DISERTASI

PENGEMBANGAN MODEL OPTIMASI UNTUK PENGKLASIFIKASIAN CITRA KELAINAN RETINA

Erwin
NIM: 0304-36-81621-002

PROGRAM STUDI DOKTOR ILMU TEKNIK
FAKULTAS TEKNIK
UNIVERSITAS SRIWIJAYA
2019

DISERTASI

PENGEMBANGAN MODEL OPTIMASI UNTUK PENGKLASIFIKASIAN CITRA KELAINAN RETINA

Diajukan untuk memenuhi salah satu syarat memperoleh gelar
Doktor Ilmu Teknik

Erwin
NIM: 0304-36-81621-002

PROGRAM STUDI DOKTOR ILMU TEKNIK
FAKULTAS TEKNIK
UNIVERSITAS SRIWIJAYA
2019

RINGKASAN

PENGEMBANGAN MODEL OPTIMASI UNTUK PENGKLASIFIKASIAN CITRA KELAINAN RETINA
Karya tulis ilmiah berupa Disertasi, 19 Juli 2019

Erwin, dibimbing oleh Prof. Drs. Saparudin, M.T., Ph.D

Program Studi Ilmu Teknik Program Doktor, Fakultas Teknik
Universitas Sriwijaya

xv + 150 halaman, 41 tabel, 50 gambar

Biometrik merupakan teknologi yang menggunakan karakteristik biologis untuk mengindentifikasi individu. Retina menjadi salah satu karakteristik yang bersifat unik dan dapat digunakan untuk diagnosis penyakit ophthalmologic. Tahapan proses pengklasifikasian citra kelainan retina dimulai dengan segmentasi, ekstraksi fitur dan klasifikasi. Pengembangan model optimasi pada tahap segmentasi dibangun dengan menggabungkan metode multilevel thresholding dan improved harmony search algorithm(MT-IHSA). Setelah proses segmentasi, proses berikutnya dilanjutkan dengan proses enhancement dengan melakukan penyesuaian intensitas secara otomatis dengan menggunakan pemerataan histogram dan dilakukan secara parsial untuk foreground yang menjadi fokus ciri suatu kelas penyakit yang dinamakan parsial foreground histogram equalization. Tahapan berikutnya, dilakukan ekstraksi fitur dengan mengunakan ekstraksi bentuk, tekstur dan warna. Selanjutnya, dilakukan pengklasifikasian citra dengan metode Convolution Neural Network(CNN). Pengujian setiap tahapan menggunakan dataset DRIVE dan STARE dan 5(lima) kelas penyakit, yaitu Central Retinal Vein Occlusion, Normal, Background Diabetic Retinopathy, Proliverative Diabetic Retinopathy dan Coats. Segmentasi yang dihasilkan oleh MT-IHSA memberikan output yang sangat baik. Semakin tinggi thresholding yang diberikan, maka nilai PSNR semakin tinggi yaitu untuk thresholding 2, 3, 4, dan 5 berturut-turut nilai PSNR adalah 38,602; 39,628; 41,032 dan 42,107. Nilai rata-rata empat parameter pada proses ekstraksi fitur yang lebih besar dari kelima penyakit setelah diuji yaitu penyakit Central Retinal Vein Occlusion dengan kontras = 0,312, korelasi = 0,978, energi = 0,197, serta homogenitas = 0,926 dan nilai rata-rata yang lebih kecil yaitu penyakit Proliverative Diabetic Retinopathy dengan kontras = 0,211, korelasi = 0,980, energi = 0,148 serta homogenitas = 0,928. Hasil klasifikasi penyakit kelainan retina dengan CNN diperoleh, untuk thresholding level 2(Th2), specificity mencapai 100% karena tidak terdapat kesalahan dalam hasil klasifikasi, sedangkan untuk sensitivity menghasilkan nilai 60% disebabkan ada 2 citra yang salah mendeteksi kelas BDR dan nilai akurasi mencapai 92%. Percobaan klasifikasi untuk 3 level sudah mencapai masing-masing 100% untuk specificity, sensitivity dan akurasi.

Kata Kunci: Convolution Neural Network, Enhancement, Ekstraksi Fitur, Improved Harmony Search Algorithm, Klasifikasi, Multilevel Thresholding, Segmentasi

SUMMARY

Development of Optimization Model for Retinal Abnormalities Image Classification
Scientific Paper in the form of Disertation, 19 July 2019

Erwin, supervised by Prof. Drs. Saparudin, M.T., Ph.D

Doctoral Program of Engineering, Faculty of Engineering
Universitas Sriwijaya

xv + 150 pages, 41 table, 50 pictures

Biometrics is a technology that uses biological characteristics to identify individuals. Retina is a unique characteristic that can be used to diagnose ophthalmologic diseases. The stages of the image classification process for retinal abnormalities begin with segmentation, feature extraction and classification. The development of optimization models was done at the segmentation stage by combining multilevel thresholding and improved harmony search algorithm methods (MT-IHSA). After the segmentation process, image enhancement was done by automatically adjusting the intensity of the image using histogram equalization. This process called partial foreground histogram equalization due to it was done partially at the foreground object because it has unique characteristics of a disease class. The next step is feature extraction using shapes, textures and colors feature extraction method. Furthermore, image classification was done using the Convolution Neural Network (CNN) method. The testing phase was carried out using two dataset, DRIVE and STARE, consisting of 5 classes of diseases Central Retinal Vein Occlusion, Normal, Background Diabetic Retinopathy, Proliverative Diabetic Retinopathy dan Coats. The segmentation result using MT-IHSA provides very good output. The higher the thresholding gives higher PSNR, for thresholding 2, 3, 4, and 5 the PSNR value is 38,602; 39,628; 41,032 and 42,107 respectively. In order to measure the feature extraction process, we use four parameters namely contrast, correlation, energy, and homogeneity. Central Retinal Vein Occlusion has the highest average value of the four parameters with contrast = 0.312, correlation = 0.978, energy = 0.197, and homogeneity = 0.926 where the smallest is Proliverative Diabetic Retinopathy with contrast = 0.211, correlation = 0.980, energy = 0.148 and homogeneity = 0.928. The classification results of retinal abnormalities with CNN were obtained. Using 2 levels of thresholding (Th2), the specificity reached 100% because there were no errors in the classification results, while sensitivity produced a value of 60% because there were 2 images that incorrectly detected and the accuracy value was 92 %. Moreover, Classification result for 3 levels have reached 100% for specificity, sensitivity and accuracy.

Key words: Classification, Convolution Neural Network, Enhancement, Feature Extraction, Improved Harmony Search Algorithm, Multilevel Thresholding, Segmentation

HALAMAN PERNYATAAN INTEGRITAS

Yang bertanda tangan dibawah ini:

Nama : Erwin

NIM : 0304-36-81621-002

Judul : PENGEMBANGAN MODEL OPTIMASI UNTUK PENGKLASIFIKASIAN CITRA KELAINAN RETINA

Menyatakan bahwa Disertasi saya merupakan hasil karya sendiri didampingi Promotor dan bukan hasil penjiplakan/*plagiat*. Apabila ditemukan unsur penjiplakan/plagiat dalam Disertasi ini, maka saya bersedia menerima sanksi akademik dari Universitas Sriwijaya sesuai aturan yang berlaku.

Demikian, pernyataan ini saya buat dalam keadaan sadar dan tanpa ada paksaan dari siapapun.

Inderalaya, Agustus 2019

Erwin

HALAMAN PENGESAHAN

PENGEMBANGAN MODEL OPTIMASI UNTUK PENGKLASIFIKASIAN CITRA KELAINAN RETINA

DISERTASI

Diajukan untuk memenuhi salah satu syarat memperoleh gelar
Doktor Ilmu Teknik

Oleh:

Erwin
NIM. 03043681621002

Telah disetujui

Pada 5 Agustus 2019

Promotor,

Prof. Drs. Saparudin, M.T., Ph.D

Mengetahui
Dekan,

Prof. Ir. Subriyer Nasir, M.S, Ph.D.
NIP. 196009091987031004

HALAMAN PERSETUJUAN

Disertasi berjudul " Pengembangan Model Optimasi untuk Pengklasifikasian Citra Kelainan Retina " telah dipresentasikan di depan tim penguji Sidang Terbuka Upacara Promosi Doktor Prodi Ilmu Teknik Program Doktor Fakultas Teknik Universitas Sriwijaya pada hari Senin, tgl 5 Agustus 2019 dan telah dinyatakan sah dan lulus.

Palembang, Agustus 2019

Promotor:

Prof. Drs. Saparudin, M.T., Ph.D. ( )

Ketua Sidang:

Prof. Dr. Ir. Anis Saggaff, MSCE.
NIP. 196210281989031002 ( )

Anggota Tim Penguji:

1. Prof. Ir. Subriyer Nasir, M.S, Ph.D. ( )
2. Prof. Dr. Ghazali bin Sulong (MSU) ( )
3. Reza Firsandaya Malik, S.T., M.T., Ph.D. ( )
4. Dr. Yusuf Hartono, M.Sc. ( )
5. Prof. Dr. Ir. Nukman, M.T ( )

Mengetahui

Dekan,

Prof. Ir. Subriyer Nasir, M.S, Ph.D.
NIP. 196009091987031004

Koordinator Program Studi

KEMENTERIAN RISET, TEKNOLOGI, DAN PENDIDIKAN TINGGI
UNIVERSITAS SRIWIJAYA
FAKULTAS TEKNIK
PRODI ILMU TEKNIK
PROGRAM DOKTOR

Prof. Dr. Ir. Nukman, M.T
NIP. 195903211987031001

KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Puji dan syukur penulis panjatkan kehadirat Allah SWT, atas segala karunia dan rahmat-Nya sehingga penulis dapat menyelesaikan Disertasi dengan judul "Pengembangan Model Optimasi untuk Pengklasifikasian Citra Kelainan Retina". Pada penyusunan Disertasi ini, penulis banyak mendapatkan ide dan saran serta bantuan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis mengucapkan rasa syukur kepada Allah SWT dan terima kasih kepada yang terhormat:

1. Orang tuaku, Ibu Hj. Khodijah yang telah memberikan do'a dan dukungannya.
2. Istri tercinta Hj. Mursiti bin Sahmad Hasan, ketiga anak-anakku, M. Syardinal, Indah Ramadhani dan Shella Rizka Maharani yang memberikan inspirasi dan motivasi untuk tetap selalu berusaha dan tawakal.
3. Bapak Prof. Dr. Ir. H. Anis Saggaff, MSCE, selaku Rektor Universitas Sriwijaya.
4. Bapak Prof. Ir. H. Zainudin Nawawi, M.M., Ph.D, selaku Wakil Rektor Bidang Akademik Universitas Sriwijaya.
5. Bapak Prof. Ir. Subriyer Nasir, M.S., Ph.D, selaku Dekan Fakultas Teknik Universitas Sriwijaya.
6. Bapak Prof. Dr. Ir. Nukman, M.T, selaku Ketua Program Studi S3-Doktor Ilmu Teknik Universitas Sriwijaya.
7. Bapak Prof. Drs. H. Saparudin, M.T., Ph.D, selaku Promotor Disertasi.
8. Bapak Prof. Ghazali bin Sulong, Dr. Reza Firsandaya Malik, M.T. dan Dr. Yusuf Haroton, selaku Tim Penguji
9. Seluruh Dosen Jurusan Sistem Komputer, Teknik Informatika dan Sistem Informasi Fasilkom Universitas Swijaya
10. Seluruh staf pegawai serta mahasiswa di lingkungan Fasilkom Universitas Sriwijaya.

Semoga disertasi ini bisa bermanfaat bagi pembaca ataupun bagi penulis sendiri. Demikian yang bisa penulis sampaikan.

Wassalamu'alaikum Wr.Wb

Palembang, Juli 2019

Penulis

DAFTAR ISI

RINGKASAN iii
SUMMARY iv
HALAMAN PERNYATAAN INTEGRITAS v
HALAMAN PENGESAHAN vi
HALAMAN PERSETUJUAN vii
KATA PENGANTAR viii
DAFTAR ISI ix
DAFTAR GAMBAR xii
DAFTAR TABEL xiv
BAB I. PENDAHULUAN 1
1.1. Pendahuluan 1
1.2. Latar Belakang 1
1.3. Rumusan Masalah 6
1.4. Tujuan Umum Penelitian 7
1.5. Tujuan Khusus 7
1.6. Batasan Masalah 8
1.7. Manfaat dan Kontribusi Penelitian 9
1.8. Kerangka Penelitian 11
1.9. Sistematika Disertasi 13
BAB II. STUDI PUSTAKA 15
2.1 Pendahuluan 15
2.2 Kelas dan Ciri Penyakit Kelainan Retina 21
2.3 Segmentasi 24
2.3.1 Multilevel Thresholding Otsu 25
2.3.2. Teknik Optimasi Segmentasi 28
2.4. Enhancement 30
2.4.1. Contrast Enhancement 31
2.5. Ekstraksi Fitur 35
2.5.1. Ekstraksi Fitur Berdasarkan Tekstur 35
2.5.2. Ekstraksi Fitur Berdasarkan Bentuk 44

2.5.3. Ekstraksi Fitur Berdasarkan Warna 45
2.6. Klasifikasi Penyakit Kelainan Retina 46
2.6.1. Metode Self Organizing Map (SOM) 46
2.6.2. Metode Multi Layer Perceptrons (MLP) 53
2.6.3. Metode Back Propagation(BP) 56
2.6.4. Metode Convolutional Neural Network (CNN) 59
BAB III. METODOLOGI PENELITIAN 66
3.1. Pendahuluan 66
3.2. Dataset Citra Retina 67
3.3. Metode Segmentasi 69
3.3.1. Teknik Hybrid Multilevel Thresholding Otsu dan Harmony Search Algorithm (MTHSA) 71
3.3.2. Teknik Hybrid Multilevel Thresholding Otsu dan IHSA(MTIHSA) 74
3.3.3. Kriteria Pengukuran Peak Signal Noise Ratio (PSNR) 75
3.4. Metode Enhancement: Parsial Foreground Histogram Equalization 76
3.5. Metode Ekstraksi Fitur 78
3.5.1. Ekstraksi Fitur Retina menggunakan Tekstur 78
3.5.2. Ekstraksi Fitur Berdasarkan Bentuk 79
3.5.3. Ekstraksi Fitur Berdasarkan Warna 79
3.6. Klasifikasi Penyakit Kelainan Retina dengan CNN 80
BAB IV. HASIL DAN PEMBAHASAN 88
4.1. Pendahuluan 88
4.2. Dataset 88
4.3. Lingkungan Penelitian 90
4.4. Hasil Penelitian dan Pembahasan 91
4.4.1. Performance Hasil Pra Pengolahan dan Enhancement 92
4.4.2. Performance Hasil Segmentasi 95
4. 4.3. Performance Proses Ekstraksi Fitur 99
4.4.3.1. Performance Ekstraksi Fitur berdasarkan Tekstur 99
4.4.3.2. Performance Ekstraksi Fitur berdasarkan Bentuk 103
4.4.3.3. Performance Ekstraksi Fitur Berdasarkan Warna 104

4.4.4. Performance Hasil Klasifikasi Penyakit Kelainan Retina 107
4.4.4.1. Performance Hasil Klasifikasi Penyakit dengan SOM 107
4.4.4.2. Performance Hasil Klasifikasi Penyakit dengan MLP 111
4.4.4.2. Performance Hasil Klasifikasi Penyakit dengan BP..... 115
4.4.4.4. Performance Hasil Klasifikasi Penyakit dengan CNN 117
BAB V. KESIMPULAN 139
5.1. Kesimpulan 139
5.2. Kontribusi 140
5.3. Saran 142
DAFTAR REFERENSI 143

DAFTAR GAMBAR

Gambar 1.1. Pembuluh darah, mokula(fovea) dan pusat pembuluh (optic disk)(Al-Bander et al., 2018) 12
Gambar 1 2. Diagram Alur Kerangka Penelitian 13

Gambar 2.1 Ruang Warna HSV(Xiong et al., 2017) 45
Gambar 2.2 Jaringan grid dan neuron tetangga 48
Gambar 2.3 Jaringan SOM 1-D 48
Gambar 2.4 Jaringan SOM 2-D 48
Gambar 2.5. Diagram blok klasifikasi dengan SOM 50
Gambar 2.6. Flowchart Algoritma Self-Organizing Map 51
Gambar 2.7. Penerapan arsitektur SOM untuk klasifikasi 52
Gambar 2.8. Arsitektur MLP Sederhana 53
Gambar 2.9. Flowchart Algoritma Multi-Layer Perceptrons 55
Gambar 2.10 Arsitektur Back Propagation 59
Gambar 2.11 Arsitektur Lenet 5 61
Gambar 2.12 Peta konvolusi dan subsampel LeNet-5 61
Gambar 2.13 Ilustrasi dari proses konvolusi 62
Gambar 2.14 Pergeseran stride pada konvolusi 63
Gambar 2.15 Padding pada konvolusi 63
Gambar 2.16 Max Pooling 64

