungan turunan fungsi *Gaussian* (Jimenez-Carretero dkk., 2013). Jika terdapat titik x dengan skala σ , maka *Hessian Matrix* $\mathcal{H}_\sigma(x)$ dapat diperoleh dengan mengikuti Persamaan 2.2.

$$\mathcal{H}_\sigma(I, x) = \frac{\partial^2 I_\sigma}{\partial x^2} = I(x) * \frac{\partial^2 \mathcal{G}_\sigma(x)}{\partial x^2} \quad (2.2)$$

dimana I menyatakan citra dan \mathcal{G}_σ menyatakan fungsi *Gaussian* dengan standar deviasi σ .

Analisis terhadap *Hessian Matrix* dilakukan untuk mengekstraksi *eigenvalues* ($|\lambda_1| \leq |\lambda_2| \leq |\lambda_3|$) dan *principal direction* ($\bar{u}_1, \bar{u}_2, \bar{u}_3$) dimana struktur *local second order* citra didekomposisi. Dari hasil analisis tersebut dapat digunakan untuk membedakan pola-pola interpretasi geometris seperti *blob-like*, *plate-like*, tubular maupun bentuk yang tidak beraturan. Oleh karena pembuluh darah memiliki bentuk tubular, maka struktur ideal dari pembuluh darah mengikuti aturan $|\lambda_1| \approx 0, |\lambda_1| \ll |\lambda_2|, \lambda_2 \approx \lambda_3$. Dengan menggunakan informasi tersebut maka Frangi (1998) mendesain suatu fungsi vessel \mathcal{V}_F^σ yang ditunjukkan Persamaan 2.3 untuk mengukur apakah suatu struktur tertentu mendekati struktur tubular yang ideal.

$$\mathcal{V}_F^\sigma = \begin{cases} 0 & \text{if } \lambda_2 > 0 \text{ or } \lambda_3 > 0 \\ \left(1 - \exp\left(-\frac{\mathcal{R}_A^2}{2\alpha^2}\right)\right) \exp\left(-\frac{\mathcal{R}_B^2}{2\beta^2}\right) \left(1 - \exp\left(-\frac{\mathcal{S}^2}{2c^2}\right)\right) & \end{cases} \quad (2.3)$$

dimana parameter α, β, c mengatur tingkat sensitifitas dari proses filtering. Sedangkan Persamaan 2.4, Persamaan 2.5, dan Persamaan 2.6 masing-masing merupakan persamaan yang digunakan untuk membedakan apakah suatu struktur merupakan *plate-like* (\mathcal{R}_A), *blob-like* (\mathcal{R}_B), atau *background* (\mathcal{S}).

$$\mathcal{R}_A = \frac{|\lambda_2|}{|\lambda_3|} \quad (2.4)$$

$$\mathcal{R}_B = \frac{|\lambda_1|}{|\sqrt{\lambda_2 \lambda_3}|} \quad (2.5)$$

$$\mathcal{S} = \sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2} \quad (2.6)$$

Bentuk akhir dari persamaan untuk mengukur struktur tubular pembuluh darah dengan Frangi Filter ditunjukkan pada Persamaan 2.7.

$$\mathcal{V}_F = \max_{\sigma_{min} \leq \sigma \leq \sigma_{max}} \mathcal{V}_F^\sigma(x) \quad (2.7)$$

2.2.3. Median Filter

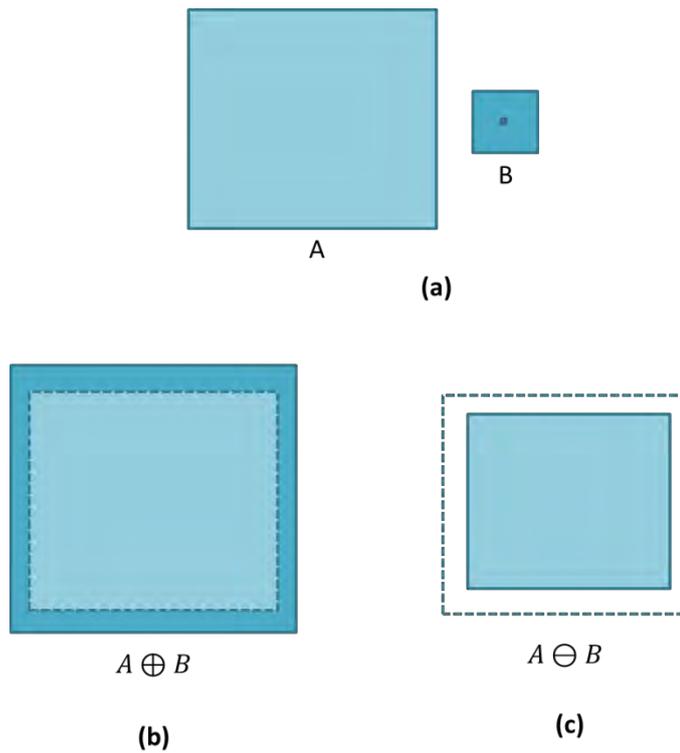
Median Filter merupakan metode penyaringan derau yang dibentuk dari R-Estimator. R-Estimator adalah salah satu *robust estimator* yang penghitungannya didasarkan pada urutan data $x_{(i)}$ dimana $x_{(i)}$ merupakan elemen data pada urutan ke- i dan n menyatakan ukuran himpunan data dengan $1 \leq i \leq n$ (Khusnuliawati dkk., 2014). Bentuk persamaan dari Median Filter $\hat{\theta}_{med}$ dapat dimodelkan dalam Persamaan 2.8.

$$\hat{\theta}_{med} = \begin{cases} \frac{1}{2} (X_{(n/2)} + X_{(1+n/2)}), & \text{for even } n \\ X_{(n+1/2)}, & \text{for odd } n \end{cases} \quad (2.8)$$

dimana jika n merupakan bilangan genap maka nilai Median Filter merupakan setengah dari penjumlahan elemen pada urutan $n/2$ dan elemen pada urutan $1 + n/2$. Sedangkan untuk n merupakan bilangan ganjil maka nilai Median Filter merupakan elemen pada urutan $n + 1/2$.

2.2.4. Operasi Morfologi

Operasi morfologi merupakan metode pengolahan citra yang berbasis pada bentuk objek (Hendriks, 2003). Pada metode ini, digunakan suatu *structuring element* tertentu yang menjadi cetakan bagi citra keluaran dari hasil analisis citra



Gambar 2.2 Ilustrasi operasi morfologi dilasi dan rotasi (a) Citra A merupakan citra asli yang akan diproses dengan operasi morfologi menggunakan structuring element B (b) citra hasil dilasi (c) citra hasil erosi

masukan. Terdapat dua operasi morfologi dasar yaitu dilasi dan erosi. Kedua operasi tersebut dikenakan pada objek yang terdapat pada citra biner. Objek citra direpresentasikan dengan nilai biner 1 sedangkan background bernilai biner 0.

Pada operasi dilasi, objek pada citra biner akan mengembang atau wilayahnya menjadi lebih luas. Sebaliknya pada operasi erosi, objek pada citra biner akan mengerucut atau wilayahnya menjadi lebih sempit. Jika suatu citra A dikenai *structuring element* B dengan β sebagai anggota dari B maka notasi untuk dilasi dan erosi ditunjukkan masing-masing oleh Persamaan 2.9 dan Persamaan 2.10 (Hendriks, 2003).

$$A \oplus B = \bigcup_{\beta \in B} A + \beta \quad (2.9)$$

$$A \ominus B = \bigcap_{\beta \in B} A - \beta \quad (2.10)$$

Gambar 2.2 menunjukkan ilustrasi dari proses dilasi dan erosi. Pada operasi dilasi, setiap piksel pada citra A yang tepat berada pada pusat piksel *structuring element* B maka tetangga dari piksel tersebut mengikuti jumlah dan pola dari ketetanggaan pusat piksel *structuring element* B . Sedangkan pada operasi erosi, setiap piksel pada citra A yang tepat berada pada pusat piksel *structuring element* B maka tetangga dari piksel tersebut akan dihapus mengikuti jumlah dan pola dari ketetanggaan pusat piksel *structuring element* B .

Operasi morfologi dilasi dan rotasi juga dapat saling dikombinasikan membentuk operasi morfologi *closing* dan *opening*. Kedua jenis morfologi ini berfungsi untuk memperhalus kontur objek. Selain itu, operasi *closing* berfungsi menyatukan dua objek yang terpisah, mempertebal area yang tipis, serta mengisi lubang-lubang kecil pada objek. Sedangkan fungsi lain dari operasi *opening* yaitu memisahkan dua objek dan menghapus objek-objek kecil. Notasi dari operasi morfologi *opening* dan *closing*, masing-masing ditunjukkan pada Persamaan 2.11 dan Persamaan 2.12. Pada operasi *opening*, citra A dikenai operasi erosi terlebih dahulu kemudian diikuti operasi dilasi. Sedangkan untuk operasi *closing*, citra A dikenai operasi dilasi terlebih dahulu kemudian diikuti operasi erosi.

$$A \circ B = (A \ominus B) \oplus B \quad (2.11)$$

$$A \cdot B = (A \oplus B) \ominus B \quad (2.12)$$

2.3. Ekstraksi Fitur Pembuluh Darah Jari

Ekstraksi fitur merupakan tahap yang penting pada rangkaian proses pengenalan pembuluh darah jari. Fitur yang dihasilkan akan menjadi informasi utama untuk menentukan apakah citra pembuluh darah jari yang diuji dapat dikenali atau tidak. Sehingga diperlukan fitur yang mampu memberikan informasi dengan hasil pengenalan yang diberikan merupakan hasil optimal.

Apabila melihat kondisi citra pembuluh darah jari yang diuji, maka dapat dilihat bahwa kondisi citra mengalami degradasi yang diakibatkan perubahan skala, rotasi, translasi maupun perbedaan intensitas pencahayaan. Sehingga pada tahap ini, fitur yang diekstraksi diharapkan mampu memberikan informasi yang

memiliki ketahanan terhadap degradasi kondisi tersebut. Pada penelitian ini, fitur yang diekstraksi adalah fitur SIFT dan fitur LEBP. Kedua fitur tersebut diekstraksi dari citra *grayscale* hasil praproses.

2.3.1. Fitur *Scale-Invariant Feature Transform (SIFT)*

Fitur SIFT pertama kali diperkenalkan oleh Lowe (2004) untuk memberikan hasil pencocokan citra yang terpercaya dan memperbaiki kinerja pengenalan objek pada kondisi citra dengan efek perubahan skala, rotasi, dan translasi (Lowe, 2004). Langkah-langkah utama untuk memperoleh fitur SIFT adalah sebagai berikut (Lowe, 2004).

1. Mendeteksi *scale-space extrema* dengan metode *difference-of Gaussian (DoG)*.

Jika ruang skala citra didefinisikan sebagai fungsi $L(x, y, \sigma)$ dengan citra masukan $I(x, y)$ dan faktor skala σ maka ruang skala $L(x, y, \sigma)$ dapat diperoleh dari hasil konvolusi dengan skala variabel Gaussian $G(x, y, \sigma)$ yang mengikuti Persamaan 2.13 dan Persamaan 2.14.

$$L(x, y, \sigma) = G(x, y, \sigma) \cdot I(x, y) \quad (2.13)$$

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad (2.14)$$

Sedangkan untuk mendeteksi *keypoint* secara stabil dengan lebih efisien maka digunakan metode DoG yang mengikuti Persamaan 2.15 dimana merupakan penghitungan dari selisih antara ruang skala terdekat yang terpisah oleh faktor pengali konstan k .

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) \cdot I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned} \quad (2.15)$$

Untuk mendeteksi *local extrema* dari $D(x, y, \sigma)$, setiap titik kemudian dibandingkan dengan delapan tetangga dari citra saat ini dan sembilan tetangga dari citra pada skala di atas maupun dibawah citra saat ini. Titik yang terpilih sebagai *local extrema* merupakan titik dengan nilai yang lebih besar atau lebih kecil jika dibandingkan dengan seluruh tetangga.

2. Menentukan lokasi dan skala dari setiap kandidat *keypoint* yang telah diperoleh dari langkah sebelumnya dengan menyesuaikan model sedetail mungkin sehingga dari informasi yang diperoleh, kandidat *keypoint* yang memiliki kontras rendah, sensitif terhadap derau, dan terletak di sekitar tepi dapat

dieliminasi. Penyesuaian model menggunakan fungsi *3D quadratic* yang mampu meningkatkan akurasi interpolasi. Untuk menghilangkan pengaruh bagian tepi, dilakukan penghitungan menggunakan *Hessian matrix*, H , dengan ukuran 2×2 . Bentuk umum *Hessian matrix* ditunjukkan oleh Persamaan 2.16. Sedangkan standar pengukuran stabilitas *keypoints* untuk menentukan suatu kandidat *keypoint* dihilangkan atau tidak maka mengikuti Persamaan 2.17 dengan nilai r menyatakan perbandingan nilai *eigenvalue* terbesar dan *eigenvalue* terkecil.

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad (2.16)$$

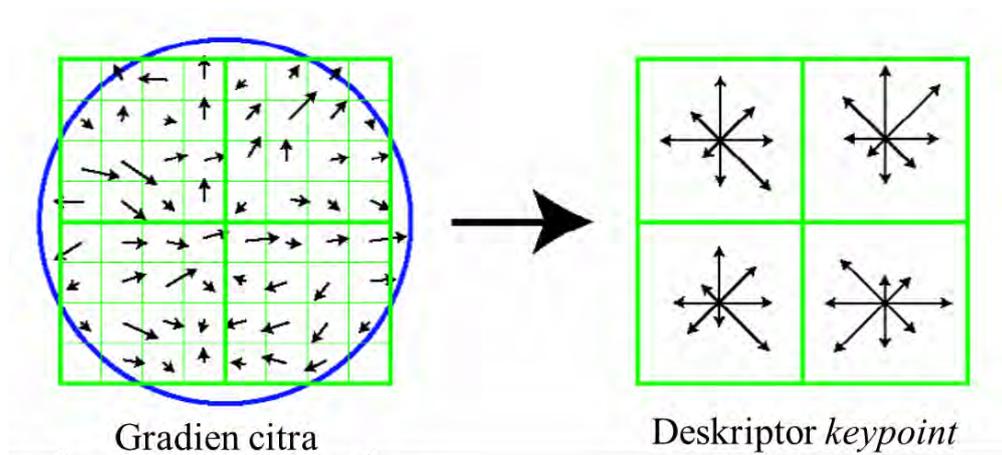
$$\frac{(D_{xx} + D_{yy})^2}{D_{xx}D_{yy} - D_{xy}^2} < \frac{(r + 1)^2}{r} \quad (2.17)$$

3. Menetapkan orientasi dari setiap *keypoint* yang telah diperoleh dengan didasarkan pada arah gradien lokal citra. Besar gradien $m(x,y)$ dan orientasinya $\theta(x,y)$ diperoleh dari hasil penghitungan piksel-piksel pada setiap contoh citra $L(x,y)$ yang mengikuti Persamaan 2.18 dan Persamaan 2.19.

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \quad (2.18)$$

$$\theta(x,y) = \tan^{-1} (L(x+1,y) - L(x-1,y)) / (L(x,y+1) - L(x,y-1)) \quad (2.19)$$

4. Menentukan deskriptor untuk *keypoint* dengan langkah pertama membangun histogram orientasi dengan jumlah 36 bin dari gradien orientasi setiap titik di daerah sekitar *keypoint*. Pembobotan dilakukan untuk setiap titik dengan besar 1.5 kali dari faktor skala σ . Deskriptor untuk setiap *keypoint* yang terbentuk dirangkum dalam sub wilayah berukuran 4×4 dimana setiap sub wilayah tersebut menyimpan informasi orientasi dalam delapan arah mata angin. Gambar 2.3 menunjukkan pembentukan deskriptor *keypoint*. Setiap *keypoint* memiliki panjang vektor fitur 128 bit dengan 8 bin histogram yang digunakan.



Gambar 2.3 Pembentukan deskriptor *keypoint* (Lowe, 2004)

2.3.2. Algoritma *Bag Of Visual Words (BVW)*

BVW merupakan algoritma untuk memperoleh apa yang disebut dengan *visual vocabulary* dengan memanfaatkan deskriptor lokal suatu citra (Haas dkk., 2011). *Vocabulary* diperoleh dengan melakukan proses *clustering* terhadap seluruh deskriptor dari citra untuk pelatihan. Setiap citra yang termasuk dalam data pelatihan maupun data uji direpresentasikan ke dalam bentuk histogram dimana jarak persamaan antar citra dihitung dari jarak persamaan antar histogram.

Tahapan untuk merepresentasikan suatu citra ke dalam bentuk histogram dengan BVW terbagi menjadi dua tahap yaitu tahap pembentukan *visual words* dan tahap penemuan kembali. Untuk tahap pembentukan *visual words*, data citra yang digunakan adalah kumpulan data citra pelatihan $R = I_1, \dots, I_m$ dengan m menyatakan jumlah citra untuk pelatihan. Langkah untuk memperoleh *visual words* ditunjukkan sebagai berikut.

1. Membentuk matriks dengan ukuran $k \times d_{128}$ untuk masing-masing citra dalam R dimana k merupakan jumlah *keypoints* yang diperoleh dari ekstraksi fitur SIFT citra dan d_{128} adalah panjang dimensi dari deskriptor fitur SIFT.
2. Melakukan *clustering* terhadap deskriptor gabungan dari keseluruhan citra pelatihan untuk memperoleh *visual words* dari deskriptor fitur SIFT. *Visual words* merupakan *keypoint* sebagai pusat kluster yang diperoleh dari hasil *clustering* menggunakan algoritma K-Means.

Setelah *visual words* diperoleh, maka memasuki tahap berikutnya yaitu tahap penemuan kembali. Pada tahap ini akan diperoleh representasi histogram dari deskriptor fitur citra. Untuk memperoleh histogram suatu citra I_j dari kumpulan data citra uji $S = I_1, \dots, I_n$ dimana m menyatakan jumlah citra uji maka langkahnya

1. Mengelompokan setiap *keypoint* dari matriks $k \times d_{128}$ untuk masing-masing citra dalam S sebagai anggota dari *visual words*.
2. Memetakan *visual words* sebagai sumbu x histogram yang menunjukkan jumlah bin histogram. Sedangkan sumbu y merupakan frekuensi yang menyatakan jumlah anggota dari setiap pusat kluster sebagai *visual words*.

2.3.3. Fitur *Local Extensive Binary Pattern (LEBP)*

Fitur LEBP merupakan salah satu fitur yang diekstraksi berdasarkan karakteristik lokal suatu citra pembuluh darah jari. Fitur LEBP merupakan kombinasi dari fitur LmBP (*Local Multilayer Binary Pattern*) dan fitur LdBP (*Local Directional Binary Pattern*) dimana fitur LmBP merupakan variasi dari bentuk fitur LBP dan fitur LdBP merupakan modifikasi dari fitur LDC. Kombinasi ini dilakukan karena karakteristik lokal dari citra pembuluh darah jari dapat secara baik ditangkap. Sehingga mampu mewakili karakteristik lokal secara hierarki dan informasi arah karakteristik lokal (Liu dkk., 2014). Fitur LEBP didefinisikan dalam bentuk gabungan vektor biner LmBP dan LdBP. Untuk memperoleh nilai fitur LmBP maupun LdBP dapat dilihat pada sub bab berikut.

a. *Local Multilayer Binary Pattern (LmBP)*

Sebagaimana telah disebutkan sebelumnya, LmBP merupakan variasi dari bentuk fitur *Local Binary Pattern (LBP)*. Apabila LBP menggunakan kernel dengan ukuran 3×3 , maka LmBP menggunakan kernel dengan jumlah $n > 3$. Penggambaran piksel-piksel yang terlibat untuk memperoleh nilai dari fitur LmBP dapat dilihat pada Gambar 2.4. Fitur LmBP atau dapat dituliskan $LmBP_{64} = [v_8^1, v_{16}^2, v_{24}^3, v_8^4, v_8^5]$ merupakan kumpulan vektor yang diperoleh dari dua tahap perbandingan (Liu dkk., 2014), yaitu.

1. Perbandingan tingkat keabuan dari piksel pusat kernel dengan piksel-piksel tetangga pada setiap lapis yang berbeda sehingga diperoleh vektor v^i dengan $i = 1, 2, 3$. Jika merujuk pada Gambar 2.4, maka untuk memperoleh v^1, v^2 dan v^3 adalah dengan membandingkan

$$v_8^1 = [a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7] \& m,$$

$$v_{16}^2 = [b_0, b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8, b_9, b_{10}, b_{11}, b_{12}, b_{13}, b_{14}, b_{15}] \& m,$$

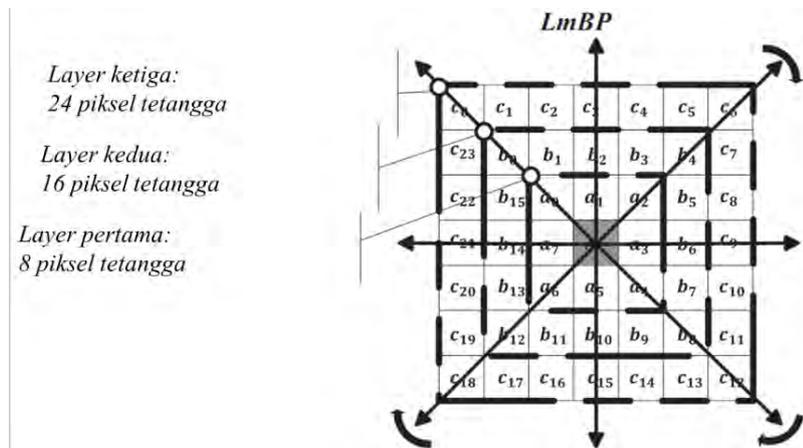
$$v_{24}^3 =$$

$$[c_0, c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8, c_9, c_{10}, c_{11}, c_{12}, c_{13}, c_{14}, c_{15}, c_{16}, c_{17}, c_{18}, c_{19}, c_{20}, c_{21}, c_{22}, c_{23}] \& m$$

2. Perbandingan terhadap tingkat keabuan dari piksel-piksel pada lapisan dalam dengan piksel-piksel pada lapisan yang lebih luar sehingga diperoleh vektor v^i dengan $i = 4, 5$. Jika merujuk pada Gambar 2.4, maka untuk memperoleh v^4 dan v^5 adalah dengan membandingkan

$$v_8^4 = [a_0 \& b_0, a_1 \& b_1, a_2 \& b_2, a_3 \& b_3, a_4 \& b_4, a_5 \& b_5, a_6 \& b_6, a_7 \& b_7],$$

$$v_8^5 = [b_0 \& c_0, b_1 \& c_1, b_2 \& c_2, b_3 \& c_3, b_4 \& c_4, b_5 \& c_5, b_6 \& c_6, b_7 \& c_7],$$



Gambar 2.4 Penggambaran piksel-piksel yang terlibat untuk memperoleh nilai dari fitur LmBP (Liu dkk., 2014)

b. Local Directional Binary Pattern (LdBp)

LdBp merupakan modifikasi dari fitur Local Directional Code (LDC) dimana fitur LDC jika dibandingkan dengan fitur LLBP maka mampu merepresentasikan informasi arah dari fitur lokal dengan lebih baik (Liu dkk., 2014). Penggambaran piksel-piksel yang terlibat untuk memperoleh nilai dari fitur LdBp dapat dilihat pada Gambar 2.5. Komponen v_v merupakan selisih piksel

tetangga dari piksel pusat m secara vertikal dan ditunjukkan oleh Persamaan 2.20. Sedangkan komponen v_h merupakan selisih piksel tetangga dari piksel pusat m secara horizontal dan ditunjukkan oleh Persamaan 2.21. Hasil dari fitur LdBP ini merupakan vektor hasil perbandingan nilai t pada pusat piksel m dengan nilai t piksel-piksel tetangganya, dimana nilai t diperoleh dari Persamaan 2.22 dengan nilai T merupakan jumlah piksel tetangga yang terlibat dan nilai θ' diperoleh dari Persamaan 2.23. Sedangkan nilai orientasi gradien θ diperoleh dari Persamaan 2.24.