Gambar 3.1 Bagan Alur Teknik Klasifikasi Penyakit Kelainan Retina 66
Gambar 3 2. Citra Retina dengan Penyakit Kelainan Retina 68
Gambar 3.3. Blok Diagram Proses Segmentasi 70
Gambar 3 4. Diagram Alur Metode MTHSA 73
Gambar 3 5. Diagram Alur MTIHSA 75
Gambar 3 6. Diagram Alir Klasifikasi Ciri Warna 80
Gambar 3.7. Arsitektur CNN dengan menggunakan Lenet-5 81
Gambar 3.8. Visualisasi C1 82
Gambar 3.9. Visualisasi Sum Sampling (Pooling) S2 83
Gambar 3.10. Visualisasi Konvolusi C3 84
Gambar 3.11. Visualisasi Subsampling (Pooling) S4 86
Gambar 3.12. Visualisasi Konvolusi C5 87

Gambar 4. 1. Contoh Citra untuk Penyakit dan Normal a. Background Diabetic Retinopathy (BDR), b. Central Retinal Vein Occlusion (CRVO), c. Coats, d. Normal dan e. Proliferative Diabetic Retinopathy (PDR) 89
Gambar 4.2. Proses Segmentasi Citra MTIHSA 96

Gambar 4.3. Grafik Perbandingan Nilai PSNR 99
Gambar 4.4 Grafik hasil quantization error nilai iterasi dari data training 108
Gambar 4.5. Grafik quantization error nilai iterasi dari data testing 109
Gambar 4.6. Grafik Quantization Error nilai Learning Rate 111
Gambar 4.7. Grafik Perbandingan Tingkat Akurasi Hasil Pelatihan..... 112
Gambar 4.8. Perbandingan Tingkat Pengukuran Hasil Pelatihan 113
Gambar 4.9. Perbandingan Tingkat Akurasi Hasil Pengujian 114
Gambar 4.10. Perbandingan Tingkat Pengukuran Hasil Pengujian 114
Gambar 4.11 Grafik Perbandingan nilai pengukuran kinerja 115
Gambar 4.12 Grafik Akurasi dengan 100 Epoch 116
Gambar 4.13. Hasil Uji CNN 118
Gambar 4.14 Grafik Spesifikasi dari th 2, th 3, th 4, th 5 pada 5 kelas penyakit 124
Gambar 4.15 Grafik Sensitivitas dari th 2, th 3, th 4, th 5 pada 5 kelas penyakit125
Gambar 4.16 Grafik Hasil Akurasi dari th 2, th 3, th 4, th 5 pada 5 kelas penyakit 126
Gambar 4.17 Grafik Hasil Spesifikasi dari th 2, th 3, th 4, th 5 pada 5 kelas penyakit 130
Gambar 4.18 Grafik Hasil Sensitivitas dari th 2, th 3, th 4, th 5 pada 5 kelas penyakit 131
Gambar 4.19 Grafik Hasil Akurasi dari th 2, th 3, th 4, th 5 pada 5 kelas penyakit 132
Gambar 4.20 Grafik Rata-rata spesifikasi, sensitivitas, dan akurasi 133

DAFTAR TABEL

Tabel 2.1. Kelebihan dan Kelemahan beberapa Metode Segmentasi 17
Tabel 2.2. Kelebihan dan Kelemahan Beberapa Metode Klasifikasi 19
Tabel 2.3. Kelas dan Ciri-Ciri Penyakit Kelainan Retina 21

Tabel 3 1. Dataset Sekunder yang Digunakan 68
Tabel 3 2. Pembagian dataset Stare untuk data latih dan data uji 69
Tabel 3 3. Pembagian dataset Stare untuk data latih dan data uji 69
Tabel 3.4. Parameter Yang Digunakan 74
Tabel 3.5. Kuantisasi 8 variasi 78
Tabel 3.6. Fitur tekstur hasil GLCM 79
Tabel 3.7. Feature Maps Connection Antara C3 Dan S2 85

Tabel 4.1. Pengukuran kinerja 91
Tabel 4.2. Hasil Praprocessing kelas BDR, CRVO, Coats, Normal dan PDR 93
Tabel 4.3. Hasil dari proses enhancement citra retina 93
Tabel 4. 4. Hasil Pengujian data training dengan Multilevel Thresholding..... 95
Tabel 4.5. Hasil Multithresholding Improved Harmony Search pada Data Training 97
Tabel 4. 6. Hasil Multithresholding Improved Harmony Search pada Data Uji... 97
Tabel 4.7. Perbandingan nilai PSNR antara metode MT-IHSA dengan metode lainnya 99
Tabel 4.8 Hasil perhitungan rata-rata setiap fitur 100
Tabel 4.9. Hasil proses ekstraksi fitur tekstur citra retina 101
Tabel 4.10. Hasil Ekstraksi Bentuk 104
Tabel 4.11. Hasil Ekstraksi Fitur Warna 105
Tabel 4.12. Perbandingan Kinerja Ekstraksi Pembuluh Darah 107
Tabel 4.13. Data testing setelah di normalisasi 109
Tabel 4.14. Hasil Klasifikasi 5 iterasi 110
Tabel 4.15. Perbandingan Hasil Pelatihan Berdasarkan Epoch 112
Tabel 4.16. Perbandingan Hasil Pengukuran Pelatihan Berdasarkan Epoch 112
Tabel 4.17. Perbandingan Hasil Pengujian 113
Tabel 4.18. Perbandingan Hasil Pengukuran Pengujian Berdasarkan Epoch 114
Tabel 4.19. Nilai Performa klasifikasi dengan Confusion Matriks untuk th 2 ... 120
Tabel 4.20 Nilai Performa klasifikasi dengan Confusion Matriks untuk th 3..... 121
Tabel 4.21. Nilai Performa klasifikasi dengan Confusion Matriks untuk th 4 ... 122
Tabel 4.22. Nilai Performa klasifikasi dengan Confusion Matriks untuk th 5 ... 123
Tabel 4.23. Nilai Performa klasifikasi dengan Confusion Matriks untuk th 2 ... 127
Tabel 4.24. Nilai Performa klasifikasi dengan Confusion Matriks untuk th 3 ... 128
Tabel 4.25. Nilai Performa klasifikasi dengan Confusion Matriks untuk th 4 ... 129

Tabel 4.26. Nilai Performa klasifikasi dengan Confusion Matriks untuk th 5 ... 129
Tabel 4.27. Data citra penyakit retina gagal dengan Th 2 134
Tabel 4.28. Data citra penyakit retina gagal dengan Th 3 135
Tabel 4.29. Data citra penyakit retina gagal dengan Th 4 136
Tabel 4.30. Data citra penyakit retina gagal dengan Th 5 136
Tabel 4.31. Perbandingan Kinerja antara Metode yang diusulkan dengan metode lain 138

BAB I. PENDAHULUAN

1.1. Pendahuluan

Teknik klasifikasi penyakit secara konvensional memiliki kelemahan, tidak akurat dan tidak praktis(Sadikoglu & Uzelaltinbulat, 2016). Hal ini menimbulkan ide pengembangan teknologi biometrik yang berdasarkan beberapa ciri alami manusia seperti wajah, retina, iris mata dan lidah. Biometrik adalah teknologi yang menggunakan karakteristik biologis untuk mengidentifikasi individu (Figueiredo et al., 2016). Salah satu aplikasi teknologi biometrik yang menggunakan ciri retina pada bidang medis yaitu teleophthalmology. Teleophthalmology adalah cabang dari telemedicine yang memberikan perawatan mata melalui peralatan medis digital dan teknologi telekomunikasi. Saat ini, aplikasi teleophthalmology mencakup akses ke spesialis mata untuk pasien di daerah terpencil, skrining penyakit mata, diagnosa dan pemantauan, serta pembelajaran jarak jauh. Solusi teleophthalmological saat ini umumnya difokuskan pada masalah mata tertentu, seperti retinopati diabetik, retinopati prematuritas, degenerasi makula, strabismus dan penyakit mata adneksa. Kondisi yang kurang umum yang dapat diungkapkan dengan menggunakan citra retina adalah oklusi arteri dan vena, chorioretinitis, anomali kongenital, dan abitropi tapetoretinal. Beberapa studi telah menggunakan pencitraan retina untuk menghubungkan kelainan ophthalmic dengan kondisi umum, termasuk hipertensi, disfungsi ginjal, mortalitas kardiovaskular, stroke subklinis dan klinis, dan gangguan kognitif. Aplikasi mobile tersedia secara luas dalam oftalmologi dan optometri sebagai alat untuk tes mata (ketajaman visual, tes warna, dan melihat citra mata) dan untuk tujuan pendidikan. Meskipun penelitian dan kemajuan yang sedang berlangsung dalam fotografi digital, teknik pencitraan digital masih menghadapi hambatan tertentu, termasuk sensitivitas dan spesifisitas yang rendah, serta kurangnya stereopsis (kesan kedalaman).

1.2. Latar Belakang

Manusia dapat dengan mudah menafsirkan objek-objek pada suatu citra sedangkan, komputer tidak dapat melakukan hal tersebut dengan mudah. Komputer

membutuhkan perangkat lunak untuk menjalankan fungsinya dan perangkat manusia yang membangun perangkat lunak dan perangkat keras. Dari ketiga perangkat itu, perangkat manusia merupakan perangkat yang paling utama dalam pengembangan ilmu dan teknologi bidang Computer Vison (Szeliski, 2010) karena manusia dapat menjadi subjek dan objek penelitian. Salah satu peran sebagai objek penelitian, manusia dibekali dengan ciri-ciri bawaan yang khusus dibandingkan makhluk lain, yang dinamakan biometrik.

Biometrik merupakan teknologi yang popular dalam menggunakan karakteristik biologis untuk mengidentifikasi individu. Setiap orang memiliki karakteristik biometrik yang bersifat unik. Biometrik banyak digunakan dalam aplikasi forensik dikarenakan karakteristik biometrik setiap orang tidak sama dan sulit untuk ditiru. Salah satu bagian dari biometrik adalah retina merupakan salah satu karakteristik yang tidak dapat dengan mudah untuk ditiru (Madane & Thepade, 2016) (Xu, Ding, Abràmoff, & Cao, 2017). Biometrik pada dasarnya mengacu pada identifikasi dan klasifikasi otomatis seseorang berdasarkan karakteristik anatomisnya seperti wajah, sidik jari, dan retina. Lidah dan retina digunakan oleh para dokter untuk mengidentifikasi gejala awal suatu penyakit.

Retina merupakan salah satu dari karakteristik biologis yang populer. Retina merupakan bagian dari mata yang memiliki permukaan sferis (Brancati et al., 2018) yang terletak di bagian paling dalam dari bola mata dan merupakan sistem biometrik yang dapat diamati secara langsung dan non-invasif (Oliveira, Pereira, & Silva, 2018). Retina memiliki pola pembuluh darah yang unik dan berbeda dari satu orang ke orang lain, bahkan diantara kembar identik (Figueiredo et al., 2016). Pola pembuluh darah retina dpat mengidentifikasi panjang, lebar, tortuositas, pola percabangan, dan sudut (Jiang, Zhang, Wang, & Ko, 2018) untuk diagnosis dan tindak lanjut dari penyakit (Brancati et al., 2018). Kondisi pembuluh retina adalah biomarker yang dapat dipercaya dari beberapa penyakit ophthalmologic dan kardiovaskular (Oliveira et al., 2018)(Jiang et al., 2018).

Identifikasi penyakit menggunakan retina merupakan bidang pengenalan pola yang beroperasi dengan memperoleh data biometrik dari individu seseorang dan identifikasi retina merupakan suatu bagian dari biometrik yang aman untuk

dipakai dalam identifikasi identitas dalam membedakan individu dan dalam membedakan suatu penyakit (Sadikoglu & Uzelaltinbulat, 2016). Klasifikasi sangat membantu untuk mengetahui pengelompokkan suatu penyakit berdasarkan ciri-ciri persamaan dan perbedaan suatu penyakit sehingga klasifikasi dapat digunakan mendiagnosa awal berbagai penyakit.

Huang, et.al(2018) telah melakukan klasifikasi pengelompokan arteri atau vena retina menggunakan fitur pencarian genetik, pengelompokan klasifikasi arteri atau vena sangat berguna hal ini dikarenakan banyak biomarker yang berhubungan dengan arteri dan vena dalam perkembangan penyakit retina seperti arteriolar venular, dan diabetic retinopati akan tetapi pada penelitian ini masih memiliki kekurangan yaitu hasil akhir yang hanya berfokus menentukan arteri atau vena belum kepada penyakit pengklasifikasi penyakit retina berdasarkan arteri dan vena (F. Huang, Dashtbozorg, Tan, & ter Haar Romeny, 2018).

Fraz, et.al(2017) melakukan penelitian sinyal biomedik untuk segmentasi eksudat dalam citra retina menggunakan isyarat kontekstual dan klasifikasi ensemble yang mengelompokan wilayah eksudat pada butiran halus dan tingkat kasar untuk menangani variabilitas ukuran eksudaretegion sehingga dapat mengurangi kepalsuan dalam segmentasi eksudat mencapai akurasi 0.8772 akan tetapi terdapat kekurangan yaitu pada penelitian ini hanya menentukan wilayah eksudat (Fraz, Jahangir, Zahid, Hamayun, & Barman, 2017). (Dash & Bhoi, 2017) yang telah melakukan penelitian menggunakan citra retina dengan teknik komputerisasi untuk mengekstraksi pembuluh darah retina. Selain itu (Shuai, Zuo, Wang, Wang, & Member, 2017), (W. Zhang et al., 2015), (Xu et al., 2017), (Kim, Sangjun, Kim, & Lee, 2016), (Tan, Acharya, Bhandary, Chua, & Sivaprasad, 2017) juga melakukan penelitian citra retina untuk segmentasi klasifikasi jenis penyakit retina.

Tahapan proses klasifikasi penyakit berdasarkan citra retina dimulai dengan segmentasi, ekstraksi fitur dan klasifikasi. Segmentasi merupakan proses memisahkan latar depan dan latar belakang. Sedangkan ekstraksi fitur citra merupakan proses mengubah data masukkan menjadi nilai ciri suatu objek. Proses segmentasi dan ekstraksi fitur sangat berguna untuk proses klasifikasi suatu objek

agar mudah untuk dikenali. Teknik pada multi-level tresholding merupakan metode yang bekerja sangat baik dalam menentukan segmentasi dengan latar belakang yang kompleks (A. K. M. Khairuzzaman & Chaudhury, 2017).

Teknik optimasi yang mengadopsi fenomena alam antara lain Harmony Search Algorithm, Neural Network, Genetic Algoritms (GA), Differential Evolution (DE), Particle Swarm Optimization (PSO) dan Bat Algoritms (BA) (Naidu, Rajesh Kumar, & Chiranjeevi, 2017) (Banimelhem & Yahya, 2011). Penelitian yang menggunakan pendekatan Improved Harmony Search Algoritma(IHSA), antara lain (Yang, Shufan, Li, & Weifeng, 2016) (Patil & Patel, 2013) (Degertekin, 2012) (Oliva, Cuevas, Pajares, Zaldivar, & Perez-cisneros, 2013) (Guo, Yue, Liu, Wang, & Li, 2015).

Pada penelitian (Cuevas, Ortega-Sánchez, Zaldivar, & Pérez-Cisneros, 2012) HSA terbukti mampu menghasilkan akurasi lebih baik dan lebih cepat daripada Algoritma Genetika, *Randomized Hough Transform, Adaptive Bacterial Foraging*. IHSA (*Improved Harmony Search Algorithm)* adalah improvisasi dari HSA (*Harmony Search Algorithm)* pada bagian pencarian karakteristik terbaik dan laju konvergensi (Mahdavi, Fesanghary, & Damangir, 2007). (Fourie, Mills, & Green, 2010) pada penelitiannya telah membuktikan bahwa IHSA lebih cepat dan menghasilkan akurasi yang lebih tepat dalam mencari solusi optimal daripada *Particle Filter* dan *Unscented Kalman Filter.*

Tahap berikutnya setelah segmentasi adalah proses meningkatkan atau *enhancement* foreground hasil segmentasi. Teknik peningkatan citra digunakan untuk memperbaiki kontras pembuluh darah pada citra retina sehingga lebih akurat dalam ekstraski fitur. Untuk citra retina, algoritma untuk meningkatkan citra retina sebagian besar digunakan sebagai pra-pengolahan dalam mendeteksi struktur retina dan lesi seperti yang dilakukan (Geetharamani & Balasubramanian, 2016) (Xiong, Li, & Xu, 2017). Selanjutnya, yaitu tahap ekstrasksi fitur dengan mengekstraksi fitur ciri citra retina yang ingin dikenali atau dibedakan dengan objek lainnya, dimana langkah ekstraksi fitur sangat penting. Fitur yang diekstrak ini kemudian digunakan sebagai parameter atau nilai masukan untuk membedakan antara objek

satu dengan yang lainnya pada tahap klasifikasi (Geetharamani & Balasubramanian, 2016) (Al-Bander, Al-Nuaimy, Williams, & Zheng, 2018).

Ada beberapa masalah yang sering ditemukan pada segmentasi citra retina, misalnya foreground berupa pembuluh darah yang salah dalam segmentasi karena segmentasi yang kurang maupun segmentasi yang lebih atau tidak perlu. Pada sistem pengenalan pembuluh darah di retina, misalnya, adanya noise pada fitur retina yang terdeteksi. Selain itu, ada pula yang mendefinisikan citra berdasarkan bentuk, geometri, dan hubungan antara fitur pada citra. Namun sebagian besar penelitian menghadapi masalah seperti kesulitan meningkatkan akurasi dalam mengenali objek karena umumnya masalah ini berbanding terbalik dengan kecepatan pengenalan atau klasifikasi suatu objek citra. Dalam hal citra medik, masalah akurasi menjadi lebih dominan dibandingkan dengan kecepatan.

Isu-isu yang ada di atas memberi peluang penelitian di bidang ini. Sehubungan dengan itu, pertanyaan penelitian yang saling terkait yang perlu dipecahkan sebagai berikut:

1. Bagaimana cara meningkatkan akurasi pada proses segmentasi pembuluh darah, optic disk, mokula, fovea atau eksudat pada citra retina dengan menggabungkan metode multilevel thresholding dan impoved harmoni search algoritma(MT-IHSA)?
2. Bagaimana meningkatkan kualitas dan mempertajam foreground berupa pembuluh darah, exudate, fovea, mokula atau optic disk pada citra retina dengan peningkatan contrast, histogram equalization dan penajaman citra?
3. Bagaimana mengekstraksi fitur pembuluh darah, exudate, fovea, mokula atau optic disk dengan berdasarkan bentuk, tekstur dan warna untuk mendapatkan ciri-ciri yang sesuai dengan penyakit kelainan retina?
4. Bagaimana cara meningkatkan akurasi dalam mengklasifikasi penyakit kelainan retina dengan menggunakan hasil dari proses segmentasi, enhancement, dan ekstraksi fitur dengan menggunakan metode Convolution Neural Network dan dibandingkan dengan metode Selt Organizing Map (SOM), Multi Layer Perceptron (MLP) dan Back Propagation (BP)?

1.3. Rumusan Masalah

Berdasarkan latar belakang masalah dan pertanyaan penelitian, isu yang ada dapat dinyatakan sebagai berikut:

1. Secara umum, citra retina selalu bervariasi sesuai dengan ciri-ciri setiap penyakit kelainan retina dalam hal bentuk, warna, dan tekstur. Penelitian-penelitian tentang segmentasi untuk memisahkan foreground dari background menghasilkan akurasi yang rendah disebabkan perhitungan parameternya belum optimal, dan karena itu harus ditingkatkan akurasinya dengan menerapkan teknik optimasi. Dengan demikian, metode optimasi yang kuat harus dibangun terhadap parameter-parameter citra sehingga menghasilkan akurasi yang optimal dalam proses segmentasi. HSA dengan algoritma optimasi yang telah dimodifikasi, telah terbukti lebih baik daripada algoritma genetika dalam berbagai kasus (Fourie et al., 2010; Mahdavi et al., 2007; Ouaddah & Boughaci, 2016). Penelitian serupa dilakukan oleh (Dash & Bhoi, 2017) dengan menggunakan citra retina untuk mengekstraksi pembuluh darah di citra retina. Metode baru berupa improved harmony search algorithm (IHSA) diusulkan untuk meningkatkan akurasi pada proses segmentasi. Selain itu (Shuai et al., 2017), (W. Zhang et al., 2015), (Xu et al., 2017), (Kim et al., 2016), (Tan et al., 2017) juga melakukan penelitian citra retina untuk segmentasi klasifikasi jenis penyakit retina.
2. Tentu, banyak citra retina yang buram, dan terlampau terang akibat pencahayaan atau intensitas citra yang tidak merata akibat sensor kamera yang digunakan. Untuk meningkatkan kualitas citra tanpa menghilangkan informasi yang ada secara signifikan secara akurat memerlukan proses enhancement sehingga tidak merusak citra dan menghilangkan informasi yang ada.
3. Kompleksitas background, resolusi citra, dan cahaya memberikan tingkat akurasi yang bervariasi tergantung pada berbagai aspek tersebut. Ini menjadi salah satu alasan mengapa ekstraksi fitur masih menjadi topik penelitian yang menarik. Algoritma yang handal dan bertahan dalam

berbagai kondisi tentu diperlukan untuk menghasilkan kinerja yang lebih baik daripada algoritma yang telah diterapkan pada penelitian sebelumnya. Berdasarkan penyakit kelainan retina terjadi ciri khas yaitu terjadi perubahan bentuk, tekstur dan warna dari kondisi yang normal sehingga ekstraksi fitur berdasarkan ciri-ciri tersebut sangat layak untuk diusulkan dalam penelitian ini.

4. Kemiripan ciri yang tinggi antar kelas penyakit merupakan masalah utama dalam proses klasifikasi penyakit, misalnya banyak algoritma digunakan untuk mengatasi masalah dalam mengklasifikasi penyakit, termasuk Neural Network (Kasar, Bhattacharyya, & Kim, 2016), *Convolutional Neural Networks(CNN)* (Wen, Zhang, Li, & Qiao, 2016), dan *Eigenface*. Untuk meningkatkan akurasi ini, sebuah teknik baru diusulkan dengan menggabungkan dua metode yaitu CNN dan IHSA pada proses pengenalan pembuluh darah, exudate, fovea, mokula dan optic disk di citra retina untuk mengklasifikasi penyakit.

1.4. Tujuan Umum Penelitian

Untuk mengembangkan metode optimasi pada klasifikasi penyakit dengan citra retina berdasarkan bentuk, warna dan tekstur yang mencakup segmentasi foreground dari background, enhancement, ekstraksi fitur dan pengenalan pembuluh darah, exudate, fovea, mokula dan optic disk di citra retina.

1.5. Tujuan Khusus

Untuk memenuhi tujuan di atas, tujuan berikut ditargetkan:

1. Untuk menghasilkan teknik baru secara akurat pada proses segmentasi citra sebagai pendekatan baru yang memanfaatkan pengembangan metode HSA menjadi IHSA secara akurat dan optimal. Teknik ini harus bisa mensegmentasikan pada citra retina dari back ground secara optimal. Selain itu, harus mampu mencari kombinasi parameter segmentasi yang paling optimal.

2. Untuk menghasikan pendekatan yang baru untuk meningkatkan kualitas citra dari citra buram, terlampau terang, dan intensitas yang tidak merata yang ditemukan dalam citra retina. Selain itu, harus mampu memperbaiki kualitas foreground citra.
3. Untuk mengusulkan teknik ekstraksi fitur pembuluh darah, exudate, fovea, mokula dan optic disk berdasarkan ciri-ciri penyakit berupa bentuk, tekstur dan warna.
4. Untuk mengusulkan metode klasifikasi penyakit berdasarkan citra retina yang baru dengan menggunakan gabungan antara metode CNN dan IHSA dan hasil proses segmentasi, enhancement dan ekstraksi fitur.

1.6. Batasan Masalah

Penelitian ini melibatkan proses lengkap dalam pengolahan citra retina termasuk segmentasi citra, enhancement, ekstraksi fitur dan pengenalan objek. Retina menurut dokter mata dapat menjadi alat diagnosis awal untuk mengetahui gejala penyakit hipertensi(tekanan darah tinggi), stroke, gejala tiroid, anemia, kolesterol tinggi, penyakit hati, tumor dan diabetes. Penyakit diabetes menjadi pilihan dalam klasifikasi citra dikarenakan 2(dua) alasan utama yaitu ciri-ciri penyakit sesuai dengan ekstraksi fitur dalam bentuk, tekstur dan warna, dan jumlah penderita diabetes di Indonesia menempati peringkat 6(enam) dunia menurut data International Diabetes Federation (IDF) Atlas 2017 dan Riset Kesehatan Dasar (Riskesdas) memperlihatkan peningkatan angka prevalensi Diabetes yang cukup signifikan, yaitu dari 6,9% di tahun 2013 menjadi 8,5% di tahun 2018; sehingga estimasi jumlah penderita di Indonesia mencapai lebih dari 16 juta orang yang kemudian berisiko terkena penyakit lain, seperti: serangan jantung, stroke, kebutaan dan gagal ginjal bahkan dapat menyebabkan kelumpuhan dan kematian.

Selain itu, seperti untuk platform pengujian dataset standar, penelitian ini menggunakan dua dataset sekunder yang diperoleh yaitu:

1. Dataset pertama yaitu DRIVE (Digital Retinal Images for Vessel Extraction) merupakan dataset segmentasi pembuluh darah di citra retina. Dataset ini digunakan pada proses segmentasi dan pengenalan objek.
2. Dataset ke dua yaitu, STARE (Structured Analysis of the Retina) merupakan hasil pengujian secara klinik yaitu mempelajari pembuluh darah di retina dan diagnosa penyakit pasien. Dataset yang tersedia terdiri dari 400 citra retina, daftar kode diagnosa dan diagnosa penyakit masing-masing citra dan hasil segmentasi pembuluh darah. Dataset citra retina digunakan pada proses pengenalan objek.

Selanjutnya, penelitian ini menggunakan performa akurasi sebagai indikator pengujian model optimasi sehingga masalah kecepatan, dan pemakaian ruang penyimpanan serta lainnya diluar kajian penelitian ini.

1.7. Manfaat dan Kontribusi Penelitian

Isu dan kelemahan hasil penelitian sebelumnya masih ada meski banyak upaya untuk memperbaiki masalah. Hal ini mendorong penulis untuk memulai sebuah penelitian untuk mengatasi kekurangan dan kelemahannya. Meskipun, beberapa penelitian telah mengklaim bahwa metode pengolahan citra yang diusulkan telah berhasil tetapi metode tersebut dapat dikaji lebih lanjut karena menggunakan dataset yang digunakan relatif kecil dan tidak bervariasi sehingga upaya peningkatan akurasi pengolahan citra dengan menggunakan beberapa sumber dataset dan teknik optimasi layak dikembangkan untuk aplikasi kehidupan nyata.

Dengan latar belakang masalah di atas, penelitian ini sangat penting untuk menyelesaikan semua kekurangan dan kelemahan termasuk melakukan optimasi semua proses dan menggunakan dataset standar dengan sampel yang memadai dan tidak bias. Meskipun demikian, signifikansi penelitian ini tidak hanya terbatas pada pengayaan pengetahuan, namun juga dapat diterapkan pada aplikasi kehidupan

nyata, misalnya media diagnosa dini penyakit, sistem deteksi penyakit, dan pengenalan individu pada sistem kehadiran, forensik, dan imigrasi.

Hasil yang diharapkan pada penelitian ini memberikan empat kelompok temuan kontribusi penelitian untuk memodelkan pengolahan citra secara optimasi yaitu, segmentasi, enhancement, ekstraksi fitur dan klasifikasi penyakit.

1. Segmentasi

Metode segmentasi baru diusulkan dengan menggunakan hybrid metode multilevel thresholding Otsu dan improved harmonic search algorithm (IHSA). Teknik ini dinamakan MTIHSA (Multilevel Threshoding-Improved Harmonic Search Algorithm). Metode Otsu yang diterapkan dalam teknik multilevel thresholding karena lebih baik dibandingkan Kapur.

2. Enhancement

Setelah proses segmentasi yang memisahkan antara foreground dan background diperoleh, dilanjutkan dengan proses enhancement. Proses enhancement baru diusulkan dalam penelitian ini dengan melakukan penyesuaian nilai intensitas dilakukan secara otomatis dengan menggunakan pemerataan histogram. Teknik ini menghasilkan histogram yang seragam atau merata. Teknik ini dilakukan secara parsial untuk foreground yang menjadi fokus ciri suatu kelas penyakit yang dinamakan Parsial Foreground Histogram Equalization.

3. Ekstraksi Fitur

Metode baru untuk mengekstraksi fitur secara lebih sesuai dengan ciri penyakit diusulkan, yaitu menggunakan ekstraksi bentuk, tekstur dan warna. Gabungan ketiga teknik ekstraksi fitur ini menghasilkan nilai masukan pada proses klasifikasi menjadi lebih akurat karena pendekatan ekstraksi fitur sudah sesuai dengan ciri-ciri penyakit.

4. Klasifikasi Penyakit

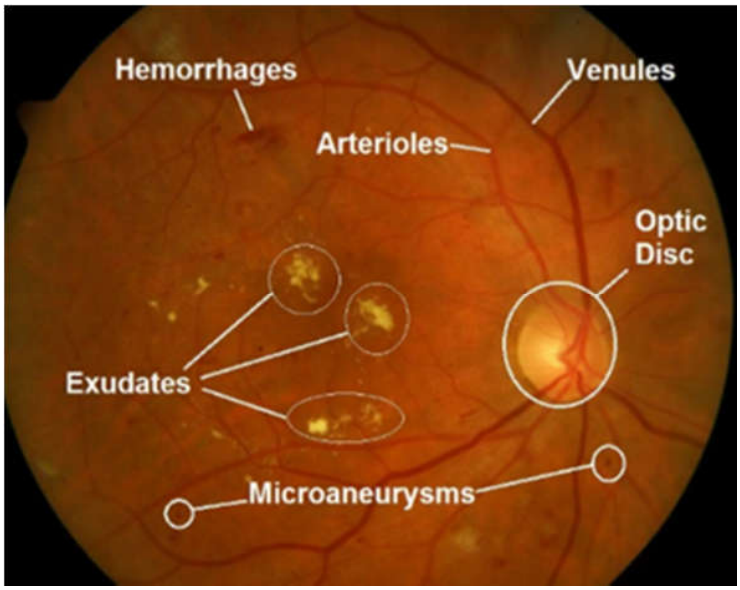
Modifikasi inisialisasi nilai vektor input pada proses klasifikasi dengan metode Convolution Neural Network(CNN) diusulkan, yaitu Modifikasi Layer

Input CNN. Pengujian klasifikasi penyakit juga dilakukan dengan menggunakan metode Self Organizing Map(SOM), Multi Layer Perceptron(MLP), dan Back Propagation(BP). Penggunaan ke-4 metode ini digunakan untuk membandingkan performa metode-metode tersebut dalam klasifikasi penyakit. Alasan menggunakan metode SOM, MLP dan BP yaitu supaya pengujian metode dilakukan secara konsisten dengan menggunakan metode metode yang sejenis pendekatannya berupa metode terbimbing (*supervised methods*).

1.8. Kerangka Penelitian

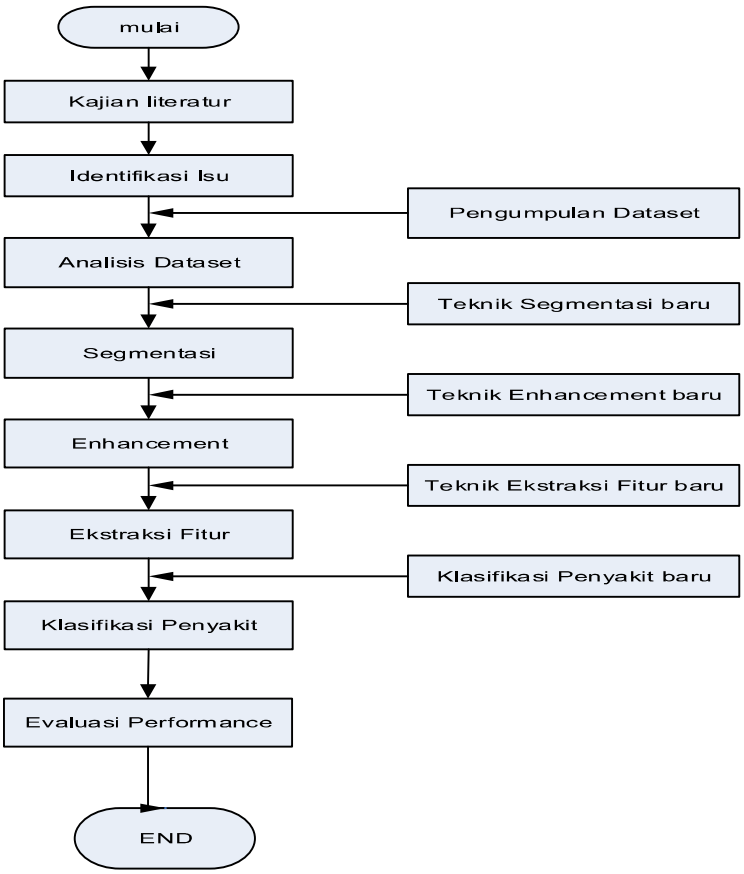
Secara garis besar bagan alur penelitian untuk membangun model optimasi teknik klasifikasi dengan harmony search algorithm dimulai dengan melakukan studi literatur untuk mendapatkan state of the art lingkup penelitian terkini, metode terkini yang telah diajukan dan mempelajari hasil penelitian yang telah diperoleh dari penelitian-penelitian sebelumnya. Selanjutnya, dilakukan pengumpulan dan menganalisis dataset retina yang standar yaitu STARE dan DRIVE.