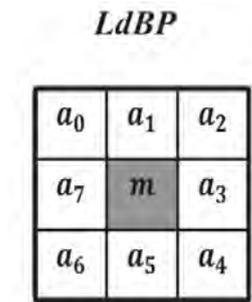
$$v_h = a_7 - a_3 \quad (2.20)$$

$$v_v = a_5 - a_1 \quad (2.21)$$

$$t = \text{mod}\left(\frac{\theta'}{2\pi/T}, T\right) \quad (2.22)$$

$$\theta' = \arctan 2(v_v, v_h) + \pi, \arctan 2(v_v, v_h) = \begin{cases} \theta & v_h > 0, v_v > 0 \\ \pi + \theta & v_h > 0, v_v < 0 \\ \theta - \pi & v_h < 0, v_v < 0 \\ \theta & v_h < 0, v_v > 0 \end{cases}, \theta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \quad (2.23)$$

$$\theta(m) = \arctan \left(\frac{v_h}{v_v}\right) \quad (2.24)$$



Gambar 2.5 Penggambaran piksel-piksel yang terlibat untuk memperoleh nilai dari fitur LdBP (Liu dkk., 2014)

2.4. Learning Vector Quantization (LVQ)

LVQ merupakan metode yang dikembangkan oleh kohonen serta berkaitan erat dengan *Vector Quantization* (VQ) dan *Self Organizing Map* (SOM). Apabila VQ dan *basic* SOM bersifat *unsupervised* maka LVQ bersifat *supervised*. LVQ dibagi menjadi tiga algoritma yaitu LVQ-1, LVQ-2, dan LVQ-3. Pada penelitian

ini digunakan LVQ-1 (Bashyal & Venayagamoorthy, 2008) dimana pada algoritma LVQ-1 terdapat dua *layer* yaitu *competitive layer* atau disebut juga *sub classes* dan *output layer*. Pada setiap *competitive layer* diinisialisasi suatu vektor pembobotan yang mirip dengan vektor masukan. Dari setiap *neuron* yang ada pada *competitive layer* maka dilakukan pencarian *winning neuron* yaitu *neuron* dengan kecocokan terbaik. Berikutnya, kelas pada *output layer* yang berhubungan dengan *winning neuron* ditetapkan memiliki nilai tinggi. Vektor pembobotan kemudian diperbarui dengan mengikuti Persamaan 2.25.

$$w_i(n) = \begin{cases} w_i(n-1) + \alpha(p - w_i(n-1)) & \text{jika hasil klasifikasi benar} \\ w_i(n-1) - \alpha(p - w_i(n-1)) & \text{jika hasil klasifikasi salah} \end{cases} \quad (2.25)$$

Pada persamaan di atas, $w_i(n)$ menyatakan besar bobot ke- i pada iterasi ke- n , α menyatakan besar *learning rate*, sedangkan p adalah vektor masukan.

[Halaman ini sengaja dikosongkan]

BAB III

METODOLOGI PENELITIAN

Bab ini memaparkan metodologi penelitian yang akan digunakan. Pada bagian ini akan dijelaskan alur dari metodologi penelitian yang terdiri dari studi literatur, perancangan metode, implementasi metode, pengujian metode dan analisa hasil uji coba. Selanjutnya dijelaskan rancangan sistem beserta detailnya.

3.1 Alur Metodologi Penelitian

Penelitian ini terdiri dari beberapa tahap yang akan dilakukan. Adapun alur metodologi penelitian yang digunakan dilihat pada Gambar 3.1.



Gambar 3.1 Alur Metodologi Penelitian

3.1.1 Studi Literatur Ekstraksi Fitur SIFT

Studi literatur tentang fitur SIFT dilakukan untuk dapat memahami bagaimana fitur ini diekstraksi dari suatu citra dan implementasinya pada citra pembuluh darah jari. Pengumpulan literatur dilakukan dengan mencari jurnal terkait fitur SIFT yaitu mulai dari pembentukan fitur SIFT, ekstraksi fitur SIFT dan bagaimana implementasinya pada kasus pembuluh darah jari. Dari hasil literatur yang telah terkumpul, langkah berikutnya yaitu mempelajari dan memahami secara mendalam literatur terkait sehingga dapat melakukan perancangan ekstraksi fitur SIFT beserta implementasinya pada kasus pembuluh darah jari.

3.1.2 Studi Literatur Ekstraksi Fitur LEBP

Studi literatur tentang fitur LEBP dilakukan untuk dapat memahami bagaimana fitur ini diekstraksi dan implementasinya pada kasus pembuluh darah jari. Langkah-langkah yang dilakukan pada tahap ini meliputi pencarian literatur dari jurnal berkaitan dengan LEBP. Setelah literatur yang berkaitan dengan fitur ini terkumpul, langkah berikutnya yaitu mempelajari secara mendalam literatur terkait sehingga dapat melakukan perancangan ekstraksi fitur beserta implementasinya pada kasus pembuluh darah jari.

3.1.3 Studi Literatur Metode Klasifikasi LVQ

Studi literatur dari metode klasifikasi LVQ dilakukan untuk dapat memahami bagaimana cara kerja metode ini, parameter-parameter yang perlu ditetapkan, serta bagaimana implementasinya untuk kasus pencocokan antara dua citra pembuluh darah jari. Literatur yang digunakan sebagai acuan dicari dari jurnal-jurnal terkait penggunaan metode LVQ terutama untuk kasus biometrika. Dari hasil literatur yang telah terkumpul, langkah berikutnya yaitu mempelajari dan memahami secara mendalam literatur terkait sehingga dapat merancang dan mengimplementasikan metode LVQ untuk kasus pencocokan antara dua citra pembuluh darah jari.

3.1.4 Perancangan Ekstraksi Fitur SIFT-LEBP berbasis *Multi-Feature Fusion*

Pada tahap ini akan dibangun suatu model *multi-feature fusion* menggunakan fitur SIFT dan fitur LEBP. Analisis lebih lanjut terkait kedua fitur

tersebut juga akan dilakukan agar hasil fusi antara kedua fitur tersebut dapat menghasilkan fitur akhir yang optimal untuk diterapkan pada kasus pengenalan pembuluh darah jari. Tahap ini merupakan tahap yang akan menghasilkan kontribusi berupa integrasi fitur *robust*.

3.1.5 Perancangan Sistem Untuk Mendeteksi PDR Perancangan Sistem Pengenalan Pembuluh Darah Jari

Tahap perancangan sistem pengenalan pembuluh darah jari merupakan keseluruhan tahap-tahap yang perlu dilakukan agar tujuan pembuatan sistem dapat tercapai dengan baik. Rancangan sistem yang akan dibangun menggambarkan alur proses bagaimana sistem mengolah citra masukan hingga mengeluarkan hasil akhir yang diharapkan. Detail dari rancangan sistem secara keseluruhan dibahas pada sub bab berikutnya.

3.1.6 Implementasi Rancangan Sistem Pencocokkan Citra Pembuluh Darah Jari

Tahapan ini dilakukan untuk mengimplementasikan metode yang diusulkan ke dalam perangkat lunak. Proses implementasi meliputi penulisan kode program, *compiling*, *debugging* hingga dihasilkan kode program yang siap dieksekusi.

Sebelum implementasi perlu dilakukan perencanaan sistem, hal ini bertujuan untuk mencari bentuk yang optimal dari aplikasi yang akan dibangun dengan mempertimbangkan berbagai faktor permasalahan dan kebutuhan yang ada pada sistem. Upaya yang dilakukan adalah dengan berusaha mencari kombinasi penggunaan teknologi dan perangkat lunak (*software*) yang tepat sehingga diperoleh hasil yang optimal dan mudah untuk diimplementasikan. Pada penelitian ini akan digunakan pemrograman Matlab dengan memanfaatkan *library* yang sudah disediakan.

3.1.7 Uji Coba dan Analisis Hasil Uji Metode Pengenalan

Uji coba yang dilakukan terdiri dari uji coba untuk melihat tingkat keberhasilan dari hasil proses pengenalan pembuluh darah jari menggunakan fitur-fitur yang berbeda. Fitur-fitur yang digunakan yaitu fitur SIFT, fitur LmBP, fitur LdBP, fitur LEBP, dan *fusion* dari fitur SIFT dengan fitur LEBP.

Hasil dari pengenalan citra pembuluh darah jari dapat dikategorikan ke dalam dua kelas yaitu citra yang berasal dari individu yang sama (*within class*) dan citra yang berasal dari dua individu yang berbeda (*between class*). Pada penelitian ini, metode yang digunakan untuk mengenali citra pembuluh darah jari adalah metode LVQ. Metode LVQ akan mengklasifikasikan suatu citra uji merupakan anggota dari kelas individu yang mana dimana jumlah kelas ditentukan sesuai jumlah individu yang citra pembuluh darahnya dijadikan sebagai data citra pelatihan.

Untuk melihat kinerja dari sistem yang telah dibuat, maka evaluasi kinerja dilakukan menggunakan *confussion matrix* yang dapat dilihat pada Tabel 3.1. Nilai *True Positive* (TP) menyatakan jumlah data yang secara aktual berada pada kelas sama dan dalam hasil pengenalan juga dinyatakan berada pada kelas sama. Sedangkan nilai *False Positive* (FP) menyatakan jumlah data yang secara aktual berada pada kelas berbeda tetapi dari hasil prediksi dinyatakan berada pada kelas sama. Nilai *True Negative* (TN) menyatakan jumlah data yang secara aktual berada pada kelas berbeda dan dalam hasil prediksi juga dinyatakan berada pada kelas berbeda. Sedangkan nilai *False Negative* (FN) menyatakan jumlah data yang secara aktual berada pada kelas sama tetapi dari hasil prediksi dinyatakan berada pada kelas berbeda.

Tabel 3.1 *Confussion matrix* untuk memperoleh nilai FAR dan FRR

Prediksi	Aktual	
	<i>Within Class</i>	<i>Between Class</i>
<i>Within Class</i>	<i>True Positive (True Accept, Hit)</i>	<i>False Positive (False Accept, False Alarm)</i>
<i>Between Class</i>	<i>False Negative (False Reject, Miss)</i>	<i>True Negative (True Reject)</i>

Nilai akurasi, *false positive rate* (FPR) dan *true positive rate* (TPR) digunakan sebagai ukuran kinerja sistem. Nilai tersebut diperoleh dari hasil analisis *confussion matrix*. Dari hasil analisis yang dilakukan maka diperoleh persamaan untuk menghitung akurasi yang ditunjukkan Persamaan 3.1., TPR yang ditunjukkan Persamaan 3.2 dan FPR yang ditunjukkan Persamaan 3.3.

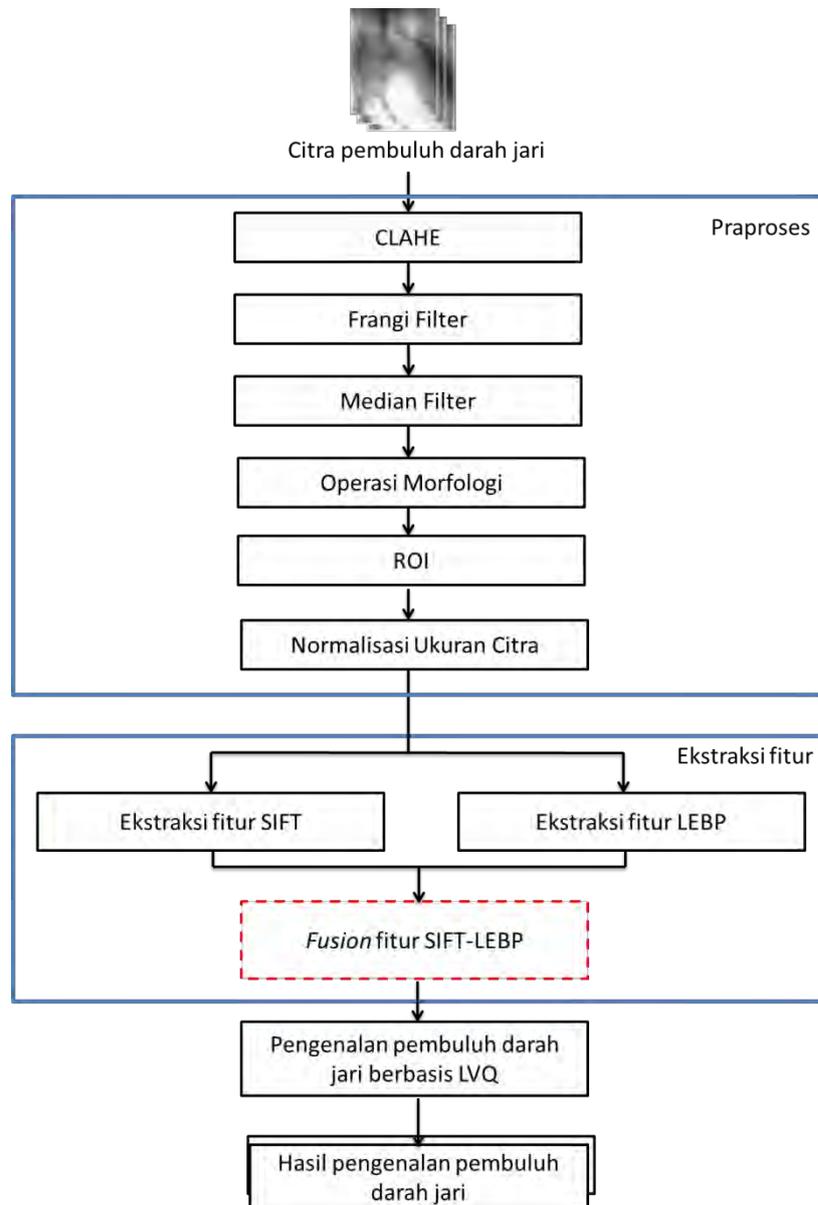
$$\text{akurasi} = \frac{TP + TN}{TP + FN + FP + TN} \quad (3.1)$$

$$\text{TPR} = \frac{TP}{TP + FN} \quad (3.2)$$

$$\text{FPR} = \frac{FP}{FP + TN} \quad (3.3)$$

3.2 Rancangan Sistem

Tahap perancangan sistem pengenalan pembuluh darah jari merupakan keseluruhan tahap-tahap yang perlu dilakukan agar tujuan pembuatan sistem dapat tercapai dengan baik. Rancangan sistem yang akan dibangun menggambarkan alur

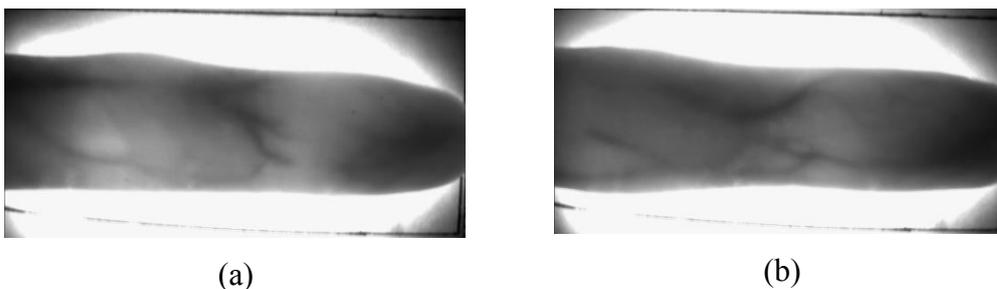


Gambar 3.2 Alur rancangan sistem pengenalan pembuluh darah jari

proses bagaimana sistem mengolah citra masukan hingga mengeluarkan hasil akhir yang diharapkan. Gambar 3.2 menunjukkan rancangan sistem secara keseluruhan. Sistem menerima masukan berupa citra pembuluh dari yang ditunjukkan pada Gambar 2.1 dalam format grayscale. Citra tersebut kemudian melalui tahap praproses untuk memperbaiki kualitas citra serta normalisasi ukuran citra. Dari tahap praproses akan diperoleh citra grayscale hasil perbaikan dan citra biner pola pembuluh darah. Citra grayscale hasil tahap praproses kemudian masuk ke tahap ekstraksi fitur. Fitur yang diekstraksi dari citra tersebut yaitu fitur SIFT dan fitur LEBP. Citra biner digunakan sebagai mask untuk ekstraksi fitur LEBP. Sehingga, piksel-piksel yang diproses pada tahap ekstraksi fitur LEBP hanya piksel-piksel yang terdeteksi sebagai pembuluh darah. Dari proses ekstraksi kedua fitur, maka akan diperoleh dua jenis histogram yang menjadi representasi dari masing-masing fitur. Proses fusion kemudian dilakukan untuk mendapatkan fitur SIFT-LEBP sebagai masukan pada proses pengenalan pembuluh darah jari dengan metode LVQ. Detail dari sistem yang dibangun akan dijelaskan sebagai berikut.

3.2.1 Citra Retina Dataset Citra Pembuluh Darah Jari

Dataset yang digunakan sebagai data uji dari sistem yang dibangun adalah dataset *finger-vein* dari *Hong Kong Polytechnic University* (Kumar & Zhou, 2012). Dataset tersebut diambil dari 156 orang dimana citra jari yang diambil adalah jari telunjuk dan jari tengah. Pengambilan citra dari masing-masing jari dilakukan sebanyak enam kali. Sehingga untuk setiap individu akan memiliki 6 citra untuk masing-masing jari yang mewakili satu sesi pengambilan. Sesi pengambilan citra terbagi menjadi dua dengan rentang interval pengambilan antara satu bulan hingga enam bulan dengan rata-rata 66,8 hari. Gambar 3.3



Gambar 3.3 Contoh citra pembuluh darah jari (a) jari telunjuk (b) jari tengah

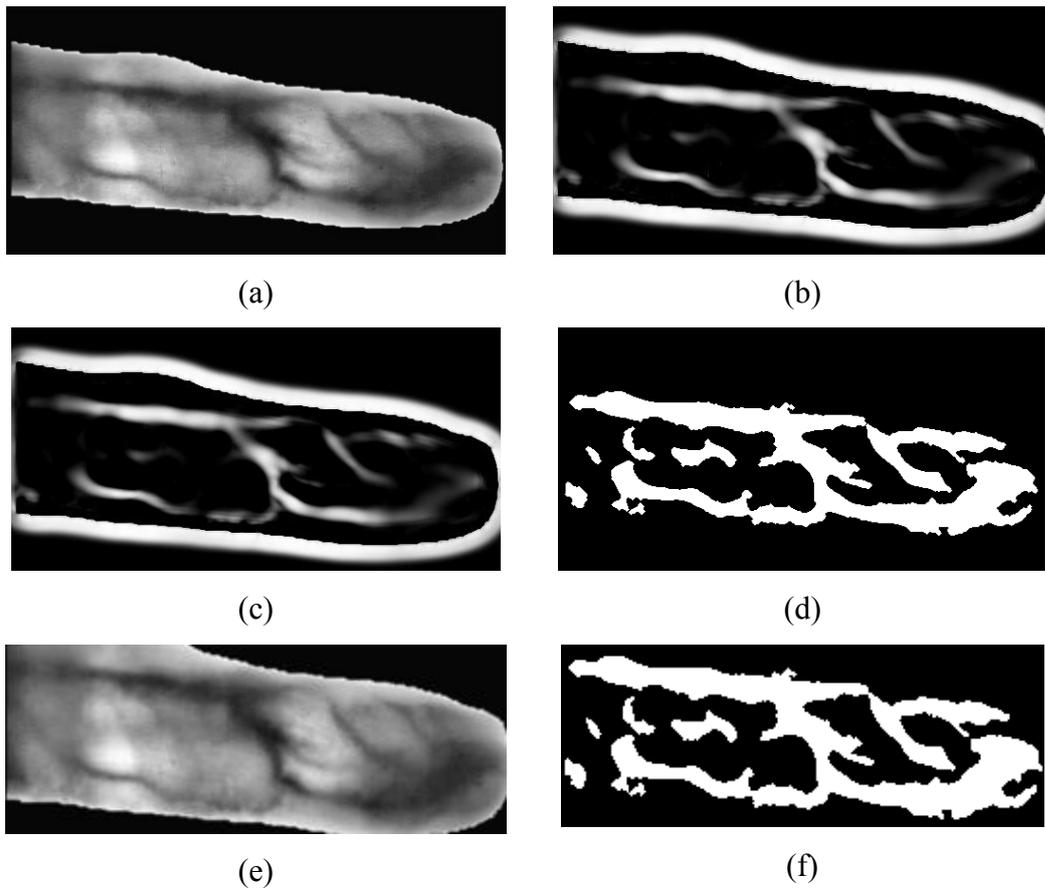
menunjukkan contoh citra pembuluh darah jari salah satu individu yang diambil dari dataset.

Pada proses pengenalan pembuluh darah jari, total citra pembuluh darah jari yang digunakan dalam uji coba berjumlah 300 citra yang diambil dari 50 individu dengan masing-masing individu terdiri dari enam citra. Sebelum melakukan pengujian, data masukan terlebih dahulu dibagi menjadi data latihan (*training*) dan data uji (*testing*). Data citra yang digunakan sebagai data *training* sejumlah 200 citra dimana setiap individu diwakili oleh empat citra. Sedangkan untuk data *testing*, data yang digunakan sejumlah 100 citra dimana setiap individu diwakili oleh dua citra.

3.2.2 Praproses

Pada tahap ini dilakukan beberapa proses dengan menggunakan metode yang telah dijelaskan pada sub bab 2.2. Urutan langkah praproses yang dilakukan ditunjukkan pada Gambar 3.2 dimana metode yang digunakan untuk perbaikan citra meliputi CLAHE, frangi filter, median filter, dan operasi morfologi. Citra pembuluh darah jari yang menjadi masukan untuk tahap praproses memiliki kontras rendah dan rentan terhadap derau sehingga pola pembuluh darah menjadi sulit terdeteksi. Metode CLAHE menghasilkan citra dengan kontras yang dapat memperjelas perbedaan antara pola pembuluh darah jari dengan latar belakang citra. Sedangkan Frangi Filter menghasilkan citra yang merepresentasikan pola pembuluh darah jari secara lebih detail yaitu mendeteksi pola pembuluh darah yang kecil dan mengurangi proyeksi dari jaringan lain. Derau pada citra kemudian dikurangi menggunakan median filter. Tahap berikutnya yaitu operasi morfologi dimana pada tahap ini dihasilkan citra biner yang menggambarkan pola pembuluh darah jari. Gambar 3.4 menunjukkan citra hasil dari setiap tahap praproses.

Citra hasil tahap praproses yang digunakan pada tahap ekstraksi fitur yaitu citra grayscale hasil metode CLAHE dan citra biner hasil operasi morfologi. Kedua jenis citra tersebut kemudian masuk ke tahap pemilihan ROI untuk mengurangi bagian citra yang termasuk background. Tahap praproses dilanjutkan dengan normalisasi ukuran citra menjadi 250x95 piksel. Pseudocode dari metode-



Gambar 3.4 Citra hasil tahap praproses (a) Citra hasil CLAHE (b) Citra hasil Frangi Filter (c) Citra hasil Median Filter (d) Citra hasil Operasi Morfologi (e) Citra grayscale hasil CLAHE yang telah melalui tahap cropping dan normalisasi ukuran citra (f) Citra biner hasil operasi morfologi yang telah melalui tahap cropping dan normalisasi ukuran citra

metode tersebut ditunjukkan pada Gambar 3.5. Sedangkan variabel serta fungsi yang terlibat pada tahap praproses ini masing-masing ditunjukkan pada Tabel 3.2 dan Tabel 3.3.

Tabel 3.2 Daftar Variabel yang Digunakan pada Tahap Praproses

No.	Nama Data	Tipe Data	Keterangan
1.	claheRs	uint8	Citra grayscale hasil dari proses perbaikan kontras dengan CLAHE.
2.	frangiv	uint8	Citra grayscale hasil dari proses frangi filter.

Tabel 3.2 Daftar Variabel yang Digunakan pada Tahap Praproses

No.	Nama Data	Tipe Data	Keterangan
3.	val	Double	Nilai skalar pada rentang 0 hingga 1 yang berfungsi sebagai batas maksimum untuk mengatur tingkat kontras citra.
4.	medRs	uint8	Citra grayscale hasil dari proses median filter
5.	invm	Logical	Citra biner hasil proses invertimg dari citra hasil median filter.
6.	se	Object	Objek yang digunakan sebagai structuring element pada proses operasi morfologi.
7.	R	Integer	Nilai radius suatu structuring element yang bernilai positif dan merupakan kelipatan 3.
8.	iclose	Logical	Citra biner yang merupakan hasil proses operasi morfologi closing.
9.	invbiner	Logical	Citra biner hasil proses invertimg dari citra hasil closing.
10.	iopen	Logical	Citra biner yang merupakan hasil proses operasi morfologi opening.
11.	P	integer	Nilai minimum dari ukuran piksel suatu objek pada citra biner yang akan dihapus dengan operasi morfologi opening.

Tabel 3.3 Daftar Fungsi yang Digunakan pada Tahap Ekstraksi Fitur SIFT

No.	Nama Fungsi	Keterangan
1.	adapthisteq	Fungsi untuk melakukan proses CLAHE pada citra.
2.	FrangiFilter2D	Fungsi untuk melakukan proses <i>filtering</i> citra dengan metode frangi.
3.	medfilt2	Fungsi untuk melakukan proses median filter pada suatu citra.
4.	imclose	Fungsi untuk melakukan proses morfologi <i>closing</i> pada suatu citra.
5.	bwareaopen	Fungsi untuk melakukan proses morfologi <i>opening</i> pada suatu citra.

Pseudocode 3. 1 Praproses Pengenalan Pembuluh Darah Jari	
Masukan	Citra pembuluh darah jari dalam format grayscale
Keluaran	Citra dalam format grayscale untuk hasil CLAHE, Frangi Filter, dan median filter Citra dalam format biner untuk hasil operasi morfologi
<pre> 1. claheRs ← adapthisteq(img, 'clipLimit', val) 2. frangiv ← FrangiFilter2D(claheRs) 3. medRs ← medfilt2(frangiv) 4. invm ← invert(medRs) 5. se ← strel('disk', R) 6. iclose ← imclose(invm, se) 7. invbiner ← invert(iclose); 8. iopen ← bwareaopen(invbiner, P) </pre>	

Gambar 3.5 Pseudocode dari Tahap Praproses

3.2.3 Ekstraksi Fitur SIFT

Fitur pertama yang diekstraksi dari citra pembuluh darah jari adalah fitur SIFT. Citra *grayscale* dari tahap praproses menjadi citra masukan untuk proses ekstraksi fitur SIFT. Dari hasil ekstraksi fitur SIFT akan diperoleh kumpulan *keypoint* dimana proses untuk memperoleh *keypoint* tersebut telah dijelaskan pada sub bab 13. Setiap citra akan menghasilkan jumlah *keypoint* yang berbeda-beda. Gambar 3.6 menunjukkan lokasi *keypoints* yang berhasil diekstraksi dari suatu citra. Masing-masing *keypoint* direpresentasikan dalam suatu nilai vektor sepanjang 1×128 sehingga jika suatu citra I memiliki *keypoint* sebanyak k maka fitur SIFT dari citra I memiliki panjang $k \times 128$. Gambar 3.7 merupakan *pseudocode* dari proses ekstraksi fitur. Sedangkan variabel serta fungsi yang terlibat pada tahap ekstraksi fitur SIFT masing-masing ditunjukkan pada Tabel 3.4 dan Tabel 3.5.

3.2.4 Ekstraksi Fitur LEBP

Fitur berikutnya yang diekstraksi dari citra *grayscale* hasil praproses adalah fitur LEBP. Untuk mendapatkan fitur LEBP, tidak semua piksel pada citra



Gambar 3.6 Kumpulan *keypoints* yang diekstraksi dari citra *grayscale*

yang diproses. Pixel-pixel yang diproses hanyalah pixel yang dideteksi sebagai pembuluh darah. Citra biner hasil praproses yang ditunjukkan pada Gambar 3.4 digunakan sebagai *masking* untuk proses ekstraksi fitur tersebut. Proses untuk memperoleh fitur LEBP ditunjukkan sesuai langkah pada sub bab 2.3.3.

Tabel 3.4 Daftar Variabel yang Digunakan pada Tahap Ekstraksi Fitur SIFT

No.	Nama Data	Tipe Data	Keterangan
1.	data_concat	uint8	Vektor untuk menyimpan deskriptor SIFT dari keseluruhan citra pelatihan.
2.	I	single	Citra dalam format single hasil konversi citra masukan.
3.	f	double	Matriks yang menyimpan fitur dari frame suatu <i>keypoint</i> , terdiri dari posisi x,y dan skala serta orientasi.
4.	d	uint8	Deskriptor dari hasil ekstraksi fitur SIFT dalam 128 dimensi vektor
5.	listpos	double	Vektor yang menyimpan nilai posisi x dan y dari <i>keypoint</i>
6.	markerInserter	Object	Objek yang menyimpan penanda lokasi dari suatu citra. Berupa bentuk beserta warna
7.	RGB	uint8	Citra hasil konversi citra grayscale ke RGB
8.	Pts	integer	Nilai yang menyimpan hasil konversi integer dari vektor listpos
9.	J	uint8	Citra RGB dari citra masukan yang telah dilengkapi penanda dati lokasi <i>keypoint</i> .
10.	shape	string	Bentuk penanda dari lokasi <i>keypoint</i> yang dapat berupa <i>x-shape</i> , <i>circle-shape</i> , dll.
11.	color	string	Warna penanda dari lokasi <i>keypoint</i> .

Tabel 3.5 Daftar Fungsi yang Digunakan pada Tahap Ekstraksi Fitur SIFT

No.	Nama Fungsi	Keterangan
1.	vl_sift	Fungsi untuk melakukan proses ekstraksi fitur SIFT
2.	vision.MarkerInserter	Fungsi untuk membuat penanda dari lokasi tertentu suatu citra
3.	repmat	Fungsi untuk mengubah dimensi citra
4.	step	Fungsi untuk memasukkan penanda dari lokasi tertentu suatu citra

Pseudocode 3. 2 Ekstraksi Deskriptor Fitur SIFT	
Masukan	Citra pembuluh darah dalam format grayscale jari hasil praproses
Keluaran	Vektor representasi histogram dari deskriptor fitur SIFT
<pre> 1. data_concat ← ∅ 2. for i ← 1 to lenTrain 3. I ← single(img) 4. [f,d] ← vl_sift(I) 5. data_concat[i] ← d 6. end 7. listpos ← f(1,2) 8. markerInserter ← vision.MarkerInserter(shape, color) 9. RGB ← repmat(img, [1 1 3]) 10. Pts ← int32(listpos); 11. J ← step(markerInserter, RGB, Pts) </pre>	

Gambar 3.7 Pseudocode dari Tahap Ekstraksi Fitur SIFT

Fitur LEBP diperoleh dari kombinasi fitur LmBP dan fitur LdBP yang direpresentasikan dalam 72 dimensi vektor fitur biner $LEBP_{72} = [LmBP_{64}, LdBP_8]$. *Pseudocode* dari proses ekstraksi fitur LEBP ditunjukkan pada Gambar 3.8. Seperti yang telah dijelaskan pada bab sebelumnya, fitur LEBP merupakan gabungan dari fitur LmBP dan LdBP. *Pseudocode* dari ekstraksi fitur LmBP dan fitur LdBP ditunjukkan pada Gambar 3.10 dan Gambar 3.11. Sedangkan untuk proses penghitungan LDC yang merupakan bagian dari fitur LdBP ditunjukkan pada Gambar 3.9. Seluruh variabel serta fungsi yang terlibat pada tahap ekstraksi fitur LEBP ini masing-masing ditunjukkan pada Tabel 3.6 dan Tabel 3.7.

Tabel 3.6 Daftar Variabel yang Digunakan pada Tahap Ekstraksi Fitur LEBP

No.	Nama Data	Tipe Data	Keterangan
1.	d_gradient_orientation	double	Matriks yang menyimpan nilai hasil penghitungan LDC untuk setiap piksel citra.
2.	ysize	integer	Nilai yang menyimpan jumlah baris dari citra.
3.	xsize	integer	Nilai yang menyimpan jumlah kolom dari citra.
4.	mask	logical	Citra bertipe biner dari pola pembuluh darah jari.
5.	lmbpVector	logical	Vektor yang menyimpan kode biner dari hasil ekstraksi fitur LmBP.
6.	lmbpBinerValue	integer	Vektor yang menyimpan hasil konversi dari setiap 8 bit kode biner LmBP fitur ke dalam nilai desimal.
7.	lmbpMatrix	uint8	Vektor yang menyimpan kode biner dari hasil ekstraksi fitur Ldbp.
8.	ldbMatrix	uint8	Vektor yang menyimpan hasil konversi dari setiap 8 bit kode biner Ldbp fitur ke dalam nilai desimal.
9.	window_image	double	Matriks dengan ukuran $m \times n$ yang merupakan kumpulan piksel dari citra yang masuk pada area <i>window</i> .
10.	N1	double	Nilai piksel yang tepat berada pada arah 90° piksel pusat <i>window</i> .
11.	N5	double	Nilai piksel yang tepat berada pada arah 270° piksel pusat <i>window</i> .

Tabel 3.6 Daftar Variabel yang Digunakan pada Tahap Ekstraksi Fitur LEBP

No.	Nama Data	Tipe Data	Keterangan
12.	N3	double	Nilai piksel yang tepat berada pada arah 0^0 piksel pusat <i>window</i> .
13.	N7	double	Nilai piksel yang tepat berada pada arah 180^0 piksel pusat <i>window</i> .
14.	v_v	double	Nilai yang menyimpan selisih dari piksel tetangga pada arah vertikal dari piksel pusat <i>window</i> .
17.	tetha02	double	Nilai yang menyimpan hasil pemrosesan nilai tetha dengan kondisi-kondisi yang memenuhi syarat.
18	den	double	Nilai yang menyimpan hasil pemrosesan tetha02 untuk memperoleh nilai LDC.
19.	v_1_8	double	Vektor yang menyimpan pola indeks piksel yang terlibat untuk memperoleh vektor v_8^1 .
20.	v_2_16	double	Vektor yang menyimpan pola indeks piksel yang terlibat untuk memperoleh vektor v_{16}^2 .
20.	v_3_24	double	Vektor yang menyimpan pola indeks piksel yang terlibat untuk memperoleh vektor v_{24}^3 .
21.	v_4_8	double	Vektor yang menyimpan pola indeks piksel yang terlibat untuk memperoleh vektor v_8^4 .

Tabel 3.6 Daftar Variabel yang Digunakan pada Tahap Ekstraksi Fitur LEBP

No.	Nama Data	Tipe Data	Keterangan
22.	v_5_8	double	Vektor yang menyimpan pola indeks piksel yang terlibat untuk memperoleh vektor v_8^5 .
23.	pixel_center	double	Nilai piksel yang berada pada pusat <i>window</i>
24.	P1	integer	Posisi baris dari piksel pusat <i>window</i>
27.	v_3_24bin	integer	Vektor yang menyimpan hasil konversi dari setiap 8 bit kode biner dari v_{24}^3 .
28.	v_4_8bin	integer	Vektor yang menyimpan hasil konversi dari setiap 8 bit kode biner dari v_8^4 .
29.	v_5_8bin	integer	Vektor yang menyimpan hasil konversi dari setiap 8 bit kode biner dari v_8^5 .
30.	ldbp8	double	Vektor yang menyimpan pola indeks piksel yang terlibat untuk memperoleh fitur LdBp.
31.	ldbp8vec	logical	Vektor yang menyimpan kode biner dari fitur LdBp.

Tabel 3.7 Daftar Fungsi yang Digunakan pada Tahap Ekstraksi Fitur LEBP

No.	Nama Fungsi	Keterangan
1.	get_gradient_orientation	Fungsi untuk memperoleh nilai LDC dari piksel yang diproses.
2.	atan	Fungsi untuk memperoleh hasil arc tangent dari suatu nilai tertentu.
3.	lmbp_feature	Fungsi untuk melakukan proses ekstraksi fitur LmBP.
4.	ldbp_feature	Fungsi untuk melakukan proses ekstraksi fitur LdBp.

Pseudocode 3. 3 Ekstraksi Deskriptor Fitur LEBP	
Masukan	Citra pembuluh darah dalam format grayscale jari hasil praproses, citra biner pembuluh darah
Keluaran	Vektor representasi histogram dari deskriptor fitur LmBP, LdbP, LEBP
<pre> 1. d_gradient_orientation ← get_gradient_orientation(img) 2. for i ← 2 to ysize-1 3. for j ← 2 to xsize-1 4. if mask(i,j) = 1 5. [lmbpVector lmbpBinerValue] ← lmbp_feature(img, 6. i, j) 7. [ldbpVector ldbpBinerValue] ← ldbp_feature(img, 8. i, j) 9. lmbpMatrix(i,j) ← lmbpBinerValue 10. ldbpMatrix(i,j) ← ldbpBinerValue 11. else 12. lmbpMatrix(i,j) ← 0 13. ldbpMatrix(i,j) ← 0 14. end 15. end 16. end </pre>	

Gambar 3.8 Pseudocode dari Tahap Ekstraksi Fitur LEBP

Pseudocode 3. 4 Proses Penghitungan Nilai LDC	
Masukan	Matriks bertipe double hasil konversi citra grayscale dari praproses
Keluaran	Matriks bertipe double hasil penghitungan nilai Local Directional Code (LDC)
<pre> 1. EPSILON ← 0.0000001 2. PI ← 3.141592653589 3. T ← 8 4. for y ← 2 to ysize-1 5. for x ← 2 to xsize-1 6. N1 ← window_image(y-1,x) 7. N5 ← window_image(y+1,x) 8. N3 ← window_image(y,x+1) 9. N7 ← window_image(y,x-1) 10. if (N5-N1 < EPSILON) 11. d_gradient_orientation(y,x) ← 0 12. else 13. v_v ← N5-N1 14. v_h ← N7-N3 15. tetha ← atan(v_h/v_v) 16. if (v_h > EPSILON and v_v > EPSILON) 17. tetha02 ← tetha + PI 18. elseif (v_h > EPSILON and v_v < EPSILON) 19. tetha02 ← tetha + (2*PI) </pre>	

Pseudocode 3. 5 Proses Penghitungan Nilai LDC	
Masukan	Matriks bertipe double hasil konversi citra grayscale dari praproses
Keluaran	Matriks bertipe double hasil penghitungan nilai Local Directional Code (LDC)
20.	<code>elseif (v_h < EPSILON and v_v < EPSILON)</code>
21.	<code>tetha02 ← tetha</code>
22.	<code>elseif (v_h < EPSILON and v_v > EPSILON)</code>
23.	<code>tetha02 ← tetha + PI</code>
24.	<code>end</code>
25.	<code>den ← (tetha02/((2*PI)/T)) + 0.5</code>
26.	<code>d_gradient_orientation(y,x) ← mod(den, T)</code>
27.	<code>end</code>
28.	<code>end</code>
29.	<code>end</code>

Gambar 3.9 Pseudocode dari Tahap Penghitungan nilai LDC

3.2.5 Fusion Fitur SIFT-LEBP

Untuk dapat melakukan proses *fusion*, fitur SIFT dan fitur LEBP perlu diubah ke dalam bentuk representasi vektor yang serupa. Representasi vektor yang digunakan yaitu histogram. Kedua fitur yang telah diekstraksi masing-masing diubah ke dalam vektor histogram dengan proses secara terpisah. Gambar 3.12 menunjukkan diagram alur dari *fusion* fitur SIFT-LEBP.

Proses untuk memperoleh deskriptor fitur SIFT dalam representasi histogram dilakukan dengan menggunakan kumpulan *keypoint* sebagai acuan. Kumpulan *keypoint* tersebut diproses dengan algoritma BVW yang telah dijelaskan pada sub bab 2.3.2 sehingga dihasilkan deskriptor fitur SIFT dalam bentuk vektor histogram.

Pseudocode 3. 6 Ekstraksi Deskriptor Fitur LmBP	
Masukan	Citra grayscale hasil praproses, nilai indek posisi baris dan kolom dari piksel yang diuji
Keluaran	Vektor biner dari fitur LmBP, vektor hasil konversi setiap 8 bit nilai biner fitur LmBP ke dalam nilai desimal
1.	Inisialisasi matriks <code>v_1_8</code> , <code>v_2_16</code> , <code>v_3_24</code> , <code>v_4_8</code> , <code>v_5_8</code> sebagai petunjuk lokasi piksel tetangga yang terlibat pada proses
2.	<code>pixel_center ← img(p1,p2)</code>

Pseudocode 3. 7 Ekstraksi Deskriptor Fitur LmBP

Masukan	Citra grayscale hasil praproses, nilai indek posisi baris dan kolom dari piksel yang diuji
Keluaran	Vektor biner dari fitur LmBP, vektor hasil konversi setiap 8 bit nilai biner fitur LmBP ke dalam nilai desimal
<pre> 3. for i ← 1 to length(v_1_8) 4. if neighbour_pixel(i) >= pixel_center 5. val ← 1 6. else 7. val ← 0 8. end 9. v_1_8vec[i] ← val 10. v_1_8bin ← bi2de(v_1_8vec) 11. end 12. for i ← 1 to length(v_2_16) 13. if neighbour_pixel(i) >= pixel_center 14. val ← 1 15. else 16. val ← 0 17. end 18. v_2_16vec ← val 19. v_2_16bin ← bi2de(v_2_16vec) 20. end 21. for i ← 1 to length(v_3_24) 22. if neighbour_pixel(i) >= pixel_center 23. val ← 1 24. else 25. val ← 0 26. end 27. v_3_24vec ← val 28. v_3_24bin ← bi2de(v_3_24vec) 29. end 30. for i ← 1 to length(v_4_8) 31. if neighbour_pixel(i) >= pixel_center 32. val ← 1 33. else 34. val ← 0 35. end 36. v_4_8vec ← val 37. v_4_8bin ← bi2de(v_4_8vec) 38. end 39. for i ← 1 to length(v_5_8) 40. if neighbour_pixel(i) >= pixel_center 41. val ← 1 42. else 43. val ← 0 44. end 45. v_5_8vec ← val 46. v_5_8bin ← bi2de(v_5_8vec) 47. end </pre>	

Pseudocode 3. 8 Ekstraksi Deskriptor Fitur LmBP	
Masukan	Citra grayscale hasil praproses, nilai indeks posisi baris dan kolom dari piksel yang diuji
Keluaran	Vektor biner dari fitur LmBP, vektor hasil konversi setiap 8 bit nilai biner fitur LmBP ke dalam nilai desimal
48.	$lmbpVector \leftarrow v_{1_8vec} + v_{2_16vec} + v_{3_24vec} + v_{4_8vec} + v_{5_8vec}$
49.	$lmbpBinerValue \leftarrow v_{1_8bin} + v_{2_16bin} + v_{3_24bin} + v_{4_8bin} + v_{5_8bin}$

Gambar 3.10 Pseudocode dari Tahap Ekstraksi Fitur LmBP

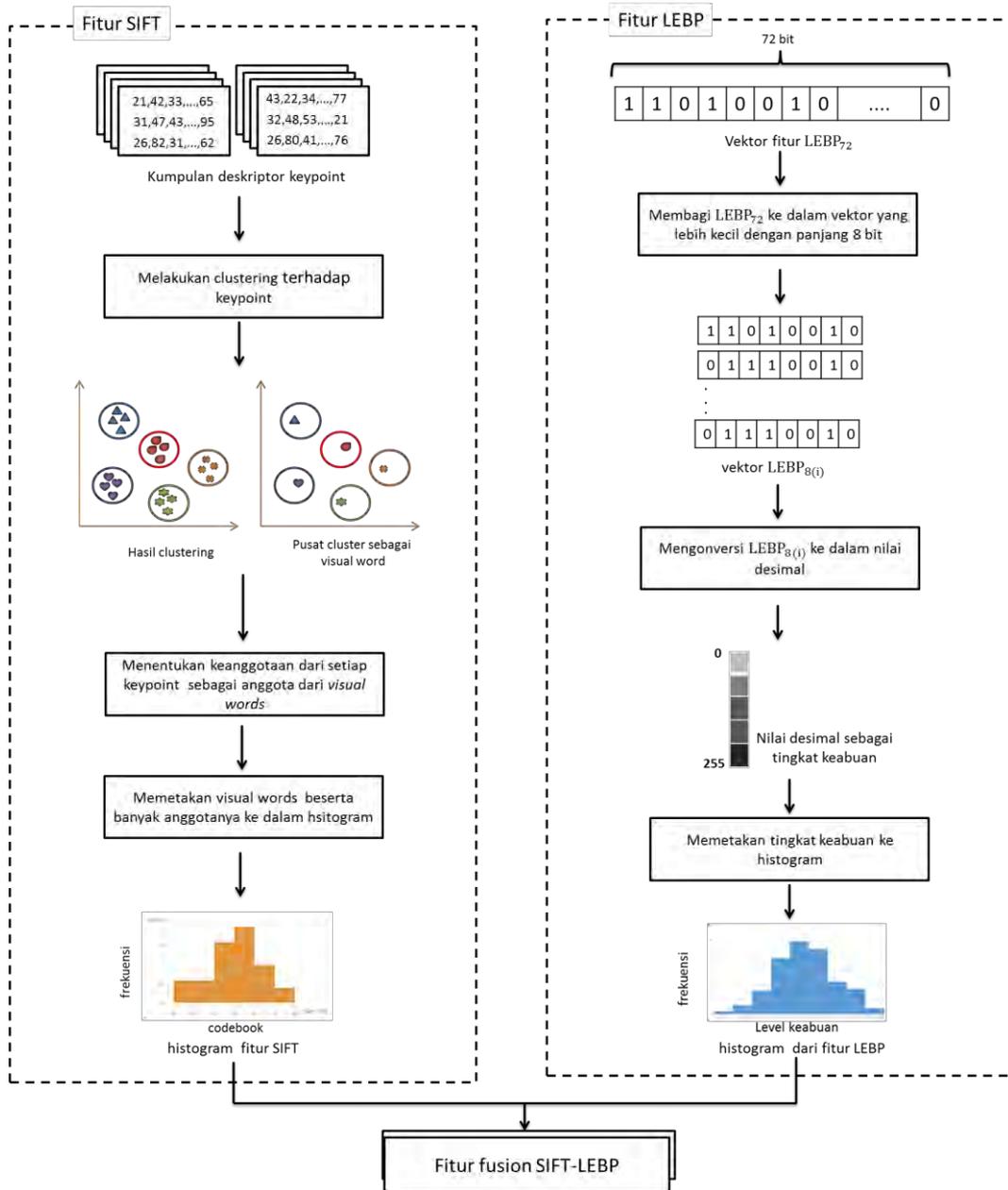
Pseudocode 3. 9 Ekstraksi Deskriptor Fitur LdbP	
Masukan	Citra grayscale hasil praproses, nilai indeks posisi baris dan kolom dari piksel yang diuji
Keluaran	Vektor biner dari fitur LdbP, vektor hasil konversi setiap 8 bit nilai biner ke dalam nilai desimal
1.	Inisialisasi matriks $ldbp8$ sebagai petunjuk lokasi piksel tetangga yang terlibat pada proses
2.	$pixel_center \leftarrow img(p1,p2)$
3.	for $i \leftarrow 1$ to $length(ldbp8)$
4.	if $neighbour_pixel(i) \geq pixel_center$
5.	$val = 1;$
6.	else
7.	$val = 0;$
8.	end
9.	$ldbp8vec \leftarrow val$
10.	$ldbp8bin \leftarrow bi2de(v_{4_8vec})$
11.	end

Gambar 3.11 Pseudocode dari Tahap Ekstraksi Fitur LdbP

Sedangkan proses untuk memperoleh deskriptor fitur LEBP dalam representasi histogram dijelaskan pada langkah-langkah sebagai berikut.