Pada kasus deteksi penyakit, bagian yang diamati oleh para ahli Ophthamologis yaitu pembuluh darah, fovea dan optic disk(Gambar 1.1). Sebagai besar masalah yang muncul diantaranya kesulitan mendeteksi pembuluh-pembuluh yang ukurannya sangat kecil, titik-titik merah kecil di retina, pendarahan pada pembuluh dan pembengkakan pembuluh darah yang berukuran mikro.



Gambar 1.1. Pembuluh darah, mokula(fovea) dan pusat pembuluh (optic disk)(Al-Bander et al., 2018)

Penelitian ini merupakan penelitian eksperimental berbasis seperti yang digambarkan pada Gambar 1.2 di bawah ini. Penelitian dimulai dengan melakukan kajian literatur untuk menemukan isu-isu permasalahan dan kelemahan-kelemahan dari metode yang ada dan juga untuk mempelajari hasil-hasil penelitian pada. Kemudian, diikuti dengan mengumpulkan dan menganalisis dataset sekunder yaitu STARE dan DRIVE. Dalam kasus ini, rincian singkat dari citra dipelajari untuk merancang pendekatan yang sesuai yang dapat diterapkan untuk memecahkan masalah yang terkait dengan kualitas citra.



Gambar 1 2. Diagram Alur Kerangka Penelitian

1.9. Sistematika Disertasi

Sistematika penulisan disertasi terdiri dari lima bab, yaitu Pendahuluan, Studi Pustaka, Metodologi, Hasil Penelitian dan Kesimpulan. Bagian Pendahuluan menjelaskan tentang latar belakang, rumusan masalah, tujuan, manfaat dan kontribusi penelitian. Bab Studi Pustaka ditempatkan pada bab dua yang membahas

hasil penelitian sebelumnya yang berkaitan dengan penelitian ini seperti mengkaji tentang segmentasi, enhancement, ekstraksi fitur dan klasifikasi penyakit berdasarkan citra retina.

Bab tiga membahas tentang teknik klasifikasi penyakit berdasarkan citra retina dengan pendekatan optimasi terdiri dari pendekatan optimasi untuk segmentasi, enhancement, ekstraksi fitur dan klasifikasi penyakit.

Bab empat menyajikan tentang dataset yang digunakan, lingkungan percobaan penelitian, evaluasi dan pembahasan hasil untuk setiap tahap segmentasi, enhancement, ekstraksi fitur dan klasifikasi.

Akhirnya, bab kesimpulan menyajikan hasil yang diperoleh dengan menghubungkan antara tujuan, masalah, metode yang diusulkan dan memberikan saran untuk penelitian selanjutnya.

BAB II. STUDI PUSTAKA

2.1 Pendahuluan

Ilmu biometrik yang digunakan untuk pendeteksian menggunakan ciri alami manusia berupa wajah, sidik jari, mata, telinga dan lidah. Struktur sidik jari dan mata merupakan ciri alami manusia yang bersifat unik. Iris mata merupakan bagian mata yang paling sering digunakan. Teknik untuk mendeteksi wajah dapat dilakukan berdasarkan karakteristik dari wajah, model wajah, dan rupa dari wajah (Szeliski, 2010).

(Geem, Kim, & Loganathan, 2001) pertama kali memperkenalkan algoritma metaheuristic baru yang terinspirasi oleh para musisi yang sedang mencari harmoni terbaik dengan terus memainkan nada berdasarkan pengalaman atau secara acak hingga mendapatkan musik yang indah. Algoritma ini adalah HSA (*Harmony Search Algorithm).* Permasalahan dalam optimasi dapat diselesaikan menggunakan HSA. Sama halnya dalam pencarian harmoni terbaik, solusi optimal dari permasalahan dicari berdasarkan aturan dan fungsi yang telah ditentukan.

Penelitian yang membahas penerapan IHSA (*Improved Harmony Search Algorithm)* untuk menyelesaikan masalah optimisasi pernah dilakukan oleh (Mahdavi et al., 2007). Hasil dari penelitian tersebut membuktikan bahwa IHSA memberikan solusi lebih baik HSA *(Harmony Search Algorithm)* dan Algoritma Genetika. (Fourie et al., 2010) pada penelitiannya mengenai *tracking system* juga membuktikan bahwa IHSA lebih baik dibandingkan dengan *Particle Filter* dan *Unscented Kalman Filter* dari segi akurasi maupun waktu komputasi. IHSA juga digunakan oleh (Karthigeyan et al., 2015) dalam penelitiannya tentang perbandingan antara IHSA, HSA, dan BBO *(Biogeography Based Optimisation).* Hasil dari penelitian tersebut juga membuktikan kemampuan IHSA lebih baik dari HS dan algoritma BBO dalam memberikan solusi terbaik berupa biaya pengeluaran yang minimum.

Retina adalah sistem biometrik yang paling stabil dan handal karena biometrik bersifat unik dan stabil (BY Rishav Kansal, Kiran Turaka, & Carol L. Shields, 2011). Retina memiliki kelebihan dibandingkan biometrik yang lainnya yaitu memilki sifat sifat yang unik seperti pola pembuluh darah. Sehingga pada penelitian (X. Wang, Jiang, & Ren, 2019) segmentasi pembuluh darah dari citra fundus oleh kerangka klasifikasi kaskade yang menggunakan pengklasifikasi jarak mahalanobis, dan pada penelitannya mengklasifikasi pembuluh darah retina dengan memutuskan apakah pixel termasuk pada pembuluh retina atau tidak yang mencapai akurasi 95.41% tetapi terdapat kekurangan yaitu dengan pengklasifikasian pembuluh darah ini belum dapat mengklasifikasikan penyakit yang berhubungan dengan penyakit penyakit retina yang diakibatkan oleh gangguan pembuluh darah.

(Fraz, Jahangir, Zahid, Hamayun, & Barman, 2017) melakukan penelitian terkait segmentasi eksudat dalam citra retina menggunakan isyarat kontekstual dan tertanam. Mereka melakukan penelitian tersebut untuk mendeteksi eksudat secara otomatis dalam skrining Diabetik Renopati (DR). Hasil yang didapatkan adalah berhasil menangani citra dengan iluminasi latar belakang yang tidak rata, variabilitas kontras pada perolehan citra, pigmentasi latar belakang dan presensi artefak yang berbeda.

Selanjutnya (Barkana, Saricicek, & Yildirim, 2017) melakukan penelitian mengenai pengukuran pembuluh darah retina secara otomatis untuk mengetahui adanya suatu masalah dalam retina mata manusia. Metode yang digunakan untuk melakukan penelitian tersebut adalah Continuous Wavelet Transform (CWT) dan Cumulative Density Function (CDF). Metode yang diusulkan mendapatkan sensitivitas dan akurasi sebesar 80,60% dan 95,91%.

(Franklin & Rajan, 2014) melakukan deteksi DR menggunakan segmentasi pembuluh darah pada citra retina. Metode yang digunakan adalah Multilayer Perceptron (MLP) neural network dan artificial neural network (ANN). Hasil yang didapatkan memberikan arahan tentang lokasi pembuluh darah.

(Morales, Naranjo, Angulo, Legaz-Aparicio, & Verdú-Monedero, 2017) melakukan penelitian karakterisasi jaringan retina melalui pemrosesan citra fundus

dan identifikasi centerlineres pembuluh darah, penelitian ini menjelaskan pendekatan baru untuk identifikasi titik signifikan pada garis tengah pembuluh darah retina, dan penelitian ini menggunakan transformasi hit-or-miss yaitu digunakan untuk mendeteksi terminal, bifurkasi dan titik persimpangan sederhana, tetapi tahap pasca-pemrosesan diperlukan untuk mengidentifikasi persimpangan yang kompleks.

Identifikasi penyakit retina salah satunya yaitu identifikasi diabetes retinopati, yang sama halnnya dilakukan oleh (Huang, Ma, JW van Triest, Wei, & Qian, 2017) yang melakukan penelitian pada otomasis deteksi neovaskularisasi citra retina menggunakan ekstrim machine leraning. Pada penelitiannya, digunakan klasifikasi penyakit diabetes retinopaiti (DR) dan diabetic retinopathy non-proliferatif (NPDR). Tetapi terdapat kekurangan pada deteksi penyakit khususnya pada neovaskularisasi yang masih sulit untuk menganalisis struktur daerah neovaskularisasi dan hasil dari preprocessing terdapat beberapa pencahayaan yang tidak merata.

Berdasarkan dari beberapa penelitian terkait segmentasi retina, dapat dilihat pada Tabel 2.1 :

Tabel 2.1. Kelebihan dan Kelemahan beberapa Metode Segmentasi

Nama	Metode	Kelebihan	Kekurangan
(Fraz et al., 2017)	isyarat kontekstual dan tertanam	Berhasil melakukan segmentasi eksudat secara otomatis	Hasil dari metode tidak terdapat persentase keberhasilan sehingga belum bisa di pastikan apakah benar bisa mendeteksi eksudat secara otomatis.
(Barkana et al., 2017)	Continuous Wavelet Transform (CWT) dan Cumulative Density Function (CDF)	Hasil yang didapatkan termaksud baik karena persentase keberhasilan 80,60% dan 95,91%.	Hasil yang didapatkan dengan menggunakan metode CWT belum

			mencapai hasil yang maksimal.
(Franklin & Rajan, 2014)	Multilayer perceptron neural network dan artificial neural network (ANN)	Hasil yang didapatkan memberikan arahan tentang lokasi pembuluh darah.	Hasil yang di dapatkan tidak memberikan persentase keberhasilan, sehingga masih diragukan apakah benar bisa memberikan arahan tentang lokasi pembuluh darah atau tidak.
(Panchal, Bhojani, & Panchal, 2016)	Ektraksi fitur dengan Metode Hybrid dan Scanning Window Analysis (SWA)	berhasil melakukan ektraksi fitur dari citra fundus	metodologi yang digunakan masih belum sempurna

Selain melakukan segmentasi, beberapa ilmuwan juga melakukan penelitian terhadap ekstraksi fitur citra retina mata. Salah satunya (Panchal et al., 2016) melakukan penelitian terkait ekstraksi fitur dengan Metode Hybrid dan Scanning Window Analysis (SWA). Hasil yang didapatkan pendekatan ini berhasil melakukan ekstraksi fitur dari citra fundus. Untuk algoritma yang telah diusulkan memberikan keakuratan keseluruhan sistem 92,70%. Kelemahannya adalah metodologi yang digunakan masih belum sempurna sehingga masih memerlukan perbaikan.

Selain melakukan ekstraksi fitur, beberapa ilmuwan juga melakukan penelitian terkait klasifikasi. Salah satunya (Welikala et al., 2017) melakukan klasifikasi terkait pembuluh darah arteri dan vena untuk melakukan klasifikasi secara otomatis. Tujuan dilakukan penelitian ini adalah untuk mendeteksi pembuluh darah arteri dan vena secara otomatis. Metode yang digunakan menggunakan yaitu Deep Convolutional Neural Network. Hasil yang didapatkan

menghasilkan keakuratan 86,97%(per/piksel basis) untuk klasifikasi arteri dan vena pada keseluruhan citra. Kinerja yang meningkat sebesar 87,10%.

Selain itu (Xu et al., 2017) melakukan klasifikasi arteri untuk mendiagnostik gejala awal penyakit retina. Tujuannya untuk mendiagnosis berbagai penyakit melalui retina mata menggunakan bantuan komputer. Metode yang digunakan yaitu intra-image regularisasi dan normalisasi antar-subjek. Metode yang diusulkan mencapai akurasi sekitar 0,923 atau sekitar 92,30%.

(Condurache & Mertins, 2012) melakukan segmentasi pembuluh darah retina dengan melakukan paradigma klasifikasi biner hysteresis. Bertujuan untuk mendesain klasifikasi pendekatan baru untuk klasifikasi biner dengan melakukan segmentasi pembuluh dalam citra retina. Metode yang digunakan adalah hysteresis-classifier dan Binary classification. Hasil dari metode tersebut berhasil mengelompokkan objek dari representasi tingkat intensitas. Sedangkan untuk paradigmanya berhasil menghubungkan struktur retina.

Berdasarkan penelitian terkait klasifikasi dapat didapatkan perbandingan yang terdapat dalam Tabel 2.2:

Tabel 2.2. Kelebihan dan Kelemahan Beberapa Metode Klasifikasi

Peneliti	Kelebihan	Kekurangan
(Welikala et al., 2017)	Klasifikasi yang didapatkan dengan metode deep learning mendapatkan keakuratan 86,97%(per/piksel basis).	Saat evaluasi ROI kenaikkan hanya sedikit yang mengidentifikasikan kinerja stabil di dalam retina.
(Xu et al., 2017)	Klasifikasi untuk diagnotik penyakit pada retina mendapatkan akurasi sekitar 0,923 atau sekitar 92,30%.	Pendekatan kami dikembangkan dan diuji pada sekumpulan citra retina beresolusi tinggi dan kinerjanya pada citra retina berkualitas rendah belum dinilai.

(Condurache & Mertins, 2012)	Berhasil mengelompokkan objek dari representasi tingkat intensitas.	Metode yang digunakan masih belum terlalu umum dan belum banyak melakukan penelitian terkait metode tersebut.
(Banerjee & Chowdhury, 2015	dapat mengidentifikasi penyakit kelainan retina yaitu Retinopati Diabetik (DR), lalu Mascular Degeneration (AMD).	Tetapi masih terdapat kekurangan yaitu hasil yang dilakukan tidak ditulis persentase keberhasilan dan masih sedikit input yang digunakan
(Morales et al., 2017)	indentifikasi poin signifikan pada centerlineres pembuluh darah, penelitian ini menjelaskan pendekatan baru untuk identifikasi titik signifikan pada garis tengah pembuluh darah retina	tetapi tahap pasca-pemrosesan diperlukan untuk mengidentifikasi persimpangan yang kompleks
(Huang et al., 2017)	klasifikasi penyakit diabetes retinopaiti (DR) dan diabetic retinopathy non-proliferatif (NPDR) sebagai diagnosis penyakit	tetapi terdapat kekurangan pada deteksi penyakit khususnya pada neovaskularisasi yang masih sulit untuk menganalisis struktur daerah neovaskularisasi dan hasil dari preprocessing terdapat beberapa pencahayaan yang tidak merata.

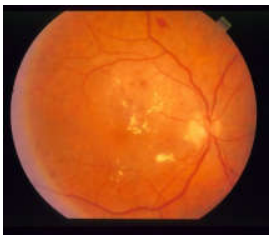
Penelitian terkait metode Convolutional Neural Network pernah dilakukan. Salah satunya (Pratt, Coenen, Broadbent, Harding, & Zheng, 2016) melakukan diagnosis terhadap peyakit diabetes dari citra fundus digital dan mengklasifikasi tingkat keparahan penyakit secara akurat. Metode yang digunakan yaitu convolutional neural network dengan hasil mencapai sensitivitas 95% dan akurasi 75% pada 5.000 citra. Kekurangan dari penelitian ini adalah dalam membuat jaringan untuk membedakan antara yang ringan, sedang dan berat.

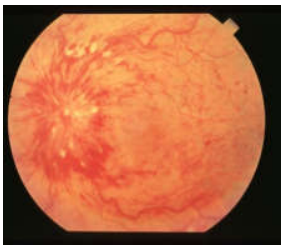
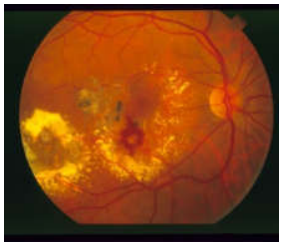
(Triwijoyo, Budiharto, & Abdurachman, 2017) melakukan penelitian terkait CNN. Penelitian yang dilakukan yaitu membuat sistem deteksi dini penyakit retinopati hipertensi(HR) untuk menentukan apakah ada gejala HR atau tidak. Kekurangannya peneliti tidak lakukan analisis efektivitas dan keakuratan penggunaan fitur utama penyakit HR seperti pembuluh retina, disk optik untuk klasifikasi gejala retinopati hipertensi.