1. Membagi vektor $LEBP_{72} = [LmBP_{64}, LdbP_8]$ ke dalam vektor yang lebih kecil dengan panjang 8 bit . Jika vektor LEBP memiliki panjang total 72 bit maka akan diperoleh vektor yang lebih kecil $LEBP_{8(i)}$ dengan $i = 1, \dots, 9$. Setiap vektor berisi nilai biner yang mewakili pola tertentu. Pola tersebut merupakan pola-pola yang akan muncul ketika fitur LEBP diekstraksi.

2. Melakukan konversi dari setiap vektor pola biner $LEBP_{8(i)}$ ke bentuk desimal dalam rentang 0 hingga 255. Nilai tersebut menyatakan tingkat keabuan yang akan dipetakan ke histogram.
3. Memetakan nilai tingkat keabuan yang diperoleh ke dalam histogram citra dengan jumlah bin $n = 256$ yang akan ditampilkan pada bagian bawah plot histogram.



Gambar 3.12 Diagram Alur dari Fusion Fitur SIFT-LEBP

Setelah masing-masing histogram dari deskriptor fitur SIFT dan fitur LEBP diperoleh, maka berikutnya dilakukan penggabungan dari kedua histogram tersebut. Hasil penjumlahan bin dari kedua histogram tersebut menjadi jumlah fitur dari hasil *fusion* fitur SIFT-LEBP. Gambar 3.13 merupakan *pseudocode* dari proses fusion fitur SIFT-LEBP. Sedangkan variabel serta fungsi yang terlibat pada tahap fusion fitur SIFT-LEBP masing-masing ditunjukkan pada Tabel 3.8 dan Tabel 3.9.

Tabel 3.8 Daftar Variabel yang Digunakan pada Tahap Fusion Fitur SIFT-LEBP

No.	Nama Data	Tipe Data	Keterangan
1.	centers	double	Vektor yang menyimpan titik pusat dari hasil clustering deskriptor SIFT dengan K-Means.
2.	distance	double	Vektor yang menyimpan jarak dari setiap keypoints fitur SIFT dengan setiap titik pusat dari hasil clustering.
3.	ind	integer	Nilai yang menyimpan indeks dari titik pusat cluster yang memiliki jarak paling minimum dengan keypoint yang diuji.
4.	h	double	Vektor yang menyimpan histogram hasil pemetaan keypoints terhadap pusat cluster.
5.	counts_temp	double	Vektor yang menyimpan deskriptor fitur SIFT yang telah direpresentasikan ke dalam histogram.
6.	lenTrain	integer	Nilai yang menyimpan jumlah data pelatihan.
7.	lenTest	integer	Nilai yang menyimpan jumlah data uji.
8.	numClusters	integer	Nilai yang menyimpan jumlah cluster.
9.	countsm	double	Vektor yang menyimpan hasil penghitungan histogram dari hasil ekstraksi fitur LmBP.

Tabel 3.8 Daftar Variabel yang Digunakan pada Tahap Fusion Fitur SIFT-LEBP

No.	Nama Data	Type Data	Keterangan
10.	countsd	double	Vektor yang menyimpan hasil penghitungan histogram dari hasil ekstraksi fitur Ldbp.
11.	counts	double	Vektor yang menyimpan hasil penghitungan histogram dari hasil ekstraksi fitur LEBP.

Tabel 3.9 Daftar Fungsi yang Digunakan pada Tahap Fusion Fitur SIFT-LEBP

No.	Nama Fungsi	Keterangan
1.	vl_kmeans	Fungsi untuk melakukan proses clustering dengan metode K-Means.
2.	vl_alldist2	Fungsi untuk menghitung jarak dari seluruh data antara dua vektor.
3.	hist	Fungsi untuk menghasilkan histogram dengan jumlah bin yang ditentukan.
4.	imhist	Fungsi untuk menghasilkan histogram dari suatu citra.

Pseudocode 3. 10 Ekstraksi Deskriptor Fusion Fitur SIFT-LEBP	
Masukan	Citra pembuluh darah dalam format grayscale jari hasil praproses
Keluaran	Vektor representasi histogram dari fusion fitur SIFT-LEBP
<ol style="list-style-type: none"> 1. Proses pembentukan deskriptor fitur SIFT ke dalam representasi histogram 2. <code>centers ← vl_kmeans(data_concat, numClusters)</code> 3. <code>for j ← 1 to lenTest</code> 4. <code>distance ← vl_alldist2(centers, data_concat)</code> 5. <code>ind ← min(distance)</code> 6. <code>h ← hist(ind,numClusters)</code> 7. <code>counts_temp ← h</code> 8. <code>end</code> 9. Proses pembentukan deskriptor fitur LEBP ke dalam representasi histogram 10. <code>countsm ← imhist(lmbpMatrix)</code> 11. <code>countsd ← imhist(ldbpMatrix)</code> 12. <code>counts ← countsm + countsd</code> 13. <code>comb ← add vektor counts_temp</code> 14. <code>comb ← add vektor counts</code> 	

Gambar 3.13 Pseudocode dari Fusion Fitur SIFT-LEBP

3.2.6 Pengenalan Pembuluh Darah Jari dengan LVQ

Tahap terakhir yang dilakukan sistem yaitu melakukan proses pengenalan pembuluh darah jari dengan algoritma *machine learning* LVQ yang telah dijelaskan pada sub bab 2.4. Proses ini terbagi menjadi dua tahap yaitu proses pelatihan dan proses uji coba. Pada proses pelatihan model dari jaringan LVQ dibentuk. Jumlah neurons menentukan kompleksitas dari model LVQ yang dihasilkan. Semakin besar ukuran dari hidden layer maka semakin banyak pula karakteristik cluster yang dipelajari oleh model serta pemetaan data masukan ke target keluaran akan semakin kompleks. Gambar 3.14 merupakan *pseudocode* dari proses pembuluh darah jari dengan metode LVQ. Sedangkan variabel serta fungsi yang terlibat pada tahap pembuluh darah jari dengan metode LVQ masing-masing ditunjukkan pada Tabel 3.10 dan Tabel 3.11.

Tabel 3.10 Daftar Variabel yang Digunakan pada Tahap Pengenalan pembuluh Darah Jari dengan Metode LVQ

No.	Nama Data	Tipe Data	Keterangan
1.	net	object	Objek yang menyimpan model hasil pelatihan dari data masukan menggunakan algoritma LVQ.
2.	hiddenSize	integer	Nilai yang menyimpan ukuran hidden layer.
3.	lvqLR	double	Nilai yang menyimpan <i>learning rate</i> dari algoritma LVQ.
4.	lvqLF	string	Tipe dari <i>learning function</i> algoritma LVQ.
5.	epochs	integer	Nilai yang menyimpan indeks dari titik pusat cluster yang memiliki jarak paling minimum dengan keypoint yang diuji.
6.	data_train	double	Vektor yang menyimpan fitur data yang digunakan untuk proses pelatihan.
7.	class	double	Vektor yang menyimpan kelas sebenarnya dari masing-masing data pelatihan.
8.	outputs	double	Vektor yang menyimpan kelas hasil prediksi dari masing-masing data uji.

Tabel 3.10 Daftar Variabel yang Digunakan pada Tahap Pengenalan pembuluh Darah Jari dengan Metode LVQ

No.	Nama Data	Tipe Data	Keterangan
9.	data_test	double	Vektor yang menyimpan fitur data yang digunakan untuk proses pengujian.
10.	tpr	double	Nilai yang menyimpan hasil <i>true positive rate</i> .
11.	fpr	double	Nilai yang menyimpan hasil <i>false positive rate</i> .

Tabel 3.11 Daftar Fungsi yang Digunakan pada Tahap Pengenalan pembuluh Darah Jari dengan Metode LVQ

No.	Nama Fungsi	Keterangan
1.	lvqnet	Fungsi untuk melakukan proses klasifikasi menggunakan algoritma LVQ sehingga terbentuk model yang akan digunakan pada tahap pelatihan.
2.	train	Fungsi untuk melakukan proses pelatihan terhadap data latih.
3.	sim	Fungsi untuk melakukan simulasi terhadap data uji berupa hasil prediksi kelas dari data uji.
4.	roc	Fungsi untuk menghitung nilai tpr dan fpr hasil proses pengujian.

Pseudocode 3. 11 Pengenalan Pembuluh Darah Jari dengan Metode LVQ	
Masukan	Vektor histogram fitur SIFT-LEBP
Keluaran	Nilai tpr, fpr, dan hasil pengenalan data uji citra pembuluh darah jari.
<pre> 1. Proses pelatihan dari pengenalan pembuluh darah jari dengan LVQ 2. net = lvqnet(hiddenSize,lvqLR,lvqLF, epochs); 3. net = train(net,data_train,class); 4. Proses pengujian dari pengenalan pembuluh darah jari dengan LVQ 5. outputs = sim(net,data_test); 6. [tpr,fpr] = roc(classOut,outputs); </pre>	

Gambar 3.14 Pseudocode dari Pengenalan Pembuluh Darah Jari dengan Metode LVQ

BAB IV

HASIL UJI COBA DAN PEMBAHASAN

Bab ini menjelaskan uji coba yang dilakukan pada sistem yang telah dibangun beserta hasil dan analisis uji coba. Pembahasan pengujian meliputi lingkungan uji coba, skenario uji coba, serta analisis setiap skenario pengujian.

4.1 Lingkungan Uji Coba

Lingkungan uji coba menjelaskan spesifikasi dari perangkat keras dan perangkat lunak yang digunakan untuk menguji sistem yang telah dibangun. Sistem yang diuji yaitu sistem pengenalan pembuluh darah jari. Penjelasan dari perangkat keras dan perangkat lunak yang digunakan yaitu sebagai berikut.

1. Perangkat keras

- a. Prosesor: Intel(R) Core(TM) i3-4030U CPU @ 1.90GHz 1.90 GHz.
- b. Memori(RAM): 2.00 GB
- c. Tipe Sistem: 64-bit sistem operasi.

2. Perangkat Lunak

- a. Sistem Operasi: Windows 7 Home Basic.
- b. Perangkat Pengembang: MATLAB 7.12.0 (R2011a).

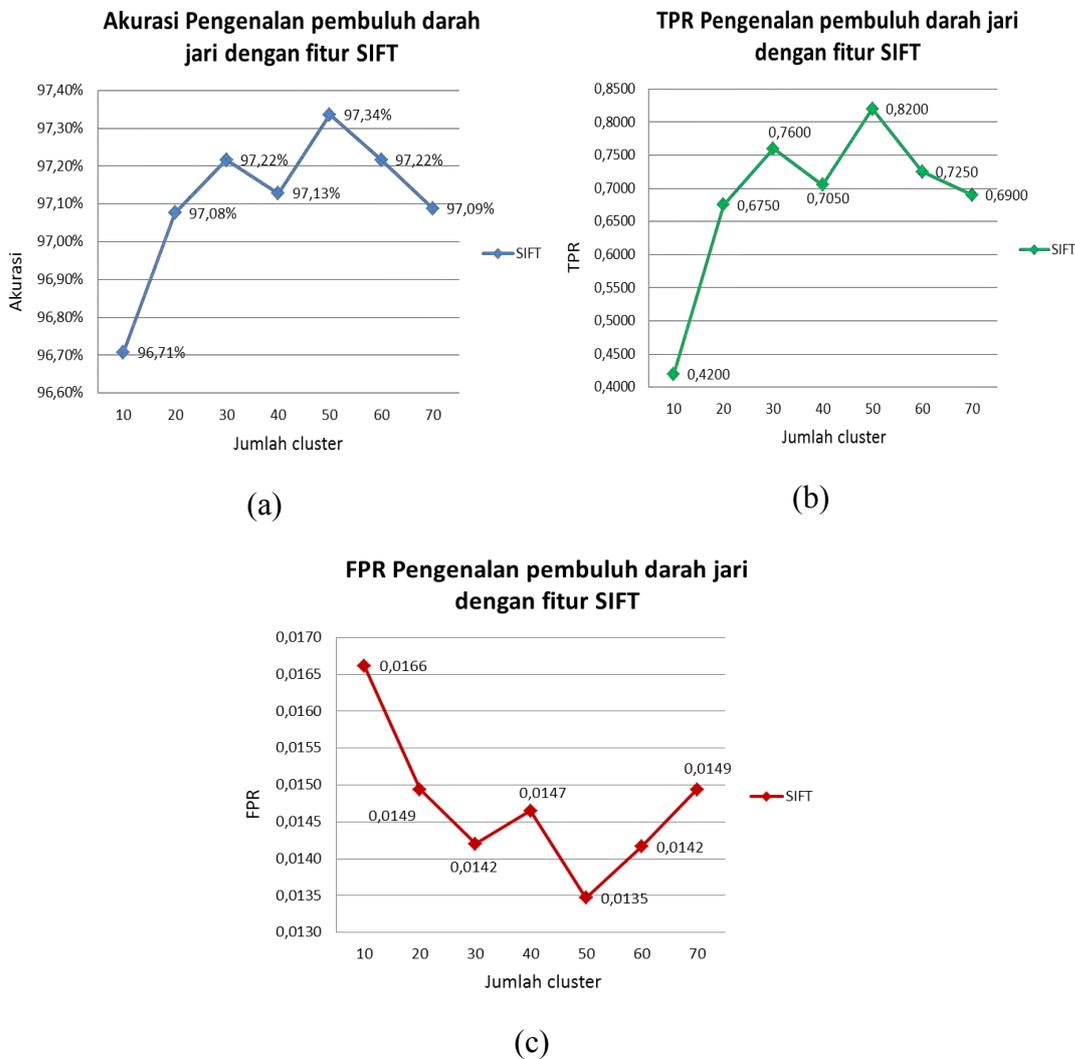
4.2 Uji Coba Pengenalan Pembuluh Darah Jari dengan Fitur SIFT

Uji coba yang dilakukan pada tahap ini dilakukan untuk menguji kinerja dari fitur SIFT terhadap kesesuaian proses pengenalan pembuluh darah jari. Pada uji coba juga dilakukan perbandingan kinerja dengan mengubah nilai parameter berpengaruh pada pembentukan histogram deskriptor fitur SIFT. Data uji coba yang digunakan adalah vektor fitur dari representasi histogram deskriptor fitur SIFT dimana langkah untuk memperoleh vektor fitur tersebut dijelaskan pada bab 2.3.1. Besar dimensi dari vektor fitur masukan yaitu $n \times c$ dengan n menyatakan jumlah citra yang diuji dan c menyatakan jumlah pusat kluster yang ditentukan untuk membentuk histogram deskriptor fitur SIFT. Nilai c ini yang akan menjadi nilai parameter berpengaruh yang akan diuji. Metode klasifikasi LVQ digunakan

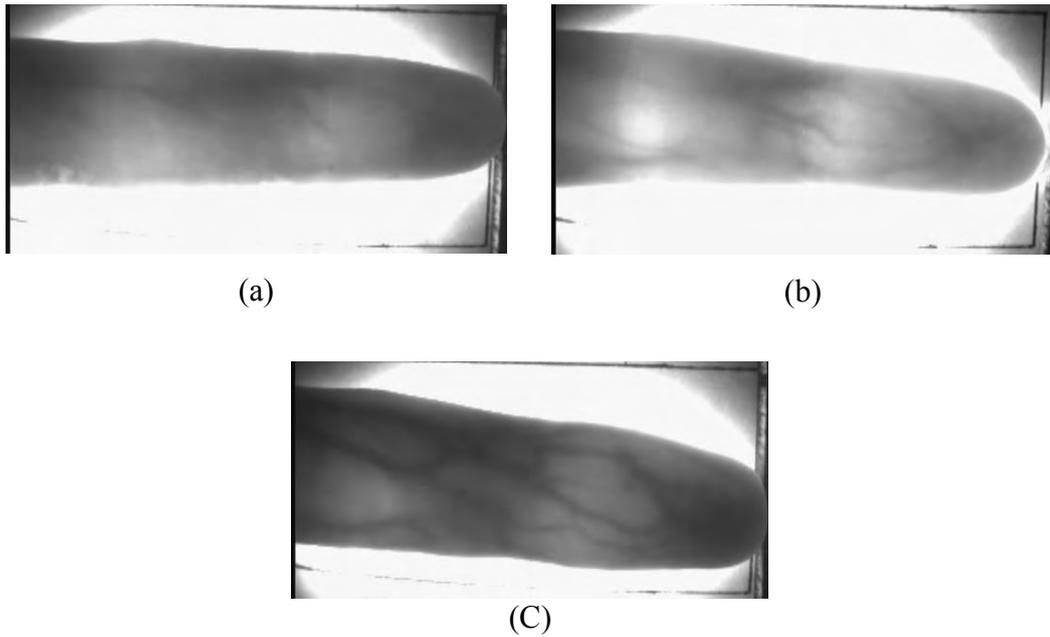
Tabel 4.1 Tetapan parameter uji coba pengenalan pembuluh darah jari dengan fitur SIFT

Parameter	Tetapan parameter
c	10, 20, 30, 40, 50, 60, 70
jumlah <i>hidden layer</i>	500
learning rate	0.01
epochs	300

untuk mendapatkan hasil dari pengenalan data masukan. Tetapan rentang parameter yang akan diuji serta parameter pada metode LVQ ditunjukkan pada Tabel 4.1.



Gambar 4.1 Hasil pengenalan pembuluh darah jari dengan fitur SIFT yang meliputi (a) Nilai akurasi (b) Nilai TPR (c) Nilai FPR



Gambar 4.2 Contoh citra yang berhasil dikenali oleh sistem pengenalan pembuluh darah jari dengan fitur SIFT (a) Citra yang mengalami efek translasi (b) Citra yang mengalami efek rotasi (c) Citra yang mengalami efek rotasi dan translasi

Pada uji coba pengenalan pembuluh darah jari dengan fitur SIFT, data training diolah menggunakan metode LVQ sehingga diperoleh model untuk melakukan klasifikasi. Data testing kemudian diolah untuk mendapatkan hasil prediksi. Dari hasil prediksi tersebut, akan diperoleh nilai akurasi, FPR, dan TPR. Proses pembagian data testing dan data training untuk uji coba dilakukan dengan membagi data menggunakan *k-fold cross validation* dengan nilai $k = 3$.

Gambar 4.1 menunjukkan hasil kinerja dari sistem pengenalan pembuluh darah jari menggunakan fitur SIFT. Dari hasil tersebut terlihat bahwa nilai akurasi tertinggi diperoleh ketika jumlah cluster ditetapkan sebanyak 50 cluster dengan nilai akurasi yang diperoleh sebesar 97,34%. Nilai TPR yang diperoleh pun juga merupakan nilai TPR tertinggi yaitu sebesar 0,8200. Sedangkan untuk nilai FPR, penetapan cluster sejumlah 50 memberikan nilai FPR terendah dengan nilai 0,0135. Pada Gambar 4.2 juga ditunjukkan contoh citra yang berhasil dikenali oleh sistem pengenalan pembuluh darah jari menggunakan fitur SIFT meskipun citra tersebut mengalami degradasi geometris yang disebabkan efek translasi, rotasi, atau keduanya.

4.3 Uji Coba Pengenalan Pembuluh Darah Jari dengan Fitur LEBP

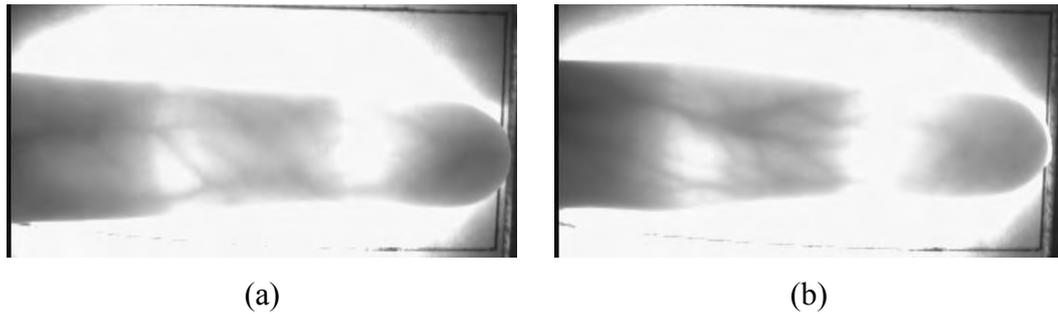
Uji coba yang dilakukan pada tahap ini dilakukan untuk menguji kinerja dari fitur LEBP terhadap kesesuaian proses pengenalan pembuluh darah jari. Seperti yang telah dijelaskan pada sub bab 2.3.3, fitur LEBP merupakan gabungan dari fitur LmBP dan LdBp. Sehingga pada uji coba ini dilakukan perbandingan kinerja dari pengenalan pembuluh darah jari menggunakan fitur LmBP, LdBp, dan LEBP. Data uji coba yang digunakan adalah vektor fitur dari representasi histogram deskriptor fitur LEBP dimana langkah untuk memperoleh vektor fitur tersebut dijelaskan pada bab 3.1.4. Besar dimensi dari vektor fitur masukan yaitu $n \times 255$ dengan n menyatakan jumlah citra yang diuji.

Pada uji coba pengenalan pembuluh darah jari dengan fitur SIFT, data training diolah menggunakan metode LVQ sehingga diperoleh model untuk melakukan klasifikasi. Proses training dilakukan dengan tetapan parameter untuk metode LVQ yaitu jumlah hidden layer 500, learning rate 0,01, dan jumlah epochs 300. Data testing kemudian diolah untuk mendapatkan hasil prediksi. Dari hasil prediksi tersebut, akan diperoleh nilai akurasi, FPR, dan TPR. Proses pembagian data testing dan data training untuk uji coba dilakukan dengan membagi data menggunakan *k-fold cross validation* dengan nilai $k = 3$.

Berdasarkan Tabel 4.2, nilai akurasi tertinggi diperoleh dari penggunaan fitur LEBP dengan nilai akurasi yang diperoleh sebesar 97,32%. Nilai TPR yang diperoleh pun juga merupakan nilai TPR tertinggi yaitu sebesar 0,8300. Sedangkan untuk nilai FPR, penggunaan fitur LEBP memberikan nilai FPR terendah dengan nilai 0,0137. Penggunaan fitur LmBP memberikan hasil tertinggi

Tabel 4.2 Hasil uji coba pengenalan pembuluh darah jari dengan fitur LmBP, LdBp, dan LEBP

Fitur	FPR	TPR	Accuracy (%)
LmBP	0,0141	0,7800	97,24
LdBp	0,0184	0,2350	96,38
LEBP	0,0137	0,8300	97,32



Gambar 4.3 Contoh citra yang berhasil dikenali oleh sistem pengenalan pembuluh darah jari dengan fitur LEBP (a),(b) citra dengan perbedaan intensitas pencahayaan

berikutnya dengan besar perbedaan yang tidak terlalu signifikan dengan penggunaan fitur LEBP. Besar nilai akurasi yang diberikan yaitu 97,24%, nilai TPR yaitu 0,7800, dan nilai FPR sebesar 0,0141. Sedangkan hasil terendah diperoleh dari penggunaan fitur LdBP. Besar nilai akurasi yang diberikan yaitu 96,38%, nilai TPR yaitu 0,2350, dan nilai FPR sebesar 0,0184. Sedangkan pada Gambar 4.3 ditunjukkan contoh citra yang berhasil dikenali oleh sistem pengenalan pembuluh darah jari menggunakan fitur LEBP meskipun citra tersebut memiliki perbedaan intensitas pencahayaan.

4.4 Uji Coba Pengenalan Pembuluh Darah Jari dengan Fitur SIFT-LEBP

Uji coba yang dilakukan pada tahap ini dilakukan untuk menguji kinerja dari fusion fitur SIFT-LEBP terhadap kesesuaian proses pengenalan pembuluh darah jari. Pada uji coba juga dilakukan perbandingan kinerja dengan mengubah nilai parameter berpengaruh pada metode klasifikasi LVQ. Daftar parameter yang diuji beserta rentang nilainya ditunjukkan pada Tabel 4.3. Data uji coba yang digunakan adalah vektor fitur SIFT-LEBP yang direpresentasikan dalam vektor histogram.

1) Uji Coba pengenalan Pembuluh Darah Jari Berdasarkan Jumlah hidden layer

Pada uji coba pengenalan pembuluh darah jari dengan fitur SIFT-LEBP, data training diolah menggunakan metode LVQ sehingga diperoleh model untuk melakukan klasifikasi. Proses training dilakukan dengan mengubah tetapan parameter dari jumlah hidden layer pada metode LVQ dengan nilai 300, 400, 500 dan 600. Sedangkan parameter lain seperti learning rate dan jumlah epochs,

Tabel 4.3 Tetapan parameter uji coba pengenalan pembuluh darah jari dengan fitur SIFT-LEBP

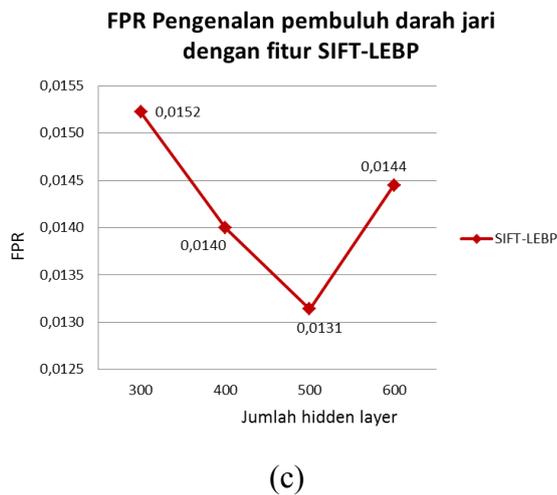
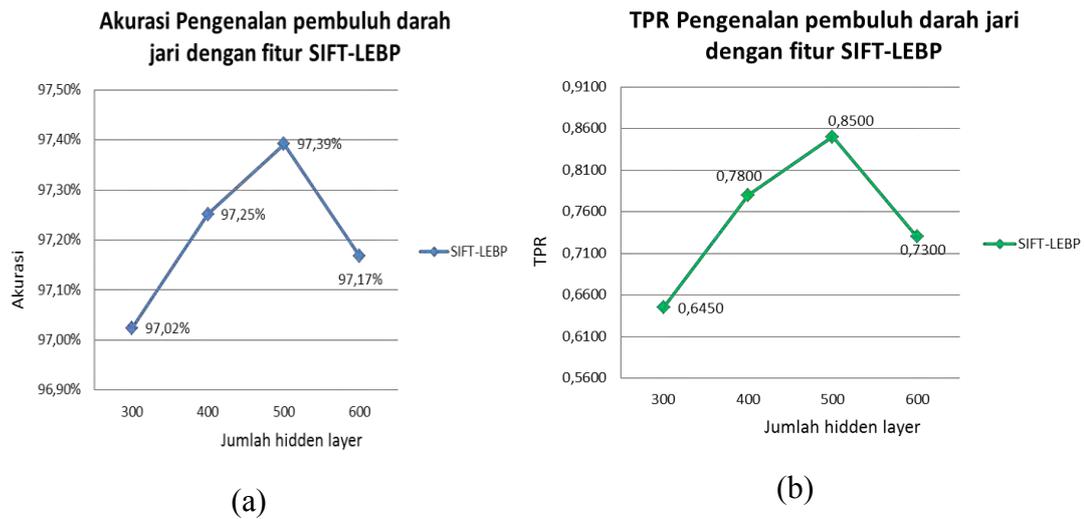
Parameter	Tetapan parameter
jumlah <i>hidden layer</i>	300, 400, 500, 600
Jumlah epochs	300, 325, 350, 375, 400
learning rate	0.01-0.05

masing-masing ditetapkan dengan nilai 0,01 dan 300. Histogram fitur SIFT yang digunakan adalah histogram yang dibentuk dengan jumlah cluster yang ditetapkan 50. Data testing kemudian diolah untuk mendapatkan hasil prediksi. Dari hasil prediksi tersebut, akan diperoleh nilai akurasi, FPR, dan TPR. Proses pembagian data testing dan data training untuk uji coba dilakukan dengan membagi data menggunakan *k-fold cross validation* dengan nilai $k = 3$.

Berdasarkan Gambar 4.4, nilai akurasi tertinggi diperoleh ketika jumlah *hidden layer* ditetapkan sebanyak 500 *nodes* dengan nilai akurasi yang diperoleh sebesar 97,39%. Nilai TPR yang diperoleh pun juga merupakan nilai TPR tertinggi yaitu sebesar 0,8500 Sedangkan untuk nilai FPR, penetapan jumlah *hidden layer* sejumlah 500 memberikan nilai FPR terendah dengan nilai 0,0131.

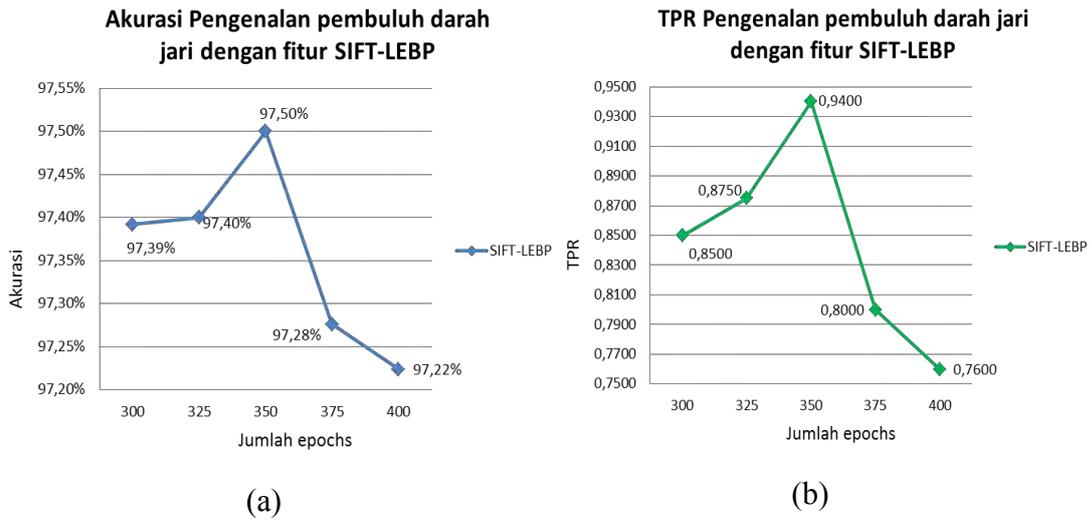
2) Uji Coba pengenalan Pembuluh Darah Jari Berdasarkan Jumlah Epochs

Pada uji coba pengenalan pembuluh darah jari dengan fitur SIFT-LEBP, data training diolah menggunakan metode LVQ sehingga diperoleh model untuk melakukan klasifikasi. Proses training dilakukan dengan mengubah tetapan parameter dari jumlah epochs pada metode LVQ dengan nilai 300, 325, 350, 375 dan 400. Sedangkan parameter lain seperti jumlah *hidden layer* dan nilai learning rate, masing-masing ditetapkan dengan nilai 500 dan 0.01. Penetapan jumlah *hidden layer* diambil dari nilai yang memberikan hasil kinerja dengan nilai akurasi dan TPR tertinggi serta nilai FPR terendah dari uji coba sebelumnya. Histogram fitur SIFT yang digunakan adalah histogram yang dibentuk dengan jumlah cluster yang ditetapkan sebanyak 50. Data testing kemudian diolah untuk mendapatkan hasil prediksi. Dari hasil prediksi tersebut, akan diperoleh nilai akurasi, FPR, dan TPR. Proses pembagian data testing dan data training untuk uji coba dilakukan dengan membagi data menggunakan *k-fold cross validation* dengan nilai $k = 3$.



Gambar 4.4 Hasil pengenalan pembuluh darah jari dengan fitur SIFT-LEBP dengan perubahan nilai parameter jumlah hidden layer yang meliputi (a) Nilai akurasi (b) Nilai TPR (c) Nilai FPR

Berdasarkan Gambar 4.5 nilai akurasi tertinggi diperoleh ketika jumlah epochs ditetapkan sebanyak 350 dengan nilai akurasi yang diperoleh sebesar 97,50%. Nilai TPR yang diperoleh pun juga merupakan nilai TPR tertinggi yaitu sebesar 0,9400. Sedangkan untuk nilai FPR, penetapan jumlah epochs sejumlah 500 memberikan nilai FPR terendah dengan nilai 0,0128.

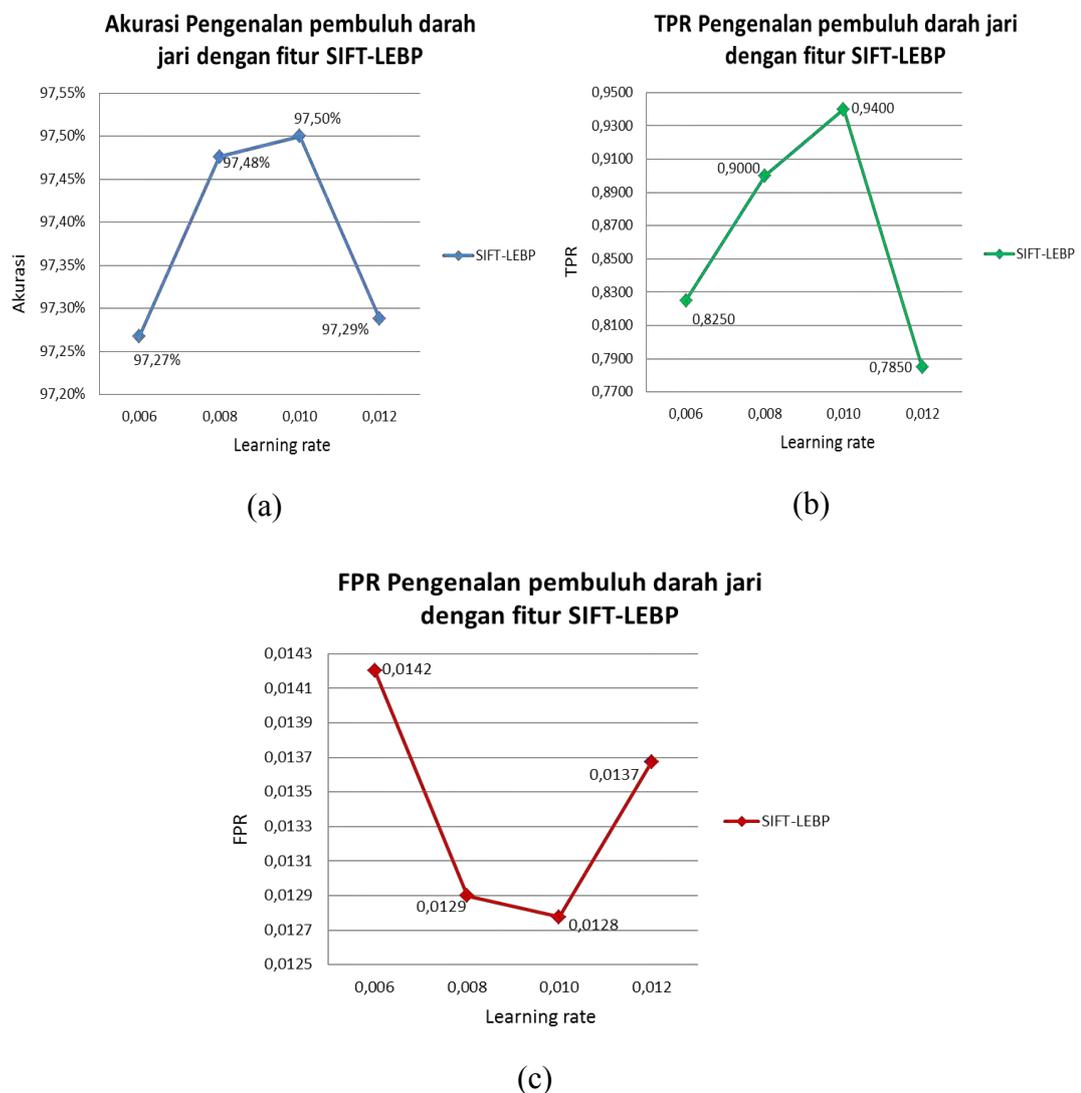


Gambar 4.5 Hasil pengenalan pembuluh darah jari dengan fitur SIFT-LEBP dengan perubahan nilai parameter jumlah epochs yang meliputi (a) Nilai akurasi (b) Nilai TPR (c) Nilai FPR

3) Uji Coba pengenalan Pembuluh Darah Jari Berdasarkan Nilai Learning Rate

Pada uji coba pengenalan pembuluh darah jari dengan fitur SIFT-LEBP, data training diolah menggunakan metode LVQ sehingga diperoleh model untuk melakukan klasifikasi. Proses training dilakukan dengan mengubah tetapan parameter dari nilai learning rate pada metode LVQ dengan nilai 0.006, 0.008, 0.01, dan 0.012. Sedangkan parameter lain seperti jumlah hidden layer dan jumlah epochs, masing-masing ditetapkan dengan nilai 500 dan 350. Penetapan nilai parameter tersebut diambil dari uji coba sebelumnya dimana nilai parameter yang

diambil adalah nilai parameter yang memberikan hasil kinerja dengan nilai akurasi dan TPR tertinggi serta nilai FPR terendah. Histogram fitur SIFT yang digunakan adalah histogram yang dibentuk dengan jumlah cluster yang ditetapkan 50. Data testing kemudian diolah untuk mendapatkan hasil prediksi. Dari hasil prediksi tersebut, akan diperoleh nilai akurasi, FPR, dan TPR. Proses pembagian data testing dan data training untuk uji coba dilakukan dengan membagi data menggunakan k-fold cross validation dengan nilai $k = 3$.



Gambar 4.6 Hasil pengenalan pembuluh darah jari dengan fitur SIFT-LEBP dengan perubahan nilai parameter learning rate yang meliputi (a) Nilai akurasi (b) Nilai TPR (c) Nilai FPR

Berdasarkan Gambar 4.6 nilai akurasi tertinggi diperoleh ketika nilai learning rate ditetapkan sebesar 0,01 dengan nilai akurasi yang diperoleh sebesar 97,50%. Nilai TPR yang diperoleh pun juga merupakan nilai TPR tertinggi yaitu sebesar 0,9400. Sedangkan untuk nilai FPR, penetapan nilai learning rate sebesar 0,01 memberikan nilai FPR terendah dengan nilai 0,0128.

4.5 Analisis Uji Coba Sistem Pengenalan Pembuluh Darah Jari

Uji coba yang dilakukan terdiri dari tiga jenis pengujian, yaitu uji coba kinerja fitur SIFT, uji coba kinerja fitur LEBP, dan uji coba kinerja fitur SIFT-LEBP. Pada sub bab ini akan dijelaskan analisis terhadap hasil dari ketiga jenis uji coba tersebut.

Pada uji coba kinerja fitur SIFT, nilai akurasi yang diperoleh cukup tinggi yaitu dengan nilai tertinggi mencapai 97,34%. Begitu pula dengan nilai TPR yang diperoleh juga memiliki nilai yang cukup tinggi yaitu dengan nilai tertinggi mencapai 0,8200. Sedangkan untuk nilai FPR, hasil yang peroleh juga cukup baik yaitu nilai terendah mencapai 0,0135. Nilai akurasi dan TPR tertinggi serta nilai FPR terendah diperoleh ketika jumlah *cluster* yang ditetapkan yaitu 50 *cluster*. Jika melihat nilai akurasi, TPR, dan FPR yang diperoleh maka dapat disimpulkan bahwa sistem memberikan hasil kebenaran yang tinggi. Selain itu, kebenaran dari ekstraksi fitur SIFT untuk pengenalan citra pembuluh darah jari dapat juga dilihat hasil citra yang berhasil dikenali pada Gambar 4.2 Dari gambar tersebut menunjukkan bahwa fitur SIFT terbukti mampu mengenali citra yang mengalami degradasi geometri meliputi efek translasi, rotasi, maupun disebabkan oleh keduanya. Dengan hasil yang telah diperoleh tersebut maka fitur SIFT yang diekstraksi dapat digunakan sebagai salah satu fitur *fusion* pada sistem pengenalan pembuluh darah jari. Uji coba juga dilakukan untuk melihat besar pengaruh dari perubahan jumlah *cluster* dalam pembentukan histogram vektor fitur SIFT terhadap nilai akurasi, TPR, dan FPR. Dari hasil uji coba yang diperoleh, perubahan nilai parameter jumlah *cluster* tidak terlalu banyak mempengaruhi hasil akurasi akan tetapi memiliki pengaruh cukup besar terhadap hasil TPR.

Pada uji coba kinerja fitur LEBP, diperoleh hasil akurasi yang cukup tinggi dari penggunaan fitur LmBP, LdBP, maupun LEBP. Fitur LEBP yang

merupakan gabungan dari fitur LmBP dan LdBP memberikan hasil akurasi sebesar 97,32%, TPR sebesar 0,8300, dan FPR sebesar 0,0137. Hasil tersebut dapat dikatakan lebih baik jika dibandingkan dengan hasil akurasi, TPR, dan FPR dengan menggunakan fitur LdBP. Penggunaan fitur LdBP menghasilkan nilai akurasi yang tidak mencapai 97% yaitu hanya 96,38%. Nilai TPR yang dihasilkan dari penggunaan fitur LdBP pun juga rendah yaitu hanya sebesar 0,2350. Akan tetapi, fitur ini memberikan nilai FPR yang cukup rendah yaitu sebesar 0,0184. Sedangkan nilai akurasi, TPR, dan FPR dari penggunaan fitur LEBP tidak jauh berbeda dengan hasil akurasi, TPR, maupun FPR dengan menggunakan fitur LmBP. Nilai akurasi dengan menggunakan fitur LmBP mencapai 97,24% dengan nilai TPR sebesar 0,7800 dan nilai FPR sebesar 0,0141. Jika melihat nilai akurasi, TPR, dan FPR yang diperoleh dari penggunaan fitur LEBP maka dapat disimpulkan bahwa sistem memberikan hasil kebenaran yang tinggi. Hasil akurasi dari fitur LEBP yang merupakan gabungan dari fitur LmBP dan fitur LdBP tidak menunjukkan penurunan nilai akurasi, TPR, maupun FPR. Hal tersebut menunjukkan proses penggabungan fitur LmBP dan LdBP untuk membentuk fitur LEBP dapat dikatakan berhasil. Selain itu, fitur LEBP yang diharapkan mampu mengatasi permasalahan degradasi citra akibat perbedaan intensitas pencahayaan terbukti dapat mengatasi permasalahan tersebut. Contoh citra dengan perbedaan intensitas pencahayaan yang mampu dikenali dengan menggunakan fitur LEBP ditunjukkan pada Gambar 4.3. Dengan hasil tersebut maka fitur LEBP yang diekstraksi dapat digunakan sebagai salah satu fitur fusion pada sistem pengenalan pembuluh darah jari.