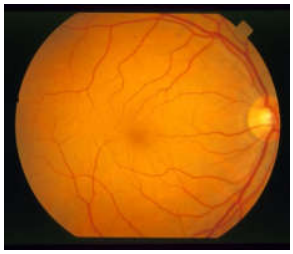
2.2 Kelas dan Ciri Penyakit Kelainan Retina

Di dalam retina terdapat berbagai macam penyakit yang dapat membahayakan mata manusia. Pada penelitian ini, ada lima kelas terdiri dari empat penyakit retina dan kelas normal. Tabel 2.3 di bawah ini merupakan jenis penyakit retina mata dan ciri penyakit retina mata (Nayak & Kaur, 2015)(Ch'ng, 2015)(Cerbulescu & Stefan, 2003):

Tabel 2.3. Kelas dan Ciri-Ciri Penyakit Kelainan Retina

No	Retina dari STARE	Jenis penyakit	Ciri-ciri
1		Background Diabetic Retinopathy	• Mikroaneurisma: Tanda klinis awal retinopati diabetik; ini terjadi sekunder untuk outpouching kapiler dinding karena kehilangan pericyte; mereka muncul sebagai titik merah kecil di lapisan retina dangkal • Dot and blot hemorrhages: Muncul mirip dengan mikroaneurisma jika berukuran kecil; mereka terjadi saat pecah mikroaneurisma di lapisan retina yang lebih dalam, seperti lapisan plexiformis di dalam inti dan luar • Perdarahan berbentuk nyala: Splinter hemorrhages yang terjadi pada lapisan serat saraf yang lebih dangkal • Edema retina dan eksudat keras: Disebabkan oleh kerusakan penghalang retina darah, memungkinkan kebocoran protein serum, lipid, dan protein dari pembuluh darah.

			• Titik kapas-wol: infark lapisan serat saraf dari oklusi arteriol precapillary; mereka sering dibatasi oleh mikroaneurisma dan hiperpermeabilitas vaskular • Loop vena dan anulus vena: Sering terjadi berdekatan dengan area nonperfusi; mereka mencerminkan peningkatan iskemia retina, dan kejadian mereka adalah prediktor yang paling signifikan dari perkembangan ke retinopati diabetik proliferatif (PDR). • Kelainan mikrovaskular intraretinal: Tempat tidur kapiler yang direnovasi tanpa perubahan proliferatif; biasanya dapat ditemukan di perbatasan retina yang tidak digunakan • Edema makula: Penyebab utama gangguan penglihatan pada pasien dengan diabetes
2		Central Retinal Vein Occlusion	• Trombosis terbentuk di vena dan menyebabkan kebocoran darah dan cairan berlebih ke retina • Sumbatan terjadi ketika pembuluh darah di mata terlalu sempit • Hemorrhage retina berbentuk bintik, dot atau punctate atau keduanya di keempat kuadran retina, vena retina melebar dan berliku, dan pembengkakan diskus optik • Cairan yang terakumulasi di area makula retina menyebabkan pembengkakan atau edema makula
3		Coats	• Tanda-tanda awal yang terdiri dari area kekuningan besar di dalam dan di bawah retina • Strabismus, atau mata yang bersilangan • Leukocoria, massa putih di belakang lensa mata • Kehilangan persepsi kedalaman • Pembuluh darah telabgiectatic melebar di retina perifer • Biasanya hanya terjadi pada satu mata dan mempengaruhi penglihatan sentral atau penglihatan tepi

			• Perdarahan subretinal jika terjadi pelepasan retina • Dapat menyebabkan strabismus (menyipitkan mata)
4		Proliferative Diabetic Retinopathy	• Pendarahan ke dalam substansi seperti jelly di tengah mata Anda (humor vitreous) ini disebut perdarahan vitreous • Pertumbuhan pembuluh darah abnormal kecil (proses yang disebut neovascularisation) di mata, dan pertumbuhan fibrosa di retina (daerah penginderaan cahaya mata) • Pelepasan retina (dimana retina ditarik menjauh dari tempat biasanya melekat pada bagian dalam mata), yang dapat terjadi jika jaringan parut menarik retina.
5		Normal	• Choroid: Lapisan mata belakang retina; mengandung pembuluh darah yang menyehatkan retina. • Kerucut: Sel-sel saraf fotoreseptor hadir di makula dan terkonsentrasi di fovea (pusat makula); memungkinkan orang untuk melihat detail dan warna yang bagus. • Fovea: Lubang atau depresi di pusat macula yang memberikan ketajaman visual terbesar. • Makula: Bagian mata di pusat retina yang memproses penglihatan yang tajam, jernih, lurus. • Photoreceptors: Sel-sel saraf sensing cahaya (batang dan kerucut) yang terletak di retina. • Retina: Lapisan sensitif cahaya jaringan yang melapisi bagian belakang mata. • Retinal Pigmented Epithelium (RPE): Lapisan sel yang melindungi dan menyuburkan retina, menghilangkan produk limbah, mencegah pertumbuhan pembuluh darah baru ke lapisan retina dan menyerap cahaya yang tidak diserap oleh sel-sel fotoreseptor; tindakan-tindakan ini

			mencegah penyebaran cahaya dan meningkatkan kejelasan visi. • Batang: Sel-sel saraf thotoreceptor di mata yang sensitif terhadap tingkat cahaya rendah dan hadir di retina, tetapi di luar makula.

2.3 Segmentasi

Segmentasi citra merupakan salah satu masalah utama dalam pengolahan citra. Segmentasi bertugas untuk memisahkan antara background dan foreground. Segmentasi mengelompokkan pixel dalam citra menjadi daerah homogen dengan banyak karakteristik (Guo et al., 2015) (Nasr-Esfahani et al., 2018). Pada penelitian ini, penulis menggunakan segmentasi untuk melakukan klasifikasi diagnosis penyakit citra pada retina. Segmentasi sendiri telah menjadi prosedur diagnosis untuk mendeteksi pembuluh darah retina secara otomatis (Condurache & Mertins, 2012). Tujuan utama dalam segmentasi ini adalah untuk menyederhanakan citra sehingga dapat diproses dan dianalisis pada poses selanjutnya.

Beberapa kendala yang ditemui selama proses segmentasi citra yakni lamanya waktu pemrosesan pada saat pengelompokan piksel dan mengakibatkan timbulnya noise pada citra. Penelitian oleh (Shang, Yuan, Jiao, Member, & Hou, 2017) untuk segmentasi citra dimana metode ini dapat mengelompokan piksel secara cepat menggunakan fuzzy c-means berdasarkan piksel kunci. Metode ini mulanya menetapkan piksel kunci yang nantinya hanya dilakukan segmentasi pada piksel kunci. Kemudian, piksel yang tidak menjadi kunci dapat dengan cepat tersegmentasi dengan menggabungkan hasil pengelompokkan piksel kunci dengan aturan kesamaan matrik pada citra. Hasil penelitian ini menunjukan bahwa algoritma ini lebih baik dalam hal kecepatan komputasi maupun pemrosesan pada citra yang memiliki noise.

Selain itu, segmentasi citra terbukti diperlukan dalam bidang medis khususnya dalam proses diagnosis penyakit. Contohnya terdapat pada klasifikasi tumor untuk mengidentifikasi osteosarcoma (Arunachalam, Mishra, Armaselu, &

Martinez, 2017). Segmentasi citra pada penelitian tersebut diawali dengan mengevaluasi citra oleh ahli patologis untuk memperkirakan persentase nekrosis, kemudian citra tersebut dipindai dalam aperio scanscope, lalu di konversi menjadi WSIs dan disimpan dalam format SVS. Proses segmentasi dimulai dengan menklasifikasi daerah tumor berdasarkan sifat tumor, gugus inti, kepadatan, dan sirkulasi. Metode K-Means digunakan untuk melakukan pengelompokan tumor menggunakan klasifikasi tumor. Selanjutnya algoritma Flood Fill digunakan untuk mengelompokan piksel yang serupa ke objek sehingga data cluster dapat dihitung dan pada akhirnya dianalisis. Hasil dari penelitian ini menyebutkan bahwa identifikasi tumor sangat meyakinkan, dimana nilai akurasi sebesar 100% dan identifikasi necrosis sekitar 90%.

2.3.1 Multilevel Thresholding Otsu

Metode Thresholding adalah salah satu metode segmentasi citra yang paling banyak digunakan (Zhou & Zhou, 2016)(Zheng, Ye, & Tang, 2017). Thresholding adalah salah satu metode yang efektif dan sederhana untuk memisahkan objek dan latar belakang. Citra tersegmentasi menjadi dua kelas, disebut thresholding bilevel, sedangkan jika tersegmentasi di lebih dari dua kelas, ini disebut multilevel thresholding. Otsu dan Kapur adalah dua metode klasik untuk thresholding.

Thresholding dikenal sebagai operasi non linier yang penting dalam segmentasi citra (Holambe & Kumbhar, 2016)(Vala & Baxi, 2013). Ide dasar thresholding adalah memilih nilai threshold optimum pada tingkat keabuan untuk memisahkan objek dan latar belakang berdasarkan distribusi tingkat keabuan (Otsu, 1979a).

Ada dua jenis thresholding, global dan lokal. Otsu adalah thresholding global yang diperkenalkan oleh Otsu pada tahun 1979 (Vala & Baxi, 2013). Otsu menggunakan variabel nilai maksimum dari perbedaan kelas sebagai kriteria segmentasi citra. Dengan mengambil tingkat intensitas (L) dari citra grayscale atau

RGB, nilai distribusi peluang intensitas citra dapat dihitung menggunakan persamaan (2.1) (Oliva et al., 2013):

$$Ph_i^c = \frac{h_i^c}{NP}, \sum_{i=1}^{NP} Ph_i^c = 1, \quad (2.1)$$

$$c = \begin{cases} 1,2,3, & \text{untuk citra RGB,} \\ 1, & \text{untuk citra grayscale,} \end{cases}$$

dengan:

i = tingkat intensitas ($0 \leq i \leq L-1$)

c = komponen citra pada grayscale atau RGB

NP = total pixel pada citra

h_i^c = nilai histogram (total pixel pada c dan tingkat intensitas i

Ph_i^c = distribusi peluang pada komponen citra c dan tingkat intensitas i

Segmentasi yang paling sederhana bi-level didefinisikan dengan persamaan (2.2) berikut:

$$C_1 = \frac{Ph_1^c}{\omega_{0(th)}^c}, \ldots \frac{Ph_{th}^c}{\omega_{0(th)}^c}, C_2 = \frac{Ph_{th+1}^c}{\omega_{1(th)}^c}, \ldots \frac{Ph_L^c}{\omega_{1(th)}^c} \quad (2.2)$$

dengan $\omega_o(th)$ dan $\omega_1(th)$ adalah distribusi peluang untuk C_1 dan C_2, dimana

$$\omega_0^c(th) = \sum_{i=1}^{th} Ph_i^c, \ \omega_1^c(th) = \sum_{i=th+1}^{L} Ph_i^c$$

$$\mu_0^c = \sum_{i=1}^{th} \frac{iPh_i^c}{\omega_0^c(th)}, \ \mu_1^c = \sum_{i=th+1}^{L} \frac{iPh_i^c}{\omega_1^c(th)}$$

Varian dari Otsu dapat dihitung dengan menggunakan persamaan (2.3) dan (2.4):

$$\sigma^{2^c} = \sigma_1^c + \sigma_2^c, \quad (2.3)$$

$$\sigma_1^c = \omega_0^c(\mu_0^c + \mu_T^c)^2, \ \sigma_2^c = \omega_1^c(\mu_1^c + \mu_1^c)^2 , \quad (2.4)$$

dengan $\mu_T^c = \omega_0^c \mu_o^c + \omega_1^c \mu_1^c$ dan $\omega_0^c + \omega_1^c = 1$.

Persamaan (2.5) merupakan fungsi objektif pada nilai σ_1^c dan σ_2^c:

$$J(\boldsymbol{TH}) = \max(\sigma^{2^c}(\boldsymbol{TH})) \tag{2.5}$$

dimana $0 \leq th_i \leq L - 1,\ \ i = 1,2,\ldots,k,$

dengan $th = th_1, th_2, \ldots, th_{k-1}$ merupakan vektor dari multilevel threshold dan varians dihitung menggunakan persamaan (2.6):

$$\sigma^{2^c} = \sum_{i=1}^{k} \sigma_i^c, \tag{2.6}$$

Lokal atau Adaptif Thresholding adalah metode segmentasi berdasarkan intensitas yang dianggap paling baik diantara metode lainnya. Hal ini juga lebih populer karena, selain baru atau terkini, thresholding dapat dengan mudah diaplikasikan secara komputasi lebih cepat pada enhancement citra (Qiang & Guoying, 2016), (Holambe & Kumbhar, 2016), (Vijay & Patil, 2016). Namun, nilai threshold yang diperoleh tidak selalu tepat karena ada faktor dari citra noise, gelap, dan ambiguitas antar kelas.

Segmentasi menggunakan multilevel thresholding memerlukan waktu yang lama dan melibatkan perhitungan yang besar. Untuk mengatasi masalah ini, beberapa metode optimasi diterapkan (Chen et al., 2016). Ada beberapa metode optimasi yang telah berhasil diterapkan pada multilevel thresholding, termasuk Genetic Algorithms (GA) dan Particle Swarm Optimization (PSO). Salah satu contoh menarik dari multilevel thresholding dengan GA ditunjukkan pada (Manikandan, Ramar, Willjuice Iruthayarajan, & Srinivasagan, 2014). Penerapan metode multilevel thresholding pada citra MRI menghasilkan nilai fungsi objektif tertinggi dibandingkan dengan metode PSO, Bacterial Foraging Algorithm(BF), Adaptive Bacterial Foraging Algorithm(ABP dan Real Coded Genetic Algorithm(RGA) dan menariknya terjadi penurunan kompleksitas waktu. Selanjutnya, (Kotte, Rajesh Kumar, & Injeti, 2015) dalam penelitiannya menggunakan multilevel thresholding untuk melakukan segmentasi dengan Improved Enhanced Differential Search Algorithm (IDSA). Segmentasi citra dengan multilevel berdasarkan PSO yang pernah dilakukan oleh (Mozaffari & Lee, 2017),(Mishra, Bose, Chandra De, & Pradhan, 2014),(Liu, Mu, Kou, & Liu,

2014),(Maryam, Mustapha, & Younes, 2017). Di sini, metode Kapur yang merupakan fungsi dari nilai threshold optimalisasi dengan metode optimasi PSO.

Penelitian ini menggunakan teknik thresholding untuk segmentasi citra (Somasundaram & Kalavathi, 2011). Teknik ini digunakan dengan metode thresholding Otsu. Metode Otsu adalah teknik thresholding klasik. Pada awalnya, metode penelitian pada thresholding bertingkat Otsu dinyatakan lebih baik daripada teknik thresholding klasik lainnya yang merupakan metode Kapur (Oliva et al., 2013), algoritma Otsu yang lebih baik untuk masalah bentuk dan pengukuran tidak seragam dalam kasus multilevel thresholding (Singh, Agarwal, Kashyap, & Bhattacharya, 2016)(Kurugollu, Sankur, & Harmanci, 2001).

2.3.2. Teknik Optimasi Segmentasi

Dikarenakan selama segmentasi citra terdapat beberapa kendala sehingga dibutuhkan proses optimasi. Metode optimasi segmentasi citra diantaranya Genetik Algorithm(GA), Particle Swarm Optimisation(PSO), Quantum Particle Swarm Optimisation (QPSO), Gray Wolf Optimizer (GWO), Bacterial Foraging Optimization (BFO) dan Conditional Random Field (CRF).

Pada optimasi segmentasi citra menggunakan QPSO, metode segmentasi yang digunakan adalah otsu multi thresholding. Ini dikarenakan multi thresholding otsu telah membuktikan hasil segmentasi lebih baik dibandingkan otsu thresholding, sebab multi thresholding otsu dapat meningkatkan kerumitan komputasi dan keterbatasan pengaplikasian algoritma. QPSO digunakan untuk mengatasi kekurangan dalam hal pencarian effisiensi (Cao, Ding, Fu, & Chen, 2016). Penelitian tersebut membuktikan multi thresholding dapat meningkatkan pencarian efisiensi. Banyaknya nilai thresholding yang digunakan sama dengan banyaknya nilai thresholding pada penelitian ini, terdapat 4 nilai threshold yakni 2, 3, 4, 5. 4 citra yang digunakan yakni citra lena, cameramen, pepper, dan rice diujikan dengan algoritma otsu bersama QPSO, PSO, dan GA. Hasil pengujian menunjukkan citra yang diuji menggunakan algoritma otsu dan QPSO lebih baik

dibanding otsu dan PSO maupun otsu dan GA. Penggabungan algoritma otsu dan QPSO menunjukan hasil yang terbaik dalam hal waktu pencarian. Metode GWO digunakan karena meskipun multi threholding merupakan salah satu bagian terpenting dalam bidang dalam segmenatasi citra. Namun semakin banyak jumlah thresholding maka semakin rumit pula komputasi dari multi thresholding, untuk itu digunakanlah algoritma GWO (A. K. Khairuzzaman & Chaudhury, 2017). Algoritma ini dibantu teknik kapur dan otsu dalam proses segmentasi. Algoritma ini kemudian dibandingkan dengan metode PSO dan BFO berbasis metode multi thresholding. MSSIM atau Mean Structural SiMilarity digunakan untuk mengetahui kualitas citra segmentasi. Hasil penelitian menunjukan bahwa metode lebih stabil dan menghasilkan solusi dengan kualitas lebih tinggi dibandingkan dengan berbasis PSO dan BFO. Untuk masalah waktu pemrosesan, penggunaan algoritma GWO lebih cepat daripada BFO meskipun lebih lambat dari metode berbasis PSO. CRF optimasi digunakan pada segmentasi citra yang bertipe sematik, dimana label sebuah piksel dapat mendeteksi adanya masalah. Metode ini dapat melakukan optimasi pada delinaeasi (Li, Yang, Chen, & Zhi, 2017). Hasil dari penelitian memperlihatkan persentase keunggulan sebanyak 2%.