Untuk menguji kinerja dari fitur SIFT-LEBP yang merupakan hasil fusion antara fitur SIFT dan LEBP, maka dilakukan perubahan nilai parameter pada metode LVQ hingga ditemukan hasil akurasi, TPR, dan FPR yang paling optimal. Parameter-parameter yang diubah yaitu jumlah hidden layer, jumlah epochs, dan nilai learning rate. Ketika jumlah hidden layer ditetapkan sejumlah 300, nilai akurasi telah mencapai 97%. Akan tetapi nilai TPR yang diperoleh hanya berkisar 0,6450 dengan nilai FPR 0,0152. Nilai TPR bertambah cukup signifikan ketika jumlah hidden layer ditetapkan menjadi 400 dan mencapai nilai tertinggi ketika jumlah hidden layer ditambah menjadi 500. Akurasi yang diperoleh dengan

jumlah 500 hidden layer yaitu sebesar 97,39% dengan nilai TPR dan FPR masing-masing 0,8500 dan 0,0131. Nilai akurasi, TPR, dan FPR mengalami penurunan ketika jumlah hidden layer ditambah hingga menjadi 600. Hal tersebut menunjukkan bahwa penetapan nilai hidden layer sejumlah 500 merupakan jumlah hidden layer yang optimal.

Sedangkan untuk nilai epochs, penetapan jumlah awal dilakukan secara acak dengan inisialisasi awal sejumlah 300. Ketika jumlah epochs ditambah dengan peningkatan konstan sebanyak 25 epochs, nilai akurasi dan TPR mengalami peningkatan diikuti penurunan nilai FPR. Nilai akurasi dan TPR tertinggi serta FPR terendah diperoleh ketika jumlah epochs yang ditetapkan sejumlah 350 dimana nilai akurasi dan TPR mencapai 97,49% dan 0,9300 sedangkan nilai FPR diperoleh sebesar 0,0128. Penetapan nilai epochs dengan jumlah 350 menunjukkan hasil optimal karena semakin bertambah nilai epochs hingga melebihi 350, maka nilai akurasi dan TPR mengalami penurunan sedangkan nilai FPR semakin bertambah.

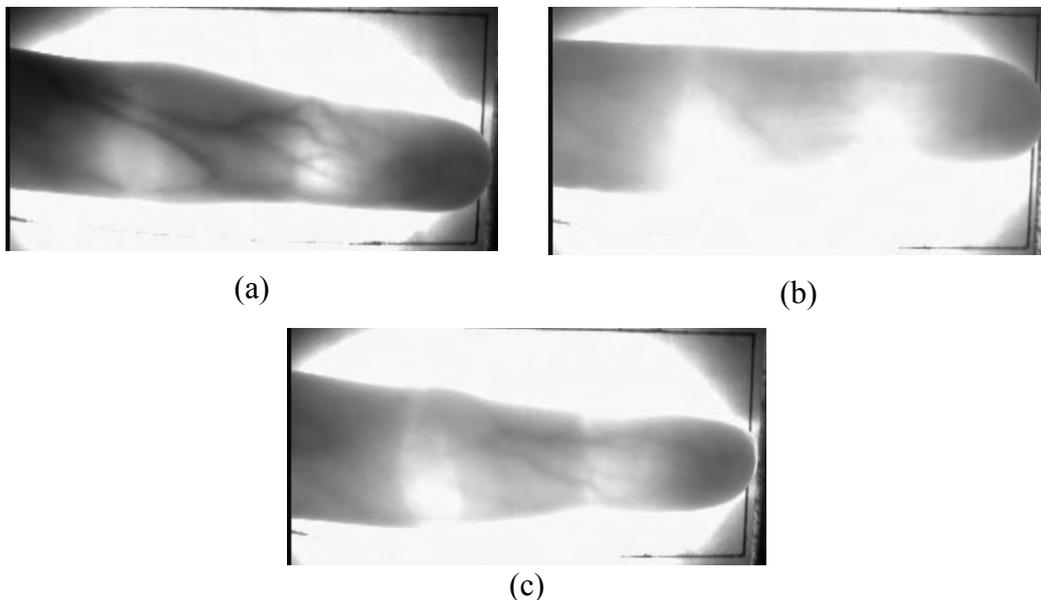
Untuk uji coba dengan pengubahan nilai learning rate, maka penetapan learning rate awal adalah nilai learning rate dari uji coba sebelumnya yaitu sebesar 0,01. Ketika nilai learning rate ini ditingkatkan dengan peningkatan sebesar 0,002 maka nilai akurasi dan nilai TPR mengalami penurunan cukup besar dan nilai FPR meningkat. Sehingga dilakukan penurunan nilai learning rate dengan penurunan konstan sebesar 0,002. Hasil akurasi, TPR, maupun FPR yang diperoleh tidak melebihi hasil akurasi, TPR, maupun FPR ketika nilai learning rate ditetapkan sebesar 0,01. Akan tetapi justru mengalami penurunan. Dari hasil uji coba pengubahan nilai parameter menunjukkan bahwa jumlah hidden layer, jumlah epochs, dan nilai learning rate memberikan pengaruh terhadap hasil kinerja dari sistem.

Pada sistem pengenalan pembuluh darah jari, hasil akurasi yang diperoleh dari proses fusi antara dua fitur yaitu fitur SIFT dan fitur LEBP memberikan hasil sebesar 97,50%. Nilai akurasi tersebut telah melebihi nilai akurasi yang diperoleh ketika menggunakan fitur tunggal. Sehingga dapat dikatakan bahwa teknik fusi untuk memperoleh fitur SIFT-LEBP telah benar. Selain itu, suatu sistem pengenalan pembuluh darah jari dikatakan memiliki kinerja yang baik ketika

Tabel 4.4 Hasil uji coba pengenalan pembuluh darah jari dengan fitur SIFT, LEBP, dan SIFT-LEBP

Fitur	FPR	TPR	Accuracy (%)
SIFT	0,0135	0,8200	97,34
LEBP	0,0137	0,8300	97,32
SIFT-LEBP	0,0128	0,9400	97,50

nilai dari *true positive* tinggi dan nilai *false negative* rendah. Jika melihat hasil kinerja dari fitur SIFT-LEBP yang merupakan hasil fusion dari fitur SIFT dan fitur LEBP pada Tabel 4.4, maka dapat terlihat bahwa nilai *true positive* dan *false negative* yang diperoleh relatif lebih baik jika dibandingkan dengan hanya menggunakan fitur tunggal SIFT atau LEBP. Hal tersebut dapat terlihat dari perbandingan dari nilai TPR dan FPR dari fitur SIFT-LEBP dengan fitur tunggal pada kondisi yang sama. Ketika parameter berpengaruh ditetapkan pada kondisi yang sama yaitu dengan jumlah hidden layer 500, jumlah epochs 300, dan nilai learning rate 0,01 maka nilai TPR sebesar 0,8500, dan nilai FPR sebesar 0,0131.

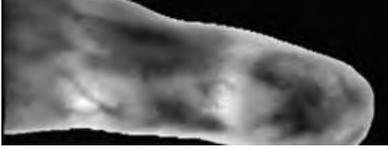
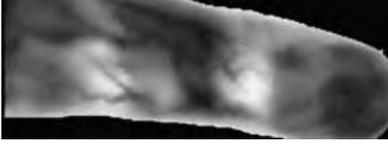


Gambar 4.7 Contoh citra yang berhasil dikenali oleh sistem pengenalan pembuluh darah jari dengan fitur SIFT-LEBP (a),(b),(c) Citra dengan degradasi transformasi geometris dan perbedaan intensitas pencahayaan

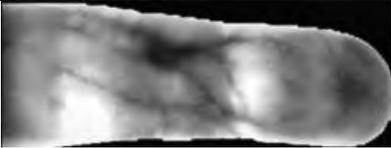
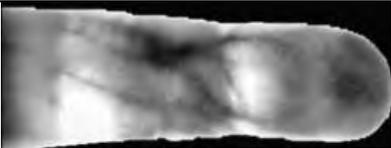
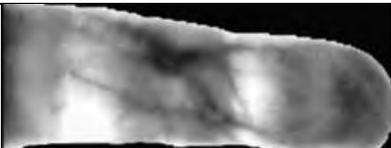
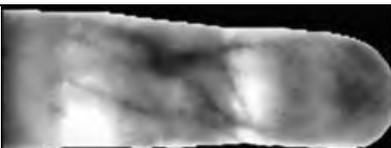
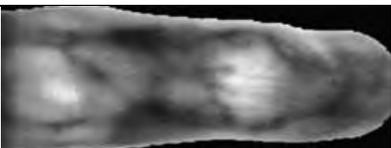
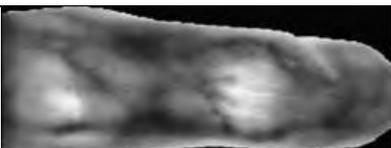
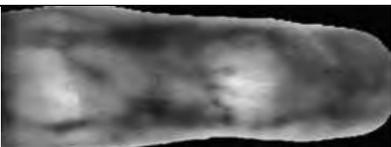
Sedangkan pada kondisi optimal yaitu pada kondisi dengan jumlah hidden layer 500, jumlah epochs 350, dan nilai learning rate 0,01 maka fitur SIFT-LEBP memberikan nilai TPR sebesar 0,9400, dan nilai FPR sebesar 0,0128. Hasil tersebut merupakan nilai TPR tertinggi jika dibandingkan dengan hasil dari penggunaan fitur tunggal. Begitu pula dengan nilai FPR merupakan nilai terendah jika dibandingkan dengan penggunaan fitur tunggal. Sedangkan untuk nilai akurasi, penggunaan fitur SIFT-LEBP juga memberikan nilai tertinggi. Hasil TPR yang tinggi disebabkan karena fitur SIFT-LEBP mampu mengenali citra mampu mengenali citra yang mengalami degradasi baik diakibatkan oleh efek rotasi, translasi, maupun perbedaan intensitas pencahayaan. Sedangkan fitur tunggal hanya mampu mengatasi beberapa degradasi citra saja. Sebagai contoh, citra pada Gambar 4.2 mampu dikenali dengan menggunakan fitur SIFT dan fitur SIFT-LEBP sedangkan fitur LEBP tidak mampu mengenali citra tersebut. Sedangkan citra pada Gambar 4.3 mampu dikenali dengan menggunakan fitur LEBP dan fitur SIFT-LEBP tetapi fitur SIFT tidak mampu mengenali citra tersebut. Fitur SIFT-LEBP juga mampu mengenali citra yang tidak mampu dikenali oleh fitur SIFT maupun fitur LEBP. Contoh citra tersebut ditunjukkan pada Gambar 4.7. Dengan demikian, dapat disimpulkan bahwa fitur SIFT-LEBP dapat dijadikan sebagai alternatif dari fitur untuk proses pengenalan pembuluh darah jari.

LAMPIRAN A

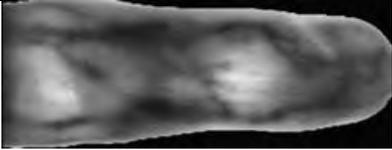
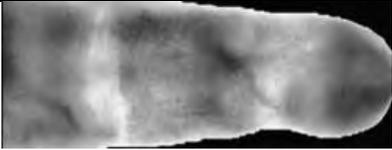
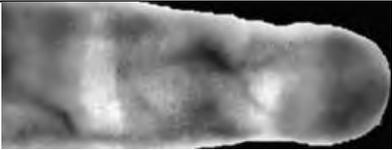
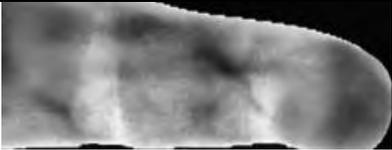
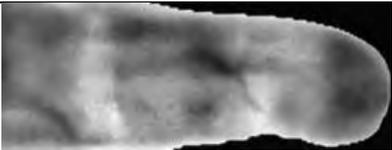
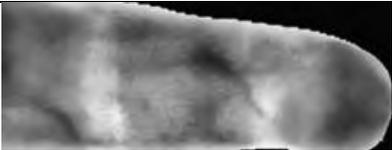
Tabel A. 1 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
1.	1_1_fl_2.bmp		
2.	1_2_fl_2.bmp		
3.	1_3_fl_2.bmp		
4.	1_4_fl_2.bmp		
5.	1_5_fl_2.bmp		
6.	1_6_fl_2.bmp		
7.	2_1_fl_2.bmp		
8.	2_2_fl_2.bmp		

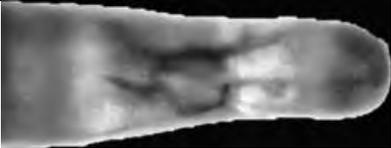
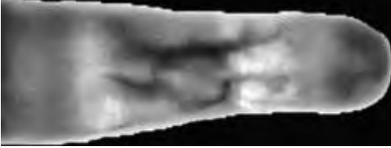
Tabel A. 2 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
9.	2_3_fl_2.bmp		
10.	2_4_fl_2.bmp		
11.	2_5_fl_2.bmp		
12.	2_6_fl_2.bmp		
13.	3_1_fl_2.bmp		
14.	3_2_fl_2.bmp		
15.	3_3_fl_2.bmp		
16.	3_4_fl_2.bmp		

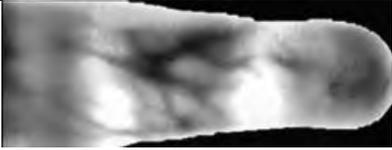
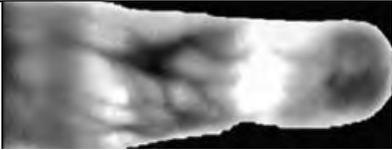
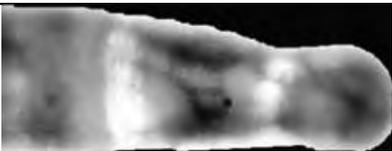
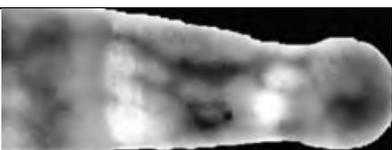
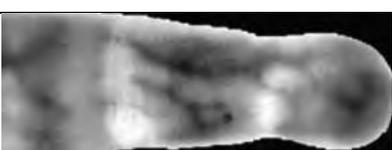
Tabel A. 3 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
17.	3_5_fl_2.bmp		
18.	3_6_fl_2.bmp		
19.	4_1_fl_2.bmp		
20.	4_2_fl_2.bmp		
21.	4_3_fl_2.bmp		
22.	4_4_fl_2.bmp		
23.	4_5_fl_2.bmp		
24.	4_6_fl_2.bmp		

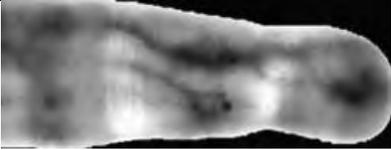
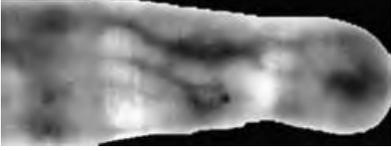
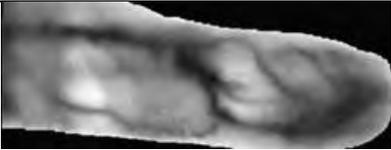
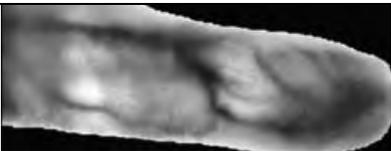
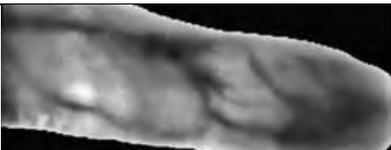
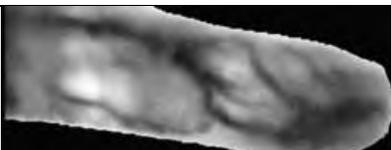
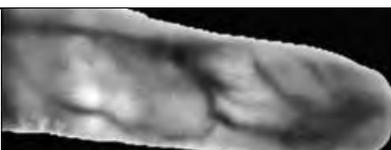
Tabel A. 4 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
25.	5_1_fl_2.bmp		
26.	5_2_fl_2.bmp		
27.	5_3_fl_2.bmp		
28.	5_4_fl_2.bmp		
29.	5_5_fl_2.bmp		
30.	5_6_fl_2.bmp		
31.	6_1_fl_2.bmp		
32.	6_2_fl_2.bmp		

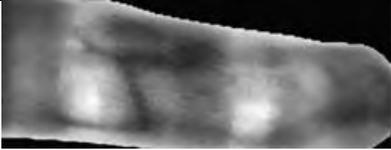
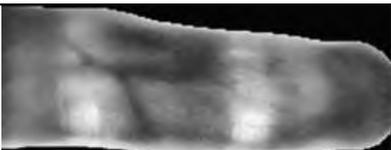
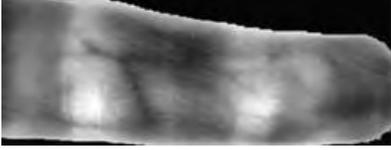
Tabel A. 5 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
33.	6_3_fl_2.bmp		
34.	6_4_fl_2.bmp		
35.	6_5_fl_2.bmp		
36.	6_6_fl_2.bmp		
37.	7_1_fl_2.bmp		
38.	7_2_fl_2.bmp		
39.	7_3_fl_2.bmp		
40.	7_4_fl_2.bmp		

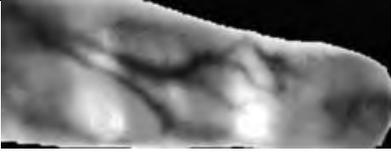
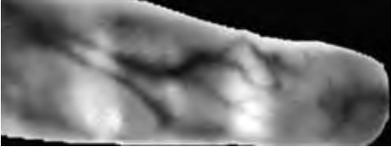
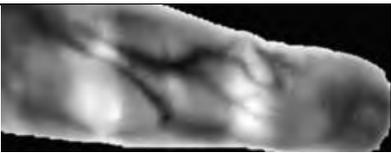
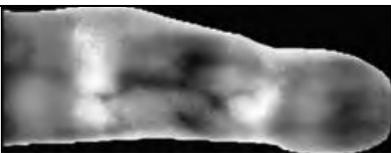
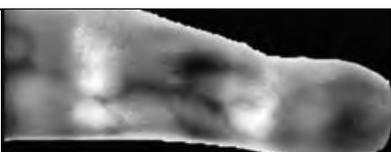
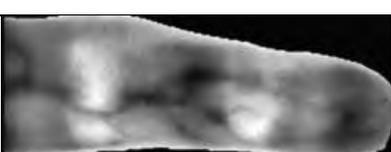
Tabel A. 6 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
41.	7_5_fl_2.bmp		
42.	7_6_fl_2.bmp		
43.	8_1_fl_2.bmp		
44.	8_2_fl_2.bmp		
45.	8_3_fl_2.bmp		
46.	8_4_fl_2.bmp		
47.	8_5_fl_2.bmp		
48.	8_6_fl_2.bmp		

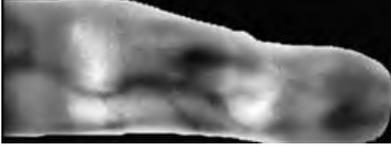
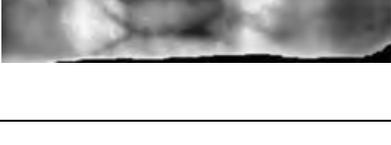
Tabel A. 7 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
49.	9_1_fl_2.bmp		
50.	9_2_fl_2.bmp		
51.	9_3_fl_2.bmp		
52.	9_4_fl_2.bmp		
53.	9_5_fl_2.bmp		
54.	9_6_fl_2.bmp		
55.	10_1_fl_2.bmp		
56.	10_2_fl_2.bmp		

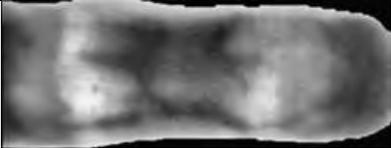
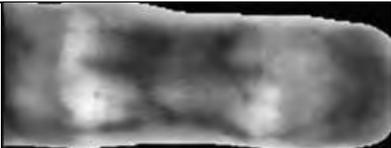
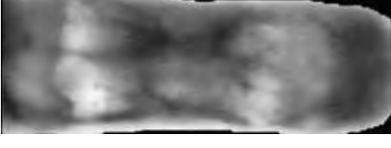
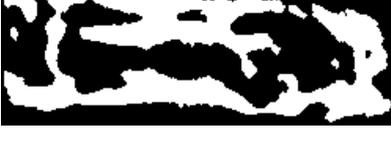
Tabel A. 8 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
57.	10_3_fl_2.bmp		
58.	10_4_fl_2.bmp		
59.	10_5_fl_2.bmp		
60.	10_6_fl_2.bmp		
61.	11_1_fl_2.bmp		
62.	11_2_fl_2.bmp		
63.	11_3_fl_2.bmp		
64.	11_4_fl_2.bmp		

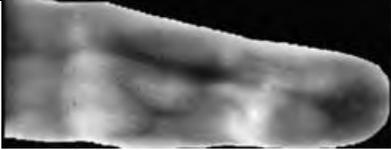
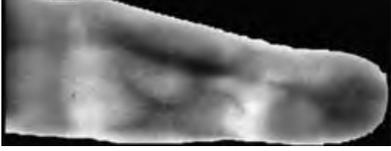
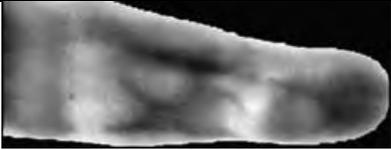
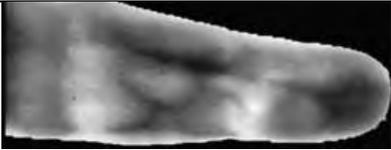
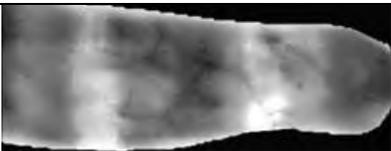
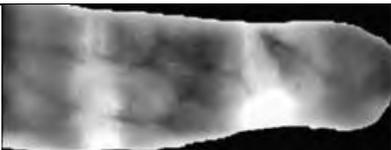
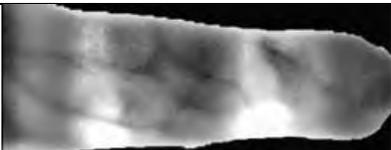
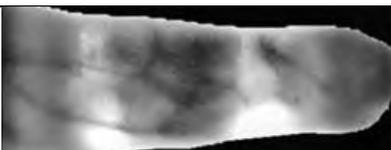
Tabel A. 9 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
65.	11_5_fl_2.bmp		
66.	11_6_fl_2.bmp		
67.	12_1_fl_2.bmp		
68.	12_2_fl_2.bmp		
69.	12_3_fl_2.bmp		
70.	12_4_fl_2.bmp		
71.	12_5_fl_2.bmp		
72.	12_6_fl_2.bmp		