Harmony Search Algorithm (HSA) merupakan sebuah metode optimasi *metahueristic* baru yang diperkenalkan oleh Z. W. Geem, J. H. Kim, dan G.V. Loganathan pada tahun 2001, dimana metode tersebut telah memperoleh hasil yang sangat baik di bidang optimasi (Manjarres et al., 2013).

HSA terinspirasi dari proses improvisasi musisi jazz, yakni dari fenomena musik opera yang terdiri dari berbagai macam alat musik dan menghasilkan melodi yang indah. Prinsip dari algoritma ini ialah meniru proses perbaikan harmoni yang dilakukan oleh kelompok paduan musik. Ketika kelompok paduan musik melakukan perbaikan pada harmoni musik yang dimainkan, maka terdapat tiga kemungkinan pilihan, yakni memainkan harmoni yang terkenal berdasarkan ingatan mereka, memainkan harmoni yang mirip dengan harmoni yang terkenal namun ada sedikit penyesuaian, atau membuat harmoni baru. Geem (2001) memformulasikan ketiga hal tersebut menjadi penggunaan *harmony memory* (HM), *pitch adjustment rate* (PAR), dan proses improvisasi secara acak.

Di sisi lain, literatur tentang metode optimasi evolusioner adalah implementasi multilevel thresholding dengan Harmony Search Algorithm (HSA), dimana HSA mengoptimalkan algoritma yang terinspirasi oleh improvisasi musisi jazz, yaitu dari fenomena musik opera yang terdiri dari berbagai alat musik. dan menghasilkan melodi yang indah. Algoritma ini diperkenalkan oleh Geem et al (Oliva et al., 2013)(Zong Woo Geem, Joong Hoon Kim, & Loganathan, 2001). Pada penelitian sebelumnya, multilevel thresholding dengan HSA digunakan untuk segmentasi citra dengan dua metode thresholding, yaitu Otsu, dan Kapur. Multilevel Thresholding dengan Otsu yang telah dioptimalkan menggunakan HSA menunjukkan hasil yang lebih baik bila dibandingkan dengan Kapur (Oliva et al., 2013). Improved Harmony Search Algorithm (IHSA) adalah metode yang lebih maju dari metode HSA sebelumnya. IHSA menggunakan parameter yang sama yang digunakan di HSA, perbedaannya terletak pada penyesuaian PAR dan BW (Mahdavi et al., 2007). IHSA juga telah diterapkan dalam masalah segmentasi untuk citra otak (Yang et al., 2016).

Setelah citra selesai dalam proses thresholding, nilai thresholding dioptimalkan dengan HSA (Zong Woo Geem et al., 2001). Pengolahan citra medis menggunakan HSA yang mampu mengoptimalkan citra otak MRI (Kalavathi, 2013)(Mahdavi et al., 2007) dan mengoptimalkan segmentasi kanker payudara pada mammogram digital (Maleki, Nooshyar, & Fatin, 2014).

2.4. Enhancement

Image Enhancement timbul dan diusulkan oleh masalah yang sering dialami oleh citra yang mempunyai noise pada saat transmisi atau buram, terlalu gelap, terang saat mendapatkan citra melalui media kamera misalnya. Enhancement harus melibatkan manipulasi tingkat abu-abu dan tingkat kecerahan gambar, pengurangan noise, penajaman, penyaringan, interpolasi, pseudocolor, perbesaran, dan sebagainya. Enhancement berguna dalam analisis citra, tampilan informasi visual dan ekstraksi (Hartono, 2014). Segmentasi citra dengan enhancement dapat digunakan untuk aplikasi lebih lanjut, misalnya untuk pengenalan objek aplikasi pada citra.

Agar citra lebih terang atau lebih gelap sehingga bisa dilakukan dengan mengubah kecerahan citra. Kecerahan dapat diperbaiki dengan menambahkan (atau mengurangkan) sebuah konstanta ke (atau di luar) setiap piksel pada citra. Karena operasi ini, histogram citrapun bergeser. Citra kontras rendah dapat diperbaiki dengan operasi peregangan kontras. Melalui operasi ini, nilai piksel abu-abu mulai dari 0 sampai 255 (pada citra 8-bit), bukan semua nilai piksel abu-abu yang digunakan secara merata. Proses enhancement citra bertujuan untuk mengklarifikasi tepi pada objek citra dan karena itulah operasi ini juga sering disebut sebagai edge sharpening. Operasi penghilangan noise pada domain spasial bertujuan untuk menghilangkan noise dengan cara noise filtering menggunakan median filter. Dari penelitian sebelumnya oleh (Z. Wang & C.Bovix, 2002),(Sayood, 2002) dan (Al-najjar & Soong, 2012) mengatakan bahwa pengukuran kualitas citra yang baik harus akurat dan konsisten dalam memprediksi kualitas.

Image enhancement termasuk pengembangan atau peningkatan citra sehingga hasilnya lebih cocok untuk operasi lebih lanjut. Peningkatan citra berarti citra lebih memuaskan untuk analisis, pemrosesan, atau tampilan. Ini mungkin termasuk dalam prosedur, misalnya, meningkatkan kompleksitas atau mencerahkan citra. Histogram citra memberikan data penting tentang keberadaan citra. Ini terdiri dari grafik yang menunjukkan berapa kali setiap level grayscale terjadi di dalam citra. Garis horizontal grafik menunjukkan kemungkinan tingkat grayscale dalam citra, misalnya, 0–255. Garis vertikal grafik menunjukkan jumlah kemunculan piksel level grayscale tersebut (Panchal et al., 2016).

2.4.1. Contrast Enhancement

Peningkatan kontras sering disebut sebagai salah satu masalah paling penting dalam pemrosesan citra. Kontras dibuat oleh perbedaan dalam pencahayaan tercermin dari dua permukaan yang berdekatan. Dalam persepsi visual, kontras ditentukan oleh perbedaan warna dan kecerahan suatu objek dengan objek lain. Teknik digital linear dan nonlinier adalah dua metode yang dipraktekkan secara luas untuk meningkatkan kontras citra (Al-amri, 2011):

A. Linear Contrast Enhancement

Jenis ini disebut Contras Stretching secara linier memperluas nilai digital asli dari data penginderaan jauh ke dalam distribusi baru. Dengan memperluas nilai input asli dari citra, kisaran total sensitivitas perangkat layar dapat dimanfaatkan. Peningkatan kontras linear juga membuat variasi halus dalam data lebih jelas. Ada tiga metode dalam peningkatan kontras linear (Al-amri, 2011):

1. Min-Max Linear Contrast Stretch

Saat menggunakan maksimum minimum kontras linear stretch, nilai minimum dan maksimum asli data ditetapkan ke kumpulan nilai baru yang ditentukan yang memanfaatkan rentang penuh nilai kecerahan yang tersedia. Pertimbangan citra dengan nilai kecerahan minimum 45 dan nilai maksimum 205. Ketika citra seperti itu dilihat tanpa penyempurnaan, nilai 0 hingga 44 dan 206 hingga 255 tidak ditampilkan. Perbedaan spektral yang penting dapat dideteksi dengan merentangkan nilai minimum 45 hingga 0 dan nilai maksimum 120. Metode ini diterapkan dengan memperhatikan jenis aplikasi citra. $g(x,y) = (f(x,y) - min)/(max - min)) * no.$ dari tingkat intensitas (i) di mana, $g\ (x,y)$ mewakili citra, di sisi kiri itu mewakili citra output, sedangkan $f\ (x,y)$ itu mewakili citra input. Dalam persamaan ini, "min" dan "max" adalah nilai intensitas minimum dan nilai intensitas minimum dalam citra saat ini. Di sini "no. dari intensity levels" menunjukkan jumlah total nilai intensitas yang dapat ditetapkan untuk sebuah piksel. Misalnya, biasanya dalam citra tingkat abu-abu, intensitas terendah yang mungkin adalah 0, dan nilai intensitas tertinggi adalah 255. Jadi "tidak ada tingkat intensitas" sama dengan 255. Berikut persamaan (2.7)-(2.9):

a. Untuk $0 \leq f_i(x,y) < A_1$, maka

$$f_0(x,y) = f_i(x,y)\frac{B_1}{A_1} \tag{2.7}$$

b. Untuk $A_1 \leq f_i(x,y) < B_2$, maka

$$f_0(x,y) = B_1 + (f_i(x,y) - A_1)\frac{B_2 - B_1}{A_2 - A_1} \tag{2.8}$$

c. Untuk $A_2 \leq f_i(x,y) \leq 255$, maka

$$f_0(x,y) = B_1 + (f_i(x,y) - A_2)\frac{255-B_2}{255-A_2} \quad (2.9)$$

Keterangan =

$f_0(x,y)$ = Nilai *contras* akhir pada titik *pixel x* dan *y*

$f_i(x,y)$ = Nilai *contras* pada titik *pixel x* dan *y*

A_1 = Nilai *contras* awal

B_1 = Nilai *brightness* awal

A_2 = Nilai *contrast* tujuan

B_2 = Nilai *brightness* tujuan

2. Percentage Linear Contrast Stretch

Persentasi kontras linear stretch mirip dengan rentang kontras linear maksimum minimum, namun metode ini menggunakan nilai minimum dan maksimum yang ditentukan terletak dalam persentase piksel tertentu dari rata-rata histogram. Standar deviasi dari rata-rata sering digunakan untuk mendorong ekor histogram melampaui nilai minimum dan maksimum semula.

3. Piecewise Linear Contrast Stretch

Ketika distribusi histogram dalam suatu citra adalah *bi* atau trimodal, seorang analis dapat meregangkan nilai-nilai tertentu dari histogram untuk peningkatan di area yang dipilih. Metode peningkatan kontras ini disebut peningkatan kontras linear piecewise. Peningkatan kontras linear piecewise melibatkan identifikasi sejumlah langkah peningkatan linier yang memperluas rentang kecerahan dalam mode histogram. Tipe ini dapat diekspresikan dengan persamaan(2.10):

$$f(x,y) = \begin{cases} ax, & 0 \le x \le x_1 \\ b(x-x_1) + y_{x1}, & x_1 \le x \le x_2 \\ c(x-x_2) + y_{x1,} & x_2 \le x \le B \end{cases} \quad (2.10)$$

Di mana: $f(x,y)$ adalah Piecewise Kontras Linear Stretch dalam citra, a, b, dan c adalah konstanta yang sesuai, merupakan lereng di daerah masing-masing dan B adalah nilai intensitas maksimum.

B. Nonlinear Contrast Enhancement

Nonlinier contrast enhancement sering melibatkan ekualisasi histogram melalui penggunaan algoritma. Metode peningkatan kontras nonlinear memiliki satu kerugian besar. Setiap nilai dalam citra input dapat memiliki beberapa nilai dalam citra output sehingga objek dalam adegan aslinya kehilangan nilai kecerahan relatifnya yang benar. Ada tiga metode dalam peningkatan kontras nonlinier (Al-amri, 2011) :

1. Histogram Equalizations

   Histogram equalizations adalah salah satu bentuk peningkatan kontras nonlinier yang paling berguna. Ketika histogram citra disetarakan, semua nilai piksel dari citra disebarluaskan sehingga ada sekitar jumlah piksel yang sama untuk setiap kelas skala grayscale yang ditentukan pengguna (misalnya, 32, 64, dan 256). Kontras meningkat pada rentang nilai kecerahan histogram yang paling banyak penduduknya (atau "puncak").

2. Adaptive Histogram Equalization

   Adaptive histogram equalizations di mana dapat membagi citra menjadi beberapa domain persegi panjang, hitung histogram penyeimbang dan ubah levelnya sehingga cocok dengan batas-batas. Tergantung pada sifat non-keseragaman citra.

3. Homomorphic Filter

   Homomorphic filter adalah filter yang mengontrol baik komponen frekuensi tinggi maupun frekuensi rendah. Filtering homomorphic bertujuan untuk menangani intensitas citra yang besar, ia memiliki model multiplikasi. Ketika citra diperoleh dengan cara optik, citra objek adalah produk dari sumber cahaya yang menerangi dan pemantulan objek, seperti yang dijelaskan persamaan (2.11) berikut ini :

$$f(x,y) = I(x,y)\,\rho\,(x,y) \tag{2.11}$$

   Di mana I adalah intensitas dari sumber cahaya yang menyinari, f adalah citra, dan $0 \leq \rho \leq 1$ adalah pemantulan objek. Untuk memisahkan dua komponen, mereka harus aditif.

Oleh karena itu, dengan mengubah citra ke dalam domain log, dimana komponen perkalian menjadi tambahan, seperti persamaan (2.12)

$$Ln(f) = ln(I) + \ln(\rho) \quad (2.12)$$

Karena logaritma natural adalah monotonik, $ln(I)$ adalah lintasan rendah dan $ln\ (\rho)$ adalah lintasan tinggi. Sekarang menjadi sebuah citra $f' = ln(f)$, yang memiliki komponen-komponen tambahan dan karenanya dapat secara selektif disaring oleh filter linear.

2.5. Ekstraksi Fitur

Teknik analisis citra retina mengandalkan teknik komputasi untuk membuat penilaian kualitatif mata agar lebih dapat direproduksi dan obyektif (Nadu, Thulasi, Simonthomas, & Asharaf, 2014). Pada penelitian ini dilakukan ekstraksi berdasarkan 3(tiga) jenis yaitu tekstur, bentuk dan warna.

2.5.1. Ekstraksi Fitur Berdasarkan Tekstur

Beberapa penelitian mengenai deteksi penyakit citra retina dilakukan menggunakan metode *Gray Level Co-occurrence Matric* tetapi belum mampu membangun sistem yang efektif untuk mengurangi waktu (Jadhav & Patil, 2018)(Nivetha, Sumathi, & Chandrasekaran, 2017). Penelitian lain mengenai deteksi penyakit retina dengan cara mendeteksi berbagai lesi dengan sifat tekstur yang berbeda dari retina manusia non-diabetes (Suryawanshi, 2017). Meskipun deteksi penyakit retina sudah dikatakan berhasil menganalisis citra retina dan memiliki kinerja yang cukup baik namun masih dibutuhkan pengembangan yang lebih luas terhadap cara mengklasifikasi secara otomatis semua penyakit pada citra retina.

Sebuah metode telah dikembangkan untuk *clustering* semua penyakit pada citra retina. Metode tersebut adalah *Self-Organizing Map*. Metode ini mampu menunjukkan hasil yang menjanjikan sebagai *clustering* penyakit (Anitha, Vijila, Hemanth, & Ahsina, 2009). Metode ini juga memiliki kelebihan yaitu kemampuan untuk menghitung korespodensi yang akurat dan kuat, dengan menemukan titik

dalam dua atau lebih citra yang sesuai pada titik yang sama (Mouravliansky & Delibasis, 2004).

Ekstraksi fitur tekstur adalah konten penting dari analisis tekstur, yang merupakan metode yang efektif untuk memecahkan masalah heterogenitas spektral dan distribusi spasial yang kompleks dalam kategori yang sama (X. Zhang, Cui, Wang, & Lin, 2017). Sangat penting untuk mengukur tekstur secara wajar dan efektif karena fitur tekstur yang diekstrak secara langsung mempengaruhi kualitas pemrosesan berikutnya. Saat ini, metode ekstraksi informasi tekstur adalah sebagai berikut: metode statistik, transformasi wavelet, metode fraktal, Markov Random Field (MRF) dan seterusnya.

Algoritma Ekstraksi Fitur Tekstur

Tuceryan dan Jain (Tuceryan & Jain, 1998) membagi metode yang berbeda untuk ekstraksi fitur ke dalam empat kategori utama, yaitu: struktural, statistik, model-based, dan transform domain, yang dijelaskan secara singkat di bagian berikut.

A. Metode Struktural

Struktur mewakili tekstur sesuai dengan sifat lokal (tekstur mikro) dan organisasi spasial (tekstur makro) dari properti lokal. Pendekatan struktural memberikan deskripsi simbolik yang baik dari citra dan berguna untuk pembentukan tekstur serta analisis tekstur (Tuceryan & Jain, 1998). Metode ini tidak cocok untuk tekstur alami karena variabilitas baik tekstur mikro dan tekstur makro dan tidak ada perbedaan yang jelas.

B. Metode Statistik

Metode statistik mewakili tekstur secara tidak langsung menurut sifat non-deterministik yang mengelola distribusi dan hubungan antara tingkat abu-abu dari suatu citra. Teknik ini adalah salah satu metode pertama dalam visi mesin. Dengan menghitung fitur-fitur lokal pada setiap titik dalam citra, dan menurunkan

sekumpulan statistik dari distribusi fitur-fitur lokal, metode statistik dapat digunakan untuk menganalisis distribusi spasial nilai-nilai abu-abu. Berdasarkan jumlah piksel yang mendefinisikan fitur lokal, metode statistik dapat diklasifikasikan ke dalam orde pertama (satu piksel), orde kedua (dua piksel) dan statistik tingkat tinggi (tiga atau lebih piksel). Perbedaan antara kelas-kelas ini adalah bahwa statistik orde pertama mengestimasi properti (misalnya rata-rata dan varians) dari nilai-nilai piksel individu dengan mengabaikan interaksi spasial antara piksel citra, tetapi dalam orde kedua dan statistik orde tinggi memperkirakan properti dari dua atau lebih nilai piksel yang terjadi di lokasi tertentu relatif terhadap satu sama lain. Fitur statistik orde kedua yang paling populer untuk analisis tekstur berasal dari matriks co-kejadian. Pendekatan berdasarkan matriks multidimensional co-occurrence baru-baru ini menunjukkan mengungguli paket wavelet (teknik berbasis transformasi) ketika diterapkan untuk klasifikasi tekstur. Deskripsi beberapa metode statistik adalah:

1. Histogram Orde Pertama

Citra adalah fungsi *f(x, y)* dari dua dimensi *x* dan *y*, *x* {0, 1, ..., *Nx* - 1} dan *y* = {0, 1, ..., *Ny* -1}. The *f(x, y)* dapat mengambil nilai-nilai diskrit *i* = {0, 1, ..., *Ng* - 1}, *Ng* adalah jumlah total tingkat intensitas (jumlah piksel) dalam citra. Histogram adalah diagram yang menunjukkan berapa banyak piksel suatu citra memiliki intensitas tertentu. Ini memiliki beberapa keuntungan, salah satunya adalah histogram dari citra dan rotasi citra-nya adalah sama, dan yang lain adalah bahwa ukuran tempat penyimpanan untuk histogram lebih rendah dari ukuran penyimpanan citra. Tingkat intensitas dihitung dengan persamaan (2.13) dan (2.14) ini:

$$h(i) = \sum_{x=0}^{N_g-1} \sum_{y=0}^{N_g-1} \delta(f(x,y), i), \tag{2.13}$$

di mana δ *(i, j)* adalah fungsi delta Kroneker:

$$\delta(i,j) = \begin{cases} 1 & j = i \\ 0 & j \neq i \end{cases} \tag{2.14}$$

2. Histogram Orde Kedua

Dalam orde kedua, hubungan antara dua piksel yang biasanya adalah lingkungan dianggap. Pada tahun 1979, Haralick (Haralick, Shanmugam, & Dinstein, 1973), mendefinisikan matriks co-occurrence sebagai statistik histogram orde kedua dan ini adalah salah satu metode analisis tekstur yang paling terkenal. Metode ini dikenal sebagai Gray Level Co-occurrence Matrix (GLCM). Hal ini berguna dalam aplikasi di mana distribusi spasial tingkat abu-abu penting (misalnya dalam sinyal radar), atau dalam aplikasi analisis citra (misalnya biomedis), dan juga berguna untuk teknik penginderaan jarak jauh yang penting dalam menangkap informasi kerusakan disebabkan oleh gempa bumi. Ada banyak jenis penelitian berdasarkan fitur tekstur Haralic. Pada urutan kedua, ukuran digunakan untuk mempertimbangkan hubungan antara kelompok dua piksel (biasanya bertetangga) dalam citra. Diasumsikan bahwa citra disimpan sebagai array 2D, *f (x, y).* Domain spasial *x* dan *y* adalah *Lx* = {1, 2, ...,*Nx*} sebagai domain spasial horizontal, *Ly* = {1, 2, ...,*Ny*} sebagai domain spasial vertikal. *Lx* × *Ly* adalah himpunan masing-masing piksel dan citra digital *I* adalah fungsi yang menetapkan nilai level abu-abu (nilai kecerahan) dari *G* = {1, 2, ...,*Ng*} untuk setiap piksel. Matriks de_nes probabilitas bergabung dengan dua piksel $P_{d,\theta}(i,j)$ yang memiliki nilai i dan j, dengan jarak dan _ sebagai sudut orientasi. Matriks co-kejadian dapat dihitung dengan menggunakan dua teknik. Pertama, piksel citra dipisahkan oleh *d* dan θ untuk arah tertentu (θ) dalam empat arah (0, 45, 90, 135). Kedua, piksel citra dipisahkan oleh jarak d dalam delapan arah (0, 45, 90, 135, 180, 225, 270, 315). Matriks histogram 2D ini mewakili transisi antara semua pasang dua tingkat abu-abu. $N_{d,\theta}(i,j)$ menunjukkan jumlah transisi antara dua piksel yang tingkat keabuannya adalah *i, j* dengan *d* piksel jauh dan θ sebagai sudut orientasi. Dimensi matriks persegi sama dengan jumlah tingkat intensitas dalam citra, untuk setiap jarak *d* dan orientasi. Jaraknya bisa 1 dan 2 piksel berdasarkan sudut (θ) dipilih antara 0 hingga 360. Untuk klasifikasi, tekstur halus nilai-nilai kecil *d* diperlukan, sedangkan tekstur kasar membutuhkan nilai besar *d.* Penurunan jumlah tingkat intensitas dengan mengukur citra ke tingkat intensitas yang lebih sedikit meningkatkan kecepatan perhitungan, dengan hilangnya beberapa informasi

tekstual. Transfer piksel dalam matriks co-kejadian kemudian menghitung sejumlah besar fitur statistik dari matriks.

3. Urutan lebih tinggi : Dalam metode ini, hubungan antara tiga atau lebih piksel dipertimbangkan.

C. Metode Berbasis Model

Analisis tekstur berbasis model seperti model Fractal dan Markov didasarkan pada konstruksi citra yang dapat digunakan untuk menggambarkan tekstur dan mensintesisnya. Metode ini mendeskripsikan citra sebagai model probabilitas atau sebagai kombinasi linear dari sekumpulan fungsi dasar (Tuceryan & Jain, 1998).

Model Fraktal berguna untuk pemodelan tekstur alam tertentu yang memiliki kualitas statistik kekasaran pada skala yang berbeda, dan juga untuk analisis tekstur dan diskriminasi. Metode ini memiliki kelemahan dalam orientasi orientasi dan tidak berguna untuk menggambarkan struktur citra lokal. Model berbasis-pixel melihat citra sebagai kumpulan piksel, sedangkan model berbasis wilayah melihat citra sebagai kumpulan sub plat. Ada berbagai jenis model berdasarkan sistem lingkungan dan sumber kebisingan yang berbeda. Tipe-tipe ini adalah model time-series satu dimensi, Auto-Regressive (AR), Moving Average (MA), dan Auto Regressive Moving Average (ARMA). Model lapangan acak menganalisis variasi spasial dalam dua dimensi, acak global, dan acak lokal. Model lapangan acak global memperlakukan seluruh citra sebagai realisasi bidang acak, dan model lapangan acak lokal mengasumsikan hubungan intensitas di lingkungan kecil. Sebuah kelas yang banyak digunakan dari model lapangan acak lokal adalah model Markov, di mana probabilitas bersyarat dari intensitas sebuah piksel hanya bergantung pada intensitas piksel di lingkungannya (yang disebut Markov tetangga).

D. Metode Transformasi Domain

Metode transformasi, seperti Fourier, Gabor, dan transformasi wavelet mewakili citra dalam ruang yang sistem koordinatnya memiliki interpretasi yang terkait erat dengan karakteristik tekstur (seperti frekuensi atau ukuran). Mereka menganalisis konten frekuensi citra. Metode berdasarkan transformasi Fourier memiliki kelemahan dalam lokalisasi spasial sehingga mereka tidak berkinerja baik. Filter Gabor menyediakan sarana untuk lokalisasi spasial yang lebih baik tetapi kegunaannya terbatas dalam praktik karena biasanya tidak ada satu resolusi filter di mana seseorang dapat melokalisasi struktur spasial dalam tekstur alami. Metode ini melibatkan transformasi citra asli dengan menggunakan filter dan menghitung energi dari citra yang diubah. Mereka didasarkan pada proses keseluruhan citra yang tidak baik untuk beberapa aplikasi yang didasarkan pada satu bagian dari citra input (Tuceryan & Jain, 1998).

Gray Level Co-occurrance Matric (GLCM)

Gray Level Co-occurrance Matric adalah metode ekstraksi fitur berbasis tekstur yang populer. GLCM menentukan hubungan tekstur antara *pixel* dengan melakukan operasi sesuai dengan statistik orde kedua dalam citra. Biasanya, dua *pixel* digunakan untuk operasi ini. Ekstraksi fitur statistik orde kedua dilakukan dengan matriks kookurensi, yaitu suatu matriks antara yang merepresentasikan hubungan ketetanggaan antar piksel dalam citra pada berbagai arah orientasi dan jarak spasial (Suresh & Shunmuganathan, 2012).

Fitur GLCM telah dihitung pada jarak 1 dan 2 masing-masing. Fitur berikut digunakan dalam algoritma ini.

Notasi (Nadu et al., 2014) :

$P(i,j)$ = (i,j) termasuk dalam normalisasi gray-tone spasial matriks ketergantungan, $= \frac{P(i,j)}{R}$.

$P_x(i)$ = i termasuk dalam matriks baris probabilitas yang diperoleh dengan menjumlahkan baris $P(i,j)$.

N_g = Jumlah graylevel yang berbeda pada citra terkuantisasi.

$\sum_i dan \sum_j \ \sum_{i=1}^{N_g} dan \sum_{j=1}^{N_g}$

$$P_y(j) = \sum_{i=1}^{N_g} P(i,j) \tag{2.20}$$

$$P_{x+y}(k) = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P(i,j),\ k = 2,3,\ldots,2Ng \tag{2.21}$$

$$P_{x-y}(k) = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P(i,j),\ k = 0,1,\ldots,2Ng - 1 \tag{2.22}$$

Langkah ini memberikan bukti tentang bagaimana tajamnya variasi struktural dalam citra. Fitur GLCM yang digunakan dalam penelitian pada umumnya adalah sebagai berikut; kontras, korelasi, energi, homogenitas, jumlah kuadrat (varians), jumlah rata-rata, jumlah varians, jumlah entropi, entropi, perbedaan varians, perbedaan entropi, ukuran informasi korelasi, koefisien korelasi maksimal (Öztürk & Akdemir, 2018). Sebuah matriks fitur GLCM yang dihasilkan dapat berhasil merepresentasikan citra dengan parameter yang lebih sedikit menggunakan properti ini (Nadu et al., 2014).

1. Contrast

   Variasi tingkat keabuan dalam matriks GLCM diwakili oleh kontras. Hal ini pada dasarnya menentukan ketergantungan linear tingkat abu-abu dua *pixel* tetangganya. Nilainya berkisar dari 0 hingga (ukuran (GLCM, 1) -1) 2. Nilai kontras adalah 0 untuk teksur yang konstan dihitung dengan persamaan (2.23)

$$Contrast = \sum_{i,j} (i-j)^2 p(i,j) \tag{2.23}$$

2. Correlation

   Korelasi memberi informasi tentang bagaimana *pixel* berkorelasi dengan *pixel* tetangganya. Nilainya berkisar dari -1 hingga 1, dimana -1 adalah korelasi negatif sempurna, 0 tidak berkorelasi dan 1 korelasi positif sempurna dengan persamaan (2.24).

$$Correlation = \sum_{i,j} \frac{(i-\mu i)(j-\mu j)p(i,j)}{\sigma i \sigma j} \tag{2.24}$$

Dimana μ dan σ adalah *mean deviation* dan *standard deviation* dari masing-masing matriks co-occurrence. *Mean deviation* adalah nilai ukuran brightness. Standard deviation adalah nilai ukuran contrast dengan

$$\mu i = \sum i(P_{i,j})$$

$$\mu j = \sum i(p_{i,j})$$

$$\sigma i = \sqrt{\sum P_{i,j}(i - \mu i)^2}$$

$$\sigma j = \sqrt{\sum P_{i,j}(j - \mu j)^2}$$

3. Angular Second Moment / Energy

   Angular Second Moment atau juga disebut *Energy* untuk mengukur keseragaman tekstur citra, yaitu pengulangan pasangan *pixel*. Ini juga membantu dalam menentukan gangguan dalam tekstur. Nilainya berkisar dari 0 hingga 1 dihitung dengan persamaan (2.25).

$$ASM/Energy = \sum_{i,j} p(i,j)^2 \quad (2.25)$$

4. Inverse Difference Moment / Homogeneity

   Inverse Difference Moment adalah ukuran dari homogeneitas lokal dan didefinisikan dengan persamaan (2.26) sebagai berikut :

$$IDM = \sum_{i,j} \frac{1}{1+(i,j)^2} P(i,j) \quad (2.26)$$

   Homogeneitas mengukur keseragaman entri nol dalam GLCM. Nilainya berkisar dari 0 hingga 1. Dihitung dengan persamaan (2.27):

$$Homogeneity = \sum_{i,j} \frac{p(i,j)}{1+|i-j|} \quad (2.27)$$

5. Sum of Squares : Variance

   Varians adalah ukuran yang memberitahu kita seberapa banyak graylevel yang bervariasi dari mean. Dihitung dengan persamaan (2.28):

$$Variance = \sum_{i,j}(i,j)P_{i,j} - \mu^2 \quad (2.28)$$

6. Sum Average. Dihitung dengan persamaan (2.29):

$$Sum\ Average = \sum_{i=2}^{2N_g} i\, P_{x+y}(i) \quad (2.29)$$

7. Sum Variance. Sum Average. Dihitung dengan persamaan (2.30):

$$Sum\ Variance = \sum_{i=2}^{2N_g} (i - f_8)^2 P_{x+y}(i) \quad (2.30)$$

8. Sum Entropy

   Sum Entropy adalah jumlah vektor acak independen. Sum Average. Dihitung dengan persamaan (2.31):

$$Sum\ Entropy = \sum_{i=2}^{2N_g} P_{x+y}(i) \log(P_{x+y}(i)) \quad (2.31)$$

9. Entropy

   Entropy adalah ukuran keacakan. Sum Average. Dihitung dengan persamaan (2.32):

$$Entropy = \sum_{b=0}^{L-1} P(b) log_2 P(b) \quad (2.32)$$

   Dimana, $P(b)N(b)/n^2$ untuk {0 ≤ b ≤ L-1}

   Dimana L adalah jumlah nilai piksel yang berbeda yang dapat mengadopsi, N(b) adalah jumlah piksel dari amplitudo (b) pada jendela piksel ukuran 'n×n'.