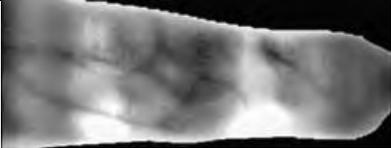
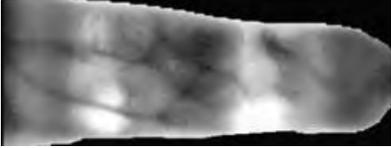
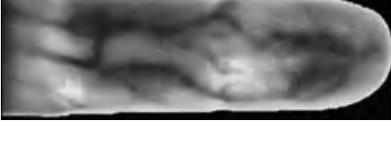
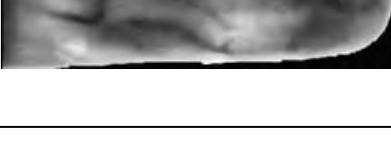
Tabel A. 10 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
73.	13_1_fl_2.bmp		
74.	13_2_fl_2.bmp		
75.	13_3_fl_2.bmp		
76.	13_4_fl_2.bmp		
77.	13_5_fl_2.bmp		
78.	13_6_fl_2.bmp		
79.	14_1_fl_2.bmp		
80.	14_2_fl_2.bmp		

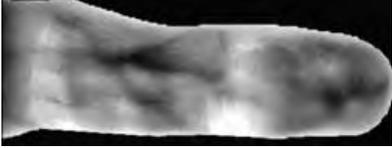
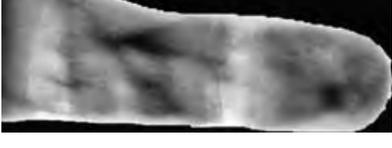
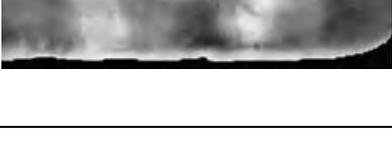
Tabel A. 11 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
81.	14_3_fl_2.bmp		
82.	14_4_fl_2.bmp		
83.	14_5_fl_2.bmp		
84.	14_6_fl_2.bmp		
85.	15_1_fl_2.bmp		
86.	15_2_fl_2.bmp		
87.	15_3_fl_2.bmp		
88.	15_4_fl_2.bmp		

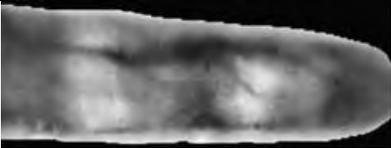
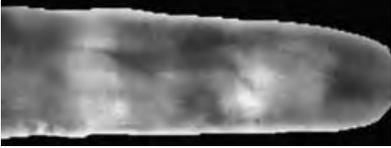
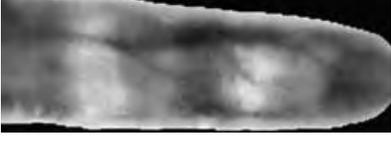
Tabel A. 12 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
89.	15_5_fl_2.bmp		
90.	15_6_fl_2.bmp		
91.	16_1_fl_2.bmp		
92.	16_2_fl_2.bmp		
93.	16_3_fl_2.bmp		
94.	16_4_fl_2.bmp		
95.	16_5_fl_2.bmp		
96.	16_6_fl_2.bmp		

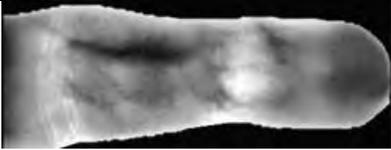
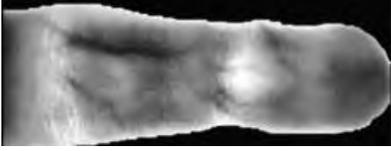
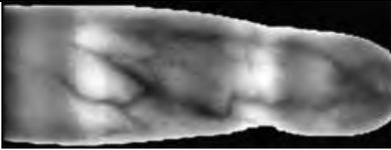
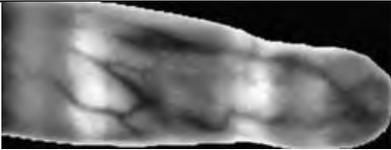
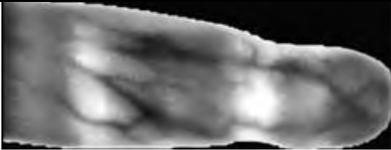
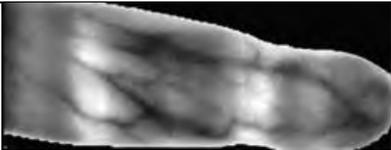
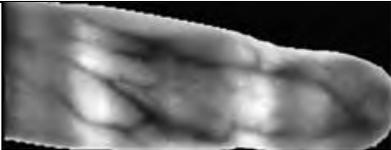
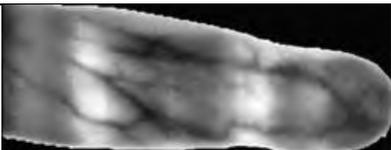
Tabel A. 13 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
97.	17_1_fl_2.bmp		
98.	17_2_fl_2.bmp		
99.	17_3_fl_2.bmp		
100.	17_4_fl_2.bmp		
101.	17_5_fl_2.bmp		
102.	17_6_fl_2.bmp		
103.	18_1_fl_2.bmp		
104.	18_2_fl_2.bmp		

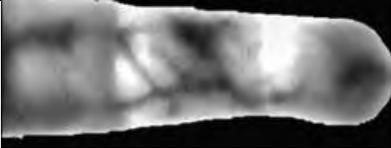
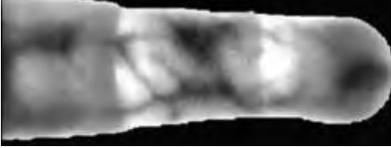
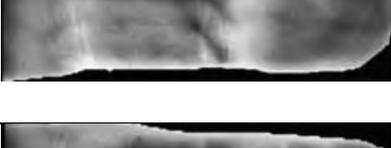
Tabel A. 14 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
105.	18_3_fl_2.bmp		
106.	18_4_fl_2.bmp		
107.	18_5_fl_2.bmp		
108.	18_6_fl_2.bmp		
109.	19_1_fl_2.bmp		
110.	19_2_fl_2.bmp		
111.	19_3_fl_2.bmp		
112.	19_4_fl_2.bmp		

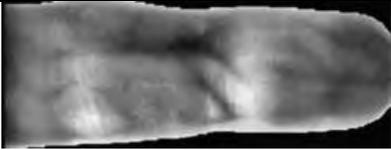
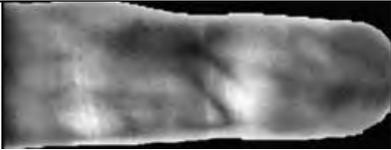
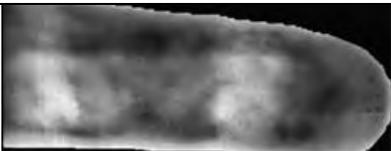
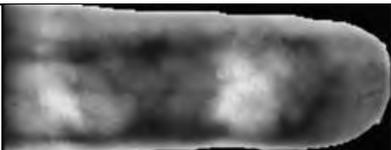
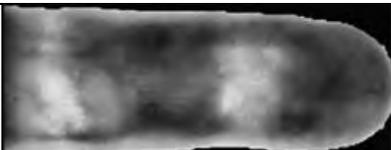
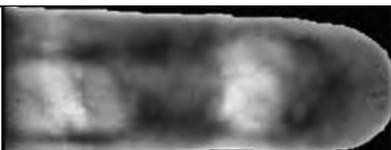
Tabel A. 15 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
113.	19_5_fl_2.bmp		
114.	19_6_fl_2.bmp		
115.	20_1_fl_2.bmp		
116.	20_2_fl_2.bmp		
117.	20_3_fl_2.bmp		
118.	20_4_fl_2.bmp		
119.	20_5_fl_2.bmp		
120.	20_6_fl_2.bmp		

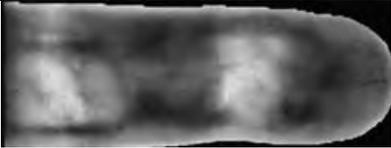
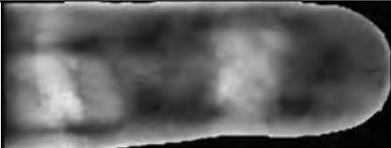
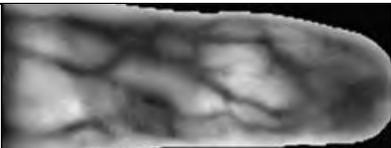
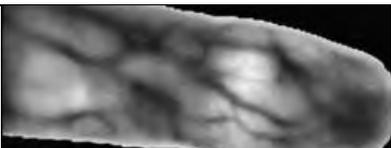
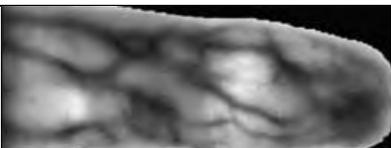
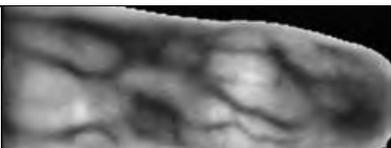
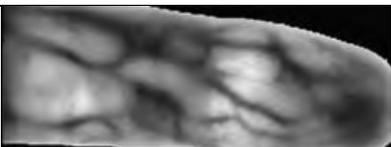
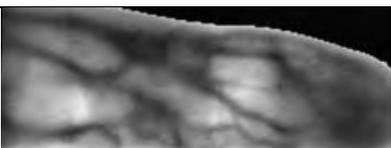
Tabel A. 16 Citra Grayscale dan Citra Biner Hasil Tahap Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
121.	21_1_fl_2.bmp		
122.	21_2_fl_2.bmp		
123.	21_3_fl_2.bmp		
124.	21_4_fl_2.bmp		
125.	21_5_fl_2.bmp		
126.	21_6_fl_2.bmp		
127.	22_1_fl_2.bmp		
128.	22_2_fl_2.bmp		

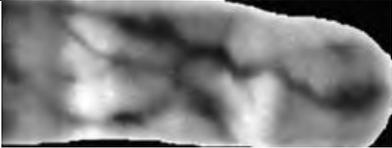
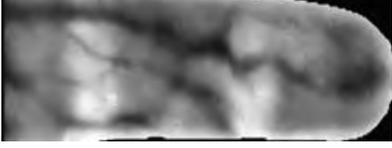
Tabel A. 17 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
129.	22_3_fl_2.bmp		
130.	22_4_fl_2.bmp		
131.	22_5_fl_2.bmp		
132.	22_6_fl_2.bmp		
133.	23_1_fl_2.bmp		
134.	23_2_fl_2.bmp		
135.	23_3_fl_2.bmp		
136.	23_4_fl_2.bmp		

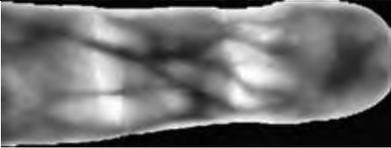
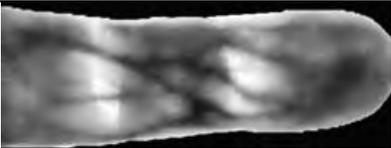
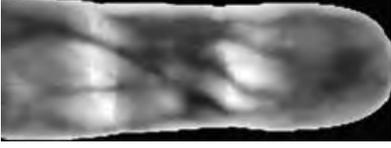
Tabel A. 18 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
137.	23_5_fl_2.bmp		
138.	23_6_fl_2.bmp		
139.	24_1_fl_2.bmp		
140.	24_2_fl_2.bmp		
141.	24_3_fl_2.bmp		
142.	24_4_fl_2.bmp		
143.	24_5_fl_2.bmp		
144.	24_6_fl_2.bmp		

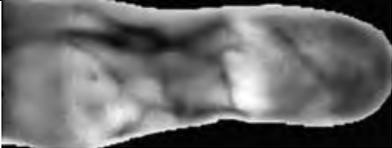
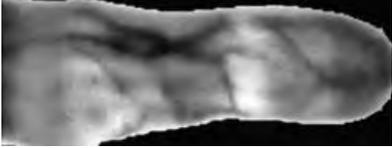
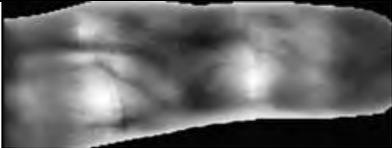
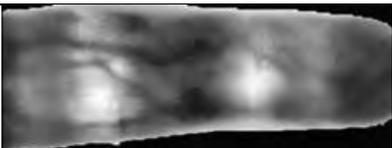
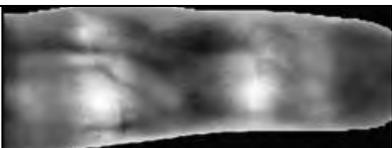
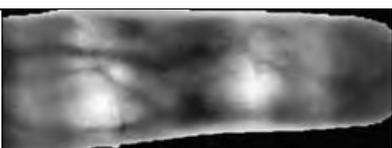
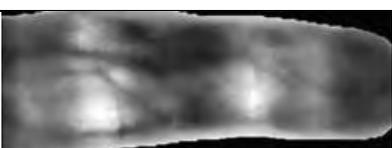
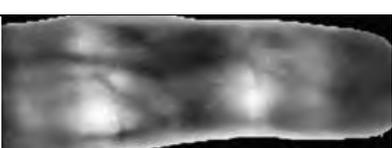
Tabel A. 19 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
145.	25_1_fl_2.bmp		
146.	25_2_fl_2.bmp		
147.	25_3_fl_2.bmp		
148.	25_4_fl_2.bmp		
149.	25_5_fl_2.bmp		
150.	25_6_fl_2.bmp		
151.	26_1_fl_2.bmp		
152.	26_2_fl_2.bmp		

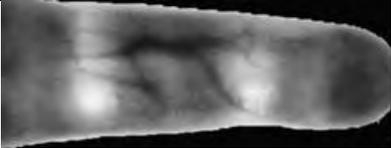
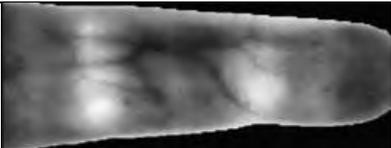
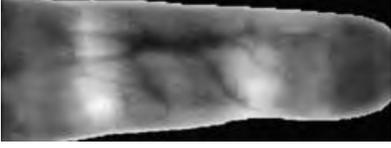
Tabel A. 20 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
153.	26_3_fl_2.bmp		
154.	26_4_fl_2.bmp		
155.	26_5_fl_2.bmp		
156.	26_6_fl_2.bmp		
157.	27_1_fl_2.bmp		
158.	27_2_fl_2.bmp		
159.	27_3_fl_2.bmp		
160.	27_4_fl_2.bmp		

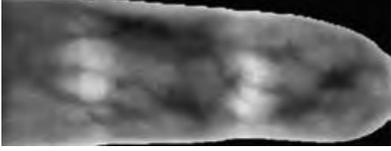
Tabel A. 21 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
161.	27_5_fl_2.bmp		
162.	27_6_fl_2.bmp		
163.	28_1_fl_2.bmp		
164.	28_2_fl_2.bmp		
165.	28_3_fl_2.bmp		
166.	28_4_fl_2.bmp		
167.	28_5_fl_2.bmp		
168.	28_6_fl_2.bmp		

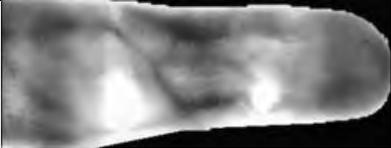
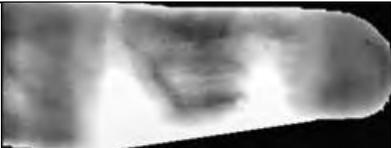
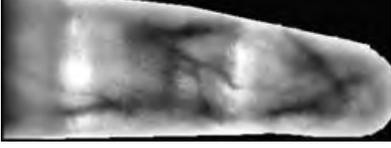
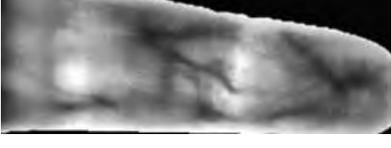
Tabel A. 22 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
169.	29_1_fl_2.bmp		
170.	29_2_fl_2.bmp		
171.	29_3_fl_2.bmp		
172.	29_4_fl_2.bmp		
173.	29_5_fl_2.bmp		
174.	29_6_fl_2.bmp		
175.	30_1_fl_2.bmp		
176.	30_2_fl_2.bmp		

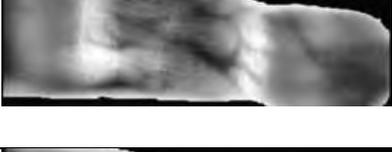
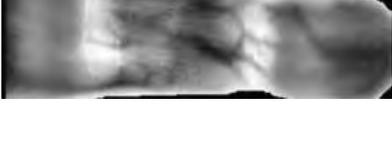
Tabel A. 23 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
177.	30_3_fl_2.bmp		
178.	30_4_fl_2.bmp		
179.	30_5_fl_2.bmp		
180.	30_6_fl_2.bmp		
181.	31_1_fl_2.bmp		
182.	31_2_fl_2.bmp		
183.	31_3_fl_2.bmp		
184.	31_4_fl_2.bmp		

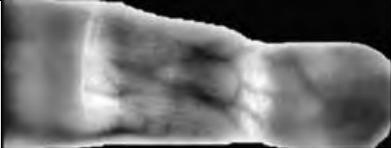
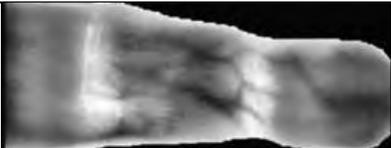
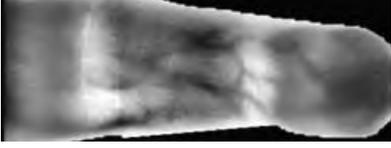
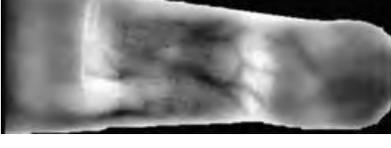
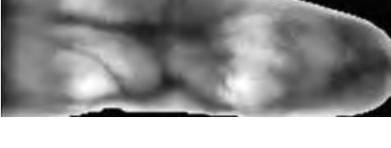
Tabel A. 24 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
185.	31_5_fl_2.bmp		
186.	31_6_fl_2.bmp		
187.	32_1_fl_2.bmp		
188.	32_2_fl_2.bmp		
189.	32_3_fl_2.bmp		
190.	32_4_fl_2.bmp		
191.	32_5_fl_2.bmp		
192.	32_6_fl_2.bmp		

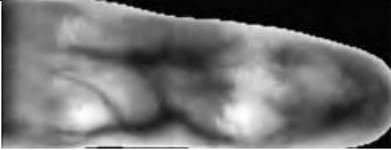
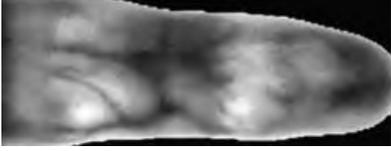
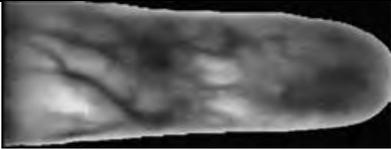
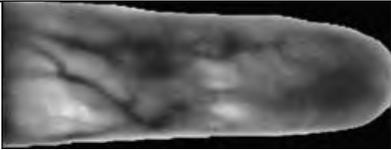
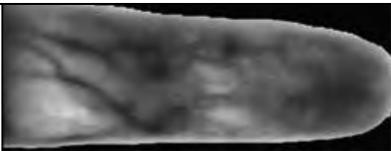
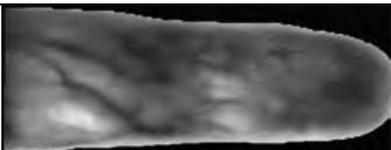
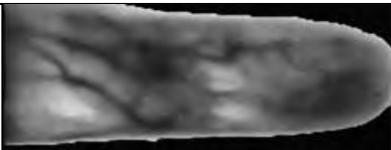
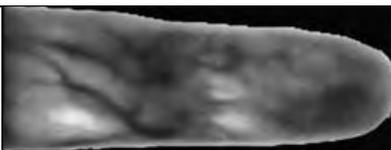
Tabel A. 25 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
193.	33_1_fl_2.bmp		
194.	33_2_fl_2.bmp		
195.	33_3_fl_2.bmp		
196.	33_4_fl_2.bmp		
197.	33_5_fl_2.bmp		
198.	33_6_fl_2.bmp		
199.	34_1_fl_2.bmp		
200.	34_2_fl_2.bmp		

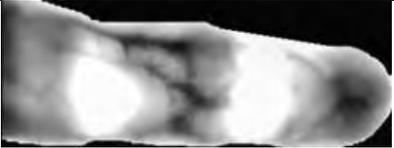
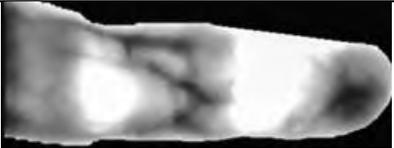
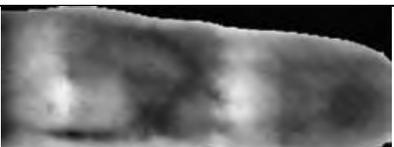
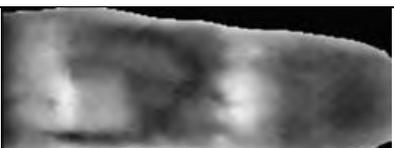
Tabel A. 26 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
201.	34_3_fl_2.bmp		
202.	34_4_fl_2.bmp		
203.	34_5_fl_2.bmp		
204.	34_6_fl_2.bmp		
205.	35_1_fl_2.bmp		
206.	35_2_fl_2.bmp		
207.	35_3_fl_2.bmp		
208.	35_4_fl_2.bmp		

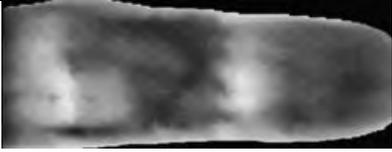
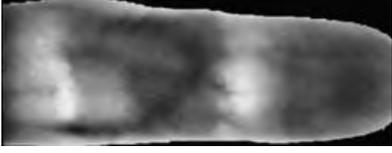
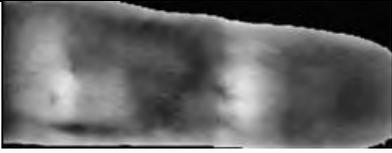
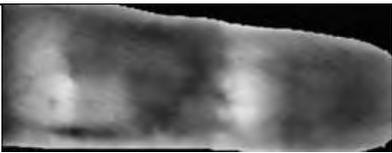
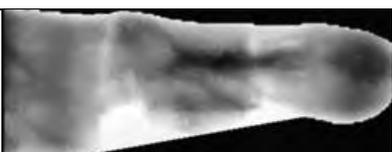
Tabel A. 27 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
209.	35_5_fl_2.bmp		
210.	35_6_fl_2.bmp		
211.	36_1_fl_2.bmp		
212.	36_2_fl_2.bmp		
213.	36_3_fl_2.bmp		
214.	36_4_fl_2.bmp		
215.	36_5_fl_2.bmp		
216.	36_6_fl_2.bmp		

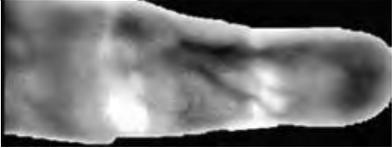
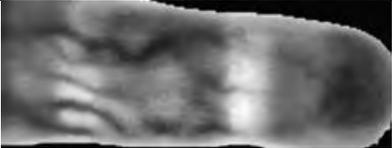
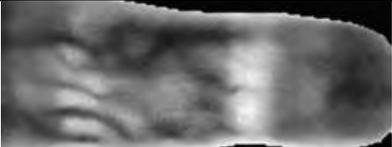
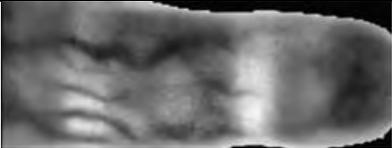
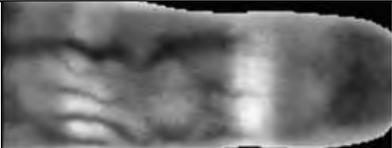
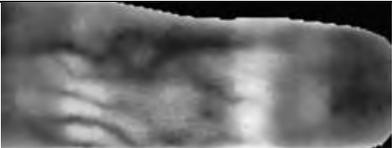
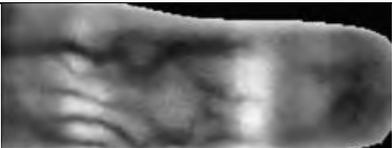
Tabel A. 28 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
217.	37_1_fl_2.bmp		
218.	37_2_fl_2.bmp		
219.	37_3_fl_2.bmp		
220.	37_4_fl_2.bmp		
221.	37_5_fl_2.bmp		
222.	37_6_fl_2.bmp		
223.	38_1_fl_2.bmp		
224.	38_2_fl_2.bmp		

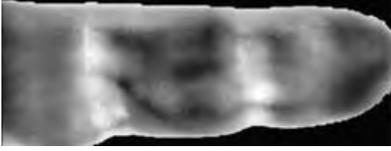
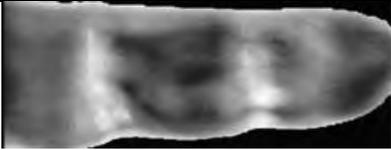
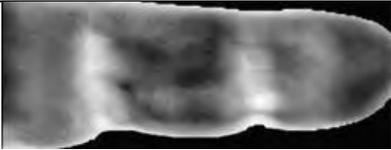
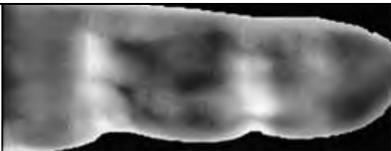
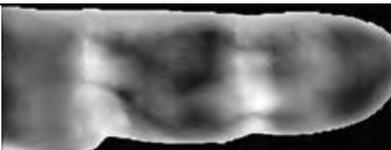
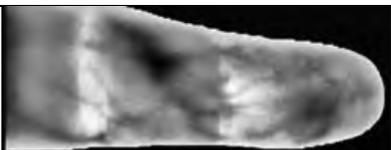
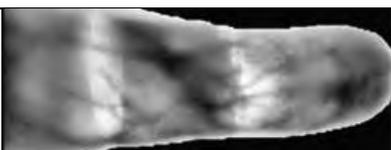
Tabel A. 29 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
225.	38_3_fl_2.bmp		
226.	38_4_fl_2.bmp		
227.	38_5_fl_2.bmp		
228.	38_6_fl_2.bmp		
229.	39_1_fl_2.bmp		
230.	39_2_fl_2.bmp		
231.	39_3_fl_2.bmp		
232.	39_4_fl_2.bmp		

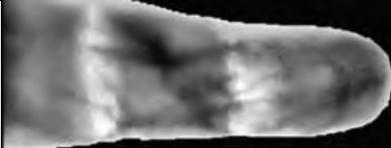
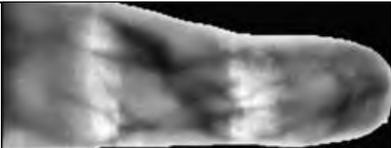
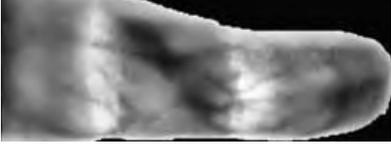
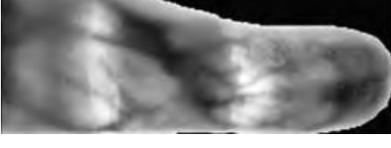
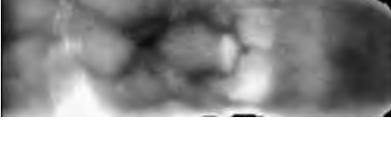
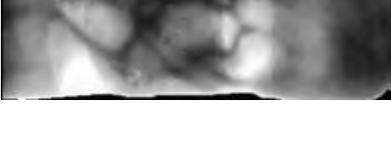
Tabel A. 30 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
233.	39_5_fl_2.bmp		
234.	39_6_fl_2.bmp		
235.	40_1_fl_2.bmp		
236.	40_2_fl_2.bmp		
237.	40_3_fl_2.bmp		
238.	40_4_fl_2.bmp		
239.	40_5_fl_2.bmp		
240.	40_6_fl_2.bmp		

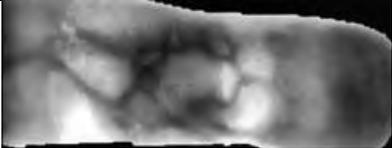
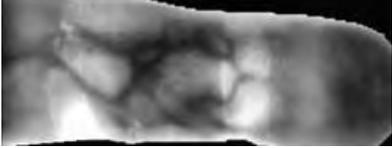
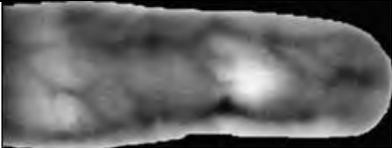
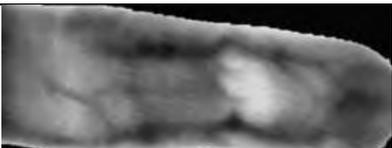
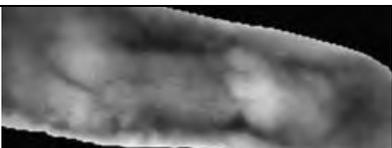
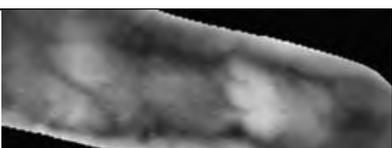
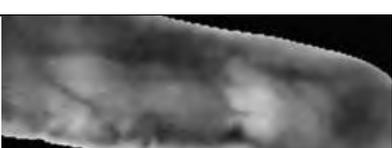
Tabel A. 31 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
241.	41_1_fl_2.bmp		
242.	41_2_fl_2.bmp		
243.	41_3_fl_2.bmp		
244.	41_4_fl_2.bmp		
245.	41_5_fl_2.bmp		
246.	41_6_fl_2.bmp		
247.	42_1_fl_2.bmp		
248.	42_2_fl_2.bmp		

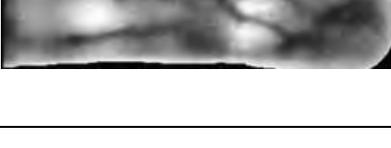
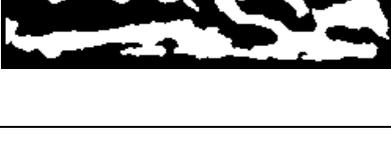
Tabel A. 32 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
249.	42_3_f1_2.bmp		
250.	42_4_f1_2.bmp		
251.	42_5_f1_2.bmp		
252.	42_6_f1_2.bmp		
253.	43_1_f1_2.bmp		
254.	43_2_f1_2.bmp		
255.	43_3_f1_2.bmp		
256.	43_4_f1_2.bmp		

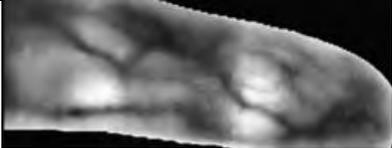
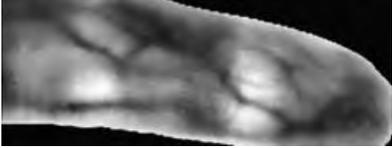
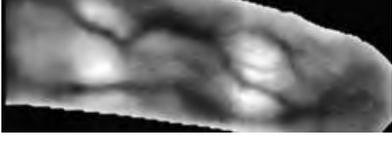
Tabel A. 33 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
257.	43_5_fl_2.bmp		
258.	43_6_fl_2.bmp		
259.	44_1_fl_2.bmp		
260.	44_2_fl_2.bmp		
261.	44_3_fl_2.bmp		
262.	44_4_fl_2.bmp		
263.	44_5_fl_2.bmp		
264.	44_6_fl_2.bmp		

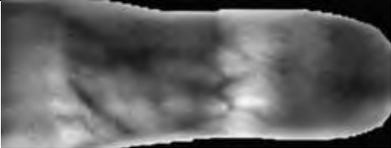
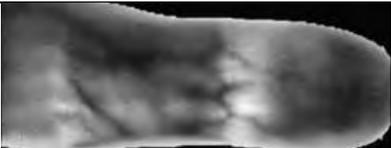
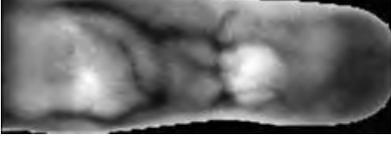
Tabel A. 34 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
265.	45_1_fl_2.bmp		
266.	45_2_fl_2.bmp		
267.	45_3_fl_2.bmp		
268.	45_4_fl_2.bmp		
269.	45_5_fl_2.bmp		
270.	45_6_fl_2.bmp		
271.	46_1_fl_2.bmp		
272.	46_2_fl_2.bmp		

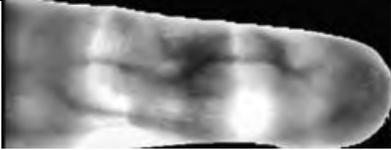
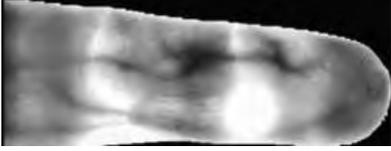
Tabel A. 35 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
273.	46_3_fl_2.bmp		
274.	46_4_fl_2.bmp		
275.	46_5_fl_2.bmp		
276.	46_6_fl_2.bmp		
277.	47_1_fl_2.bmp		
278.	47_2_fl_2.bmp		
279.	47_3_fl_2.bmp		
280.	47_4_fl_2.bmp		

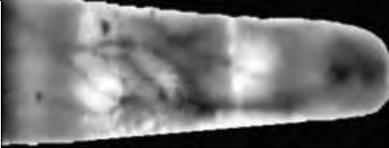
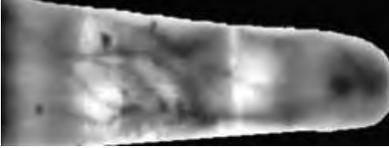
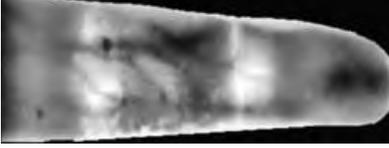
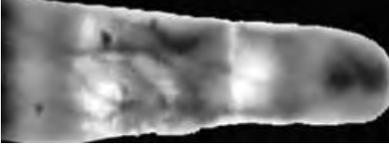
Tabel A. 36 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
281.	47_5_fl_2.bmp		
282.	47_6_fl_2.bmp		
283.	48_1_fl_2.bmp		
284.	48_2_fl_2.bmp		
285.	48_3_fl_2.bmp		
286.	48_4_fl_2.bmp		
287.	48_5_fl_2.bmp		
288.	48_6_fl_2.bmp		

Tabel A. 37 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
289.	49_1_fl_2.bmp		
290.	49_2_fl_2.bmp		
291.	49_3_fl_2.bmp		
292.	49_4_fl_2.bmp		
293.	49_5_fl_2.bmp		
294.	49_6_fl_2.bmp		
295.	50_1_fl_2.bmp		
296.	50_2_fl_2.bmp		

Tabel A. 38 Citra Grayscale dan Citra Hasil Tahap Biner Praproses

No.	Nama Citra	Citra Grayscale	Citra Biner
297.	50_3_f1_2.bmp		
298.	50_4_f1_2.bmp		
299.	50_5_f1_2.bmp		
300.	50_6_f1_2.bmp		

BAB V

KESIMPULAN DAN SARAN

Bab ini membahas mengenai kesimpulan yang dapat diambil dari hasil uji coba yang telah dilakukan sebagai jawaban dari rumusan masalah yang telah dikemukakan. Selain itu, juga terdapat saran yang ditujukan untuk pembangunan penelitian lebih lanjut.

5.1 Kesimpulan

Dari hasil uji coba yang telah dilakukan terhadap sistem yang dibangun untuk proses pengenalan pembuluh darah jari pada penelitian ini, dapat diambil kesimpulan sebagai berikut:

1. Metode pembentukan deskriptor histogram dari fitur SIFT untuk pengenalan pembuluh darah jari dengan metode klasifikasi LVQ telah menunjukkan hasil yang relatif baik yaitu dengan nilai akurasi yang diperoleh mencapai 97,34% , nilai TPR mencapai 0,8200, dan nilai FPR yang rendah yaitu sebesar 0,0135.
2. Metode pembentukan deskriptor histogram LEBP yang merupakan gabungan dari fitur LmBP dan fitur LdBP untuk pengenalan pembuluh darah jari dengan metode klasifikasi LVQ juga telah menunjukkan hasil yang relatif baik yaitu dengan hasil akurasi sebesar 97,32%, TPR sebesar 0,8300, dan FPR sebesar 0,0137.
3. Teknik fusi untuk membentuk fitur gabungan SIFT-LEBP telah menunjukkan hasil yang relatif baik dengan hasil akurasi yang tidak jauh berbeda dengan hasil yang diperoleh jika hanya menggunakan fitur tunggal SIFT atau LEBP. Hal tersebut terlihat dari hasil akurasi yang diperoleh pada kondisi optimum yaitu sebesar 97,50%.
4. Fitur SIFT-LEBP yang diusulkan sebagai fitur untuk sistem pengenalan pembuluh darah jari memberikan hasil yang relatif lebih baik jika dibandingkan dengan hanya menggunakan fitur tunggal. Hal tersebut dapat dilihat dari nilai TPR sebesar 0,9400, dan nilai FPR sebesar 0,0128.

5. Parameter yang digunakan pada proses klasifikasi menggunakan LVQ dapat mempengaruhi hasil akurasi, TPR, maupun FPR. Parameter-parameter tersebut yaitu jumlah hidden layer, jumlah epochs, dan nilai learning rate.
6. Pada penelitian ini jumlah cluster untuk pembentukan histogram fitur SIFT diperoleh paling optimal ketika ditetapkan sejumlah 50. Sedangkan parameter pada klasifikasi LVQ diperoleh hasil optimal ketika jumlah hidden layer ditetapkan sebanyak 500, jumlah epoch ditetapkan sebanyak 350, dan nilai learning rate sebesar 0,01.

5.2 Saran

Saran yang diberikan untuk pengembangan sistem pengenalan pembuluh darah jari dalam penelitian ini antara lain:

1. Perbaiki metode ekstraksi fitur LdBP untuk meningkatkan hasil akurasi dan TPR serta menurunkan nilai FPR sehingga hasil akurasi dan TPR dari fitur LEBP dapat meningkat serta nilai FPR dari LEBP dapat turun.
2. Dapat dikembangkan metode lain pada tahap praproses yang memungkinkan peningkatan kinerja sistem.
3. Dapat digunakan suatu metode tertentu untuk penentuan parameter optimum secara otomatis.

DAFTAR PUSTAKA

- Hendriks, C. (2003). *Structure Characterization Using Mathematical Morphology*. Cris Luengo.
- Bashyal, S., & Venayagamoorthy, G. (2008). Recognition of facial expressions using Gabor wavelets and learning vector quantization. *Engineering Applications of Artificial Intelligence* 21.7, 1056-1064.
- Castanedo, F. (2013). A review of data fusion techniques. *The Scientific World Journal*.
- de Vries, G.-J., Pauws, S., & Biehl, M. (2015). Facial Expression Recognition Using Learning Vector Quantization. *Springer International Publishing*, 760-771.
- Frangi, A., Niessen, W., Vincken, K., & Viergever, M. (1998). Multiscale vessel enhancement filtering. *Medical Image Computing and Computer-Assisted Intervention—MICCAI'98*, 130-137.
- Haas, S., Donner, R., Burner, A., Holzer, M., & Langs, G. (2011). Superpixel-Based interest points for effective bags of visual words medical image retrieval. *Medical Content-Based Retrieval for Clinical Decision Support*, 58-68.
- Jimenez-Carretero, D., Santos, A., Kerkstra, S., Dewi Rudyanto, R., & Ledesma-Carbayo, M. (2013). 3D Frangi-based lung vessel enhancement filter penalizing airways. *Biomedical Imaging (ISBI), 2013 IEEE 10th International Symposium, X*, 926-929.
- Khellat-kihel, S., abrishamba, R., Cardoso, N., Monteiro, J., & Benyettou, M. (2014). Finger vein recognition using Gabor filter and Support Vector Machine. *Image Processing, Applications and Systems Conference (IPAS)*, 1-6.
- Khusnuliawati, H., Wijaya, A., & Soelaiman, R. (2014). *IDENTIFIKASI PARAMETER OPTIMAL DARI METODE FUZZY C-MEANS DAN ESTIMASI ROBUST SPASIAL PADA SEGMENTASI CITRA DENGAN NOISE*. Surabaya: Teknik Informatika ITS.

- Kim, H.-G., Lee, E., Yoon, G.-J., & Yang, S.-D. (2012). Illumination normalization for SIFT based finger vein authentication. *Advances in Visual Computing*, 21-30.
- Kumar, A., & Zhou, Y. (2012). Human Identification Using Finger Images. *XXI(TRANSACTIONS ON IMAGE PROCESSING)*.
- Ladoux, P.-O., Rosenberger, C., & Dorizzi, B. (2009). Palm vein verification system based on SIFT matching. *Advances in Biometrics*, 1290-1298.
- Liu, F., Yang, G., Yin, Y., & Wang, S. (2014). Singular value decomposition based minutiae matching method for finger vein recognition. *Neurocomputing*, 75-89.
- Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision* 60.2 , 91-110.
- Meng, X., Yang, G., Yin, Y., & Xiao, R. (2012). Finger vein recognition based on local directional code. *Sensors* 12, 14937-14952.
- Miura, N., Nagasaka, A., & Miyatake, T. (2004). Feature extraction of finger-vein patterns based on repeated line tracking and its application to personal identification. *Machine Vision and Applications*, 194-203.
- Miura, N., Nagasaka, A., & Miyatake, T. (2007). Extraction of finger-vein patterns using maximum curvature points in image profiles. *IEICE TRANSACTIONS on Information and Systems*, 1185-1194.
- Ojala, T., Pietikäinen, M., & Mäenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *Pattern Analysis and Machine Intelligence*, 971-987.
- Pang, S., Yin, Y., Yang, G., & Li, Y. (2012). Rotation invariant finger vein recognition. *Biometric recognition*, 151-156.
- Park, K. (2012). Finger vein recognition by combining global and local features based on SVM. *Computing and Informatics* 30.2, 295-309.
- Rosdi, B., Shing, C., & Suandi, S. (2011). Finger vein recognition using local line binary pattern. *Sensors* 11, 11357-11371.
- Song, W., Kim, T., Kim, H., Choi, J., Kong, H.-J., & Lee, S.-R. (2011). A finger-vein verification system using mean curvature. *Pattern Recognition Letters*, 1541-1547.

- Wang, H., Wang, X., Zheng, J., Deller, J., Peng, H., Zhu, L., . . . Bao, H. (2014). Video object matching across multiple non-overlapping camera views based on multi-feature fusion and incremental learning. *Pattern Recognition* 47, 3841-3851.
- Xu, X., & Zhao, Y. (2015). Multimodal Face Recognition for Profile Views Based on SIFT and LBP. *Face and Facial Expression Recognition from Real World Videos.*, 20-30.
- Yakno, M., Saleh, J., & Rosdi, B. (2011). Low contrast hand vein image enhancement. *In Signal and Image Processing Applications (ICSIPA)*, 390-392.
- Yang, L., Yang, G., Yin, Y., & Zhou, L. (2014). A Survey of Finger Vein Recognition . *Biometric Recognition 9th Chinese Conference. IX.* Shenyang: Springer.
- Yu, J., Qin, Z., Wan, T., & Zhang, X. (2013). Feature integration analysis of bag-of-features model for image retrieval. *Neurocomputing*, 355-364.
- Zhang, B., Gao, Y., Zhao, S., & Liu, J. (2010). Local derivative pattern versus local binary pattern: face recognition with high-order local pattern descriptor. *Image Processing, IEEE Transactions on*, 19, 533-544.

BIODATA PENULIS



Penulis, Hardika Khusnuliawati, lahir di kota Sukoharjo pada tanggal 31 Agustus 1992. Penulis adalah anak pertama dari tiga bersaudara dan dibesarkan di kota Sukoharjo, Jawa Tengah.

Penulis menempuh pendidikan formal di SD Negeri Tawang 1 (1998-2004), SMPN 1 Tawang Sari (2004-2007), SMAN 1 Sukoharjo (2007-2010). Pada tahun 2010, penulis memulai pendidikan S1 jurusan Teknik

Informatika Fakultas Teknologi Informasi di Institut Teknologi Sepuluh Nopember Surabaya, Jawa Timur.

Di jurusan Teknik Informatika, penulis mengambil bidang minat Komputasi Cerdas dan Visualisasi. Penulis juga aktif dalam organisasi kemahasiswaan seperti Himpunan Mahasiswa Teknik Computer (HMTC) sebagai staf dan Keluarga Muslim Informatika sebagai sekretaris departemen. Dan penulis juga beberapa kali menjadi semifinalis dalam beberapa perlombaan diantaranya IT COMPFEST pada tahun 2012 dan Gemastik pada tahun 2013. Penulis dapat dihubungi melalui alamat email hardika.khusnulia@gmail.com