10. Difference Variance

    Difference Variance mengukur seberapa jauh satu set nomor tersebar. Sum Average. Dihitung dengan persamaan (2.33):

$$DV = variance\ dari\ P_{x-y} \quad (2.33)$$

11. Difference Entropy. Sum Average. Dihitung dengan persamaan (2.34):

$$DE = \sum_{i=0}^{N_g-1} P_{x-y}(i) \log(P_{x-y}(i)) \quad (2.34)$$

12. Information of Correlation. Sum Average. Dihitung dengan persamaan (2.35) dan (2.36):

$$IC = \frac{HXY - HXY1}{\max\{HX, HY\}} \quad (2.35)$$

$$IC = (1 - \exp[-2{,}0(HXY2 - HXY)])^{1/2} \quad (2.36)$$

$$HXY = -\sum_{i,j} P(i,j) \log\big(P(i,j)\big) \quad (2.37)$$

Dimana HX dan HY adalah entropi dari P_x dan P_y, dan

$$HXY1 = -\sum_{i,j} P(i,j)\log(P_x(i)P_y(j))$$

$$HXY2 = -\sum_{i,j} P_x(\mathrm{i})P_y(\mathrm{j})\log(P_x(i)P_y(j))$$

13. Maximal Correlation Coefficient

Koefisien korelasi menunjukkan kekuatam dan arah hubungan linier antara dua variabel acak. Sum Average. Dihitung dengan persamaan (2.38):

$$MCC = (Second\ largest\ eigenvalues\ dari\ Q)^2 \quad (2.38)$$

Dimana,

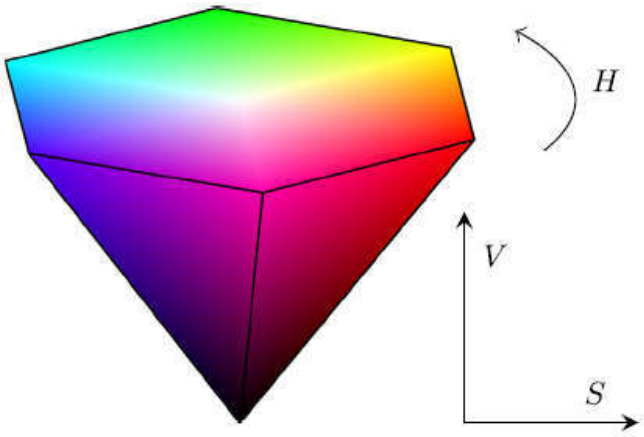
$$Q(i,j) = \sum_k \frac{P(i,k)P(j,k)}{P_x(i)P_y(k)}$$

2.5.2. Ekstraksi Fitur Berdasarkan Bentuk

Pada penelitian sebelumnya segmentasi pembuluh retina dilakukan untuk menemukan struktur retina lainnya seperti disk optik dan fovea. Pada pembuluh darah tersegmentasi digunakan untuk mengenali citra retina, berbagai algoritma segmentasi pembuluh darah disajikan untuk memantau secara otomatis perkembangan penyakit tertentu. (Sreejini & Govindan, 2015) (Sadikoglu & Uzelaltinbulat, 2016). *Threshold based segmentation* merupakan teknik penting yang digunakan dalam banyak aplikasi citra medis. Kriteria Otsu memilih ambang optimal dengan memaksimalkan varians kelas (Sivakamasundari & Natarajan, 2015) (Arora, Acharya, Verma, & Panigrahi, 2008). Pada penelitian sebelumnya segmentasi citra dilakukan dengan melibatkan dua buah teknik *thresholding* yakni *otsu* dan *kapur*. Perbedaan pada kedua teknik tersebut yakni algoritma *otsu* dapat memaksimalkan varian kelas (Otsu, 1979b) sedangkan algortima *kapur* dapat memaksimalisasi entropi untuk mengukur homogenitas antar kelas, Setelah hasil *thresholding* mendapatkan nilai intensitas citra keabuan, selanjutnya hasil tersebut ditingkatkan menggunakan algoritma *harmony search* (Oliva et al., 2013) (Dillak & Bintiri, 2012).

2.5.3. Ekstraksi Fitur Berdasarkan Warna

Untuk membedakan suatu objek dengan warna tertentu dapat menggunakan nilai hue yang merupakan representasi dari cahaya tampak (merah, jingga, kuning, hijau, biru, ungu). Nilai hue dapat dikombinasikan dengan nilai saturation dan value yang merupakan tingkat kecerahan suatu warna. Untuk mendapatkan ketiga nilai tersebut, perlu dilakukan konversi ruang warna citra yang semula RGB (Red, Green, Blue) menjadi HSV (Hue, Saturation, Value).



Gambar 2.1 Ruang Warna HSV(Xiong et al., 2017)

membagi komponen Hue menjadi beberapa daerah warna seperti Gambar 2.1 dan dihitung dengan persamaan (2.39)-(2.42) di bawah ini.

$$R' = \frac{R}{255} \tag{2.39}$$

$$G' = \frac{G}{255}$$

$$B' = \frac{B}{255}$$

$$C\ max = \max(R', G', B')$$

$$C\ min = \min(R', G', B')$$

$$\Delta = Cmax - Cmin$$

$$H = \begin{cases} 0^{\circ} & \Delta = 0 \\ 60^{\circ} \times \frac{G'-B'}{\Delta}, & Cmax = R' \\ 60^{\circ} \times \frac{G'-B'}{\Delta}, +2 & Cmax = G' \\ 60^{\circ} \times \frac{R'-G'}{\Delta}, +4 & Cmax = B' \end{cases} \quad (2.40)$$

$$S = \begin{cases} 0, & Cmax = 0 \\ \frac{\Delta}{Cmax}, & Cmax \neq 0 \end{cases} \quad (2.41)$$

$$V = Cmax \quad (2.42)$$

2.6. Klasifikasi Penyakit Kelainan Retina

2.6.1. Metode Self Organizing Map (SOM)

Sistem self-organizing adalah jenis sistem yang dapat mengubah struktur dan fungsi internal mereka dalam menanggapi keadaan eksternal dan rangsangan. Elemen dari sistem semacam itu dapat mempengaruhi atau mengatur elemen lain dalam sistem yang sama, menghasilkan stabilitas struktur atau fungsi yang lebih besar dari keseluruhan terhadap fluktuasi eksternal. Aspek utama dari sistem pengaturan diri adalah peningkatan kompleksitas, munculnya fenomena baru (keseluruhan lebih dari jumlah bagian-bagiannya) dan pengaturan internal oleh loop umpan balik positif dan negative (Miljković, 2017).

Self-Organizing Map

Neuron dari neural network dengan neuron komunikasi lateral secara topologis diatur sebagai self-organizing map adalah hal yang umum dalam neurobiologi. Sebagian besar otak tidak memiliki pasangan input-output yang diinginkan tersedia dan harus belajar dalam mode unsupervised.

SOM adalah jaringan neural lapisan tunggal dengan unit-unit yang diatur sepanjang grid n-dimensi. Sebagian besar aplikasi menggunakan grid dua dimensi dan persegi panjang, meskipun banyak aplikasi juga menggunakan grid heksagonal, dan seseorang, tiga, atau lebih banyak ruang dimensi. SOM menghasilkan citra proyeksi dimensi rendah dari distribusi data berdimensi tinggi, di mana hubungan kesamaan antara item data dipertahankan,

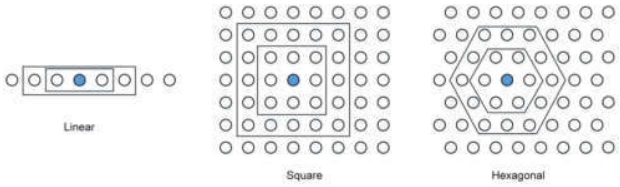
Prinsip Self-Organization pada SOM

Tiga proses umum dalam menyusun self-organization pada SOM (Miljković, 2017):

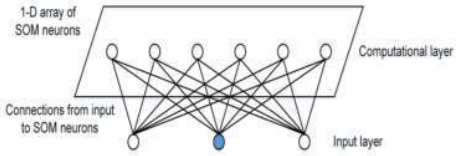
1. Proses Kompetitif
   Untuk setiap vektor pola masukan yang disajikan ke peta, semua neuron menghitung nilai dari fungsi diskriminan. Neuron yang paling mirip dengan vektor pola input adalah pemenang (Best Matching Unit, BMU).
2. Proses Koperasi
   Neuron pemenang (BMU) menemukan lokasi spasial dari lingkungan topologi neuron yang tereksitasi. Neuron dari lingkungan ini kemudian dapat bekerja sama.
3. Adaptasi Synaptic
   Menyediakan bahwa neuron bersemangat dapat mengubah nilai-nilai mereka dari fungsi diskriminan terkait dengan vektor pola masukan yang disajikan oleh proses penyesuaian berat.

Topologi Umum

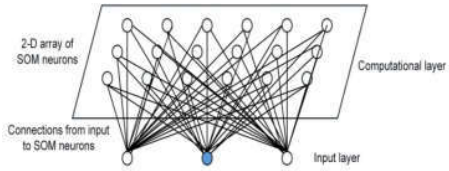
Topologi SOM bisa dalam satu, dua (paling umum) atau bahkan tiga dimensi. Grid dua dimensi yang paling banyak digunakan di SOM adalah grid persegi panjang dan heksagonal. Topologi tiga dimensi dapat berbentuk silinder atau bentuk toroid. 1-D (linear) dan 2-D grid diilustrasikan pada Gambar 2.2 dengan SOM yang sesuai pada Gambar 2.3 dan menurut Gambar 2.4 (Miljković, 2017).



Gambar 2.2 Jaringan grid dan neuron tetangga



Gambar 2.3 Jaringan SOM 1-D



Gambar 2.4 Jaringan SOM 2-D

Algoritma SOM

Algoritma pada jaringan saraf tiruan SOM sebagai berikut (Anitha et al., 2009):

1. Jika matriks vektor fitur ukuran $k \times m$ (k adalah jumlah dimensi vektor fitur, dan m adalah jumlah data), inisialisasi:
    - Jumlah yang diinginkan dari kelas j atau *cluster*
    - Jumlah komponen i dari matriks fitur vektor (k adalah baris dari matriks)
    - Jumlah vektor $X_{m,\,i}$ = Jumlah data (kolom matriks)
    - Bobot awal W_{ji} secara acak dengan interval waktu 0-1
    - *Learning rate* awal α (0)
    - Jumlah iterasi (e epoch)
2. Jalankan iterasi pertama sampai total iterasi (epoch)

3. Hitung vektor citra mulai dari 1 sampai m menggunakan persamaan (2.43) berikut :

$$D(j) = \sum_i (W_{ij} - X_{m,1})^2 \quad (2.43)$$

Untuk semua j

- Kemudian menentukan nilai minimum dari *D(j)*
- Lakukan perubahan pada bobot *j* dengan minimum dari *D(j)*, dengan persamaan (2.44).

$$W_{ij}^{new} = W_{ij}(old) + \alpha(X_i - W_{ij}(old)) \quad (2.44)$$

4. Memodifikasi *learning rate* (η) untuk iterasi dengan persamaan (2.45) berikutnya:

$$\alpha(t+1) = 0{,}6\alpha(t) \quad (2.45)$$

Untuk t dimulai dari iterasi pertama menuju *e*.

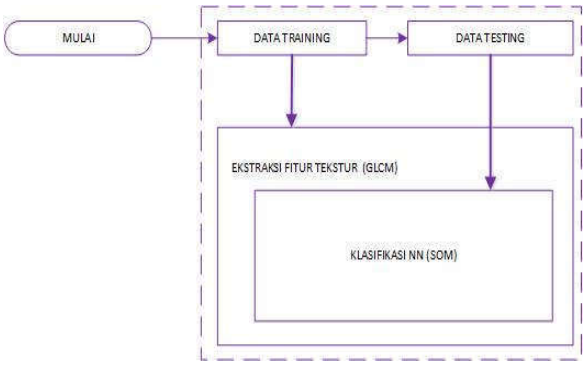
5. Menguji kondisi terminasi

Iterasi dihentikan jika perbedaan antara W_{ij} dan W_{ij} iterasi sebelumnya hanya sedikit atau perubahan bobot hanya perubahan yang sangat kecil, maka iterasi telah mencapai konvergensi sehingga dapat dihentikan.

6. Gunakan berat W_{Ji} yang telah konvergensi ke pengelompokan vektor fitur untuk setiap citra, dengan menghitung vektor jarak dengan bobot optimal.
7. Membagi citra (X_m) ke dalam kelas:
   - Jika D(1)<D(2)<D(3)<D(4)<D(5), maka citra yang termasuk dalam kelas 1
   - Jika D(2)<D(1)<D(3)<D(4)<D(5), maka citra yang termasuk dalam kelas 2
   - Jika D(3)<D(2)<D(1)<D(4)<D(5), maka citra yang termasuk dalam kelas 3
   - Jika D(4)<D(3)<D(2)<D(1)<D(5), maka citra yang termasuk dalam kelas 4

- Jika D(5)<D(4)<D(3)<D(2)<D(1), maka citra yang termasuk dalam kelas 5

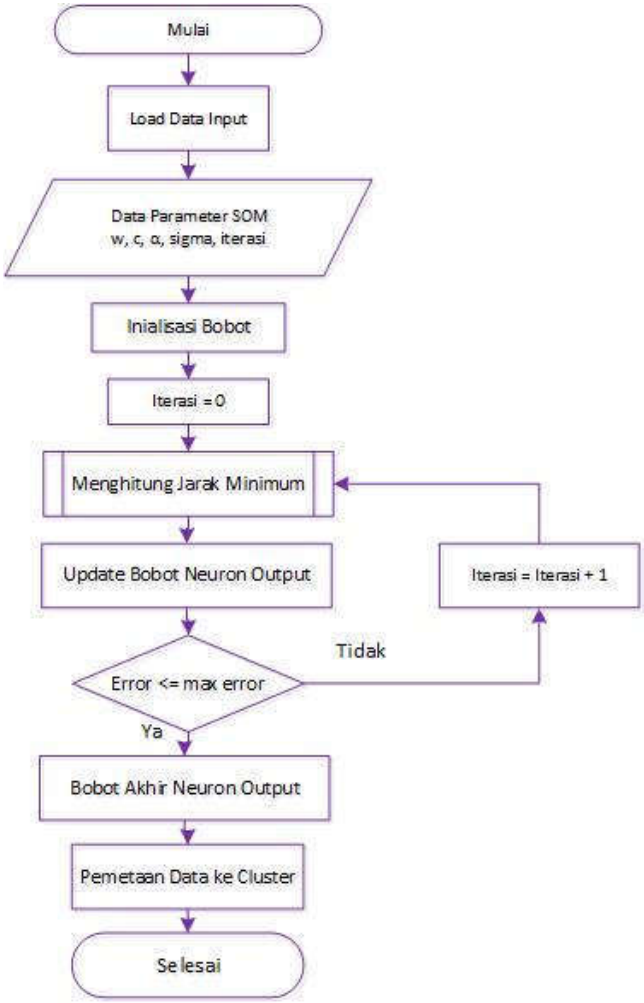
Proses klasifikasi citra menggunakan metode Self-Organizing Map. Tahap awal proses klasifikasi yang dilakukan adalah proses pelatihan input data pelatihan atau *training data*. Bagan klasifikasi jaringan saraf tiruan tersebut disajikan dalam bentuk *diagram blok* pada Gambar 2.5.



Gambar 2.5. Diagram blok klasifikasi dengan SOM

Algoritma Pelatihan SOM

Terdapat algoritma yang dirancang untuk klasifikasi penyakit retina pada penelitian ini yakni algoritma *Self-Organizing Map.* Perancangan algoritma tersebut disajikan dalam bentuk *flowchart* ditunjukkan pada Gambar 2.6.



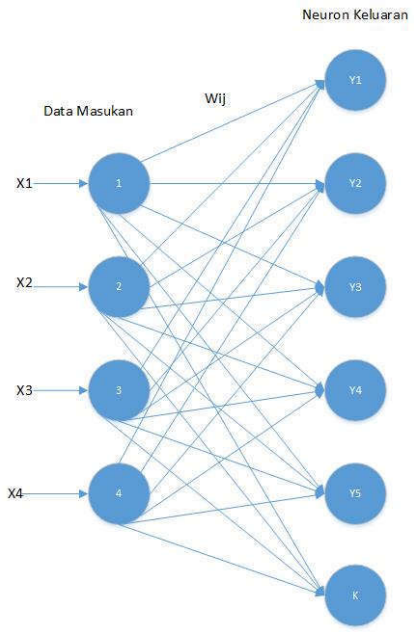
Gambar 2.6. Flowchart Algoritma Self-Organizing Map

Berdasarkan *flowchart* Gambar 2.6 diatas maka dapat dijelaskan bahwa tahapan *klasifikasi* dengan *Self-Organizing Map* adalah sebagai berikut :

1. Memuat data input berupa nilai vektor berdasarkan nilai hasil ekstraksi fitur tekstur GLCM.
2. Menentukan parameter input seperti ukuran map (x,y) sebagai jumlah *cluster,* bobot neuron, *learning rate*, *sigma*, dan *epoch*.
3. Inisialisasi bobot neuron output secara acak dengan rentang minimum dan maksimum sesuai bobot pada data *training*.
4. Saat iterasi = 0 sebagai iterasi awal untuk memulai proses *training* data.

5. Menghitung jarak minimum. Data input dipilih secara acak kemudian dihitung jaraknya dengan masing-masing *neuron* output sampai menemukan *neuron* pemenang.
6. Mengupdate nilai bobot *neuron* output.
7. Kondisi apakah error saat ini lebih kecil dari error maksimum. Jika Ya, maka lanjut ke langkah 8, jika Tidak maka kembali ke langkah 5.
8. Bobot akhir *neuron* yang didapatkan pada masing-masing *cluster* setelah proses *training* dan proses *testing*.
9. Memetakan data input kedalam *cluster.* Data input dihitung jaraknya dengan masing-masing *neuron* output menggunakan rumus *Euclidean Distance*. *Neuron* output dengan jarak yang terkecil menjadi *cluster*-nya.

Pada arsitektur jaringan SOM, terdiri dari 2 layer, layer input dan layer output. Setiap *node* dari input terhubung dengan seluruh *node* pada output layer. Namun *node* pada output layer (neuron) tidak terhubung satu sama lain. *Node* dalam output layer ini disebut juga *map node*. Untuk memudahkan arsitekturnya dapat dilihat pada Gambar 2.7 dibawah ini :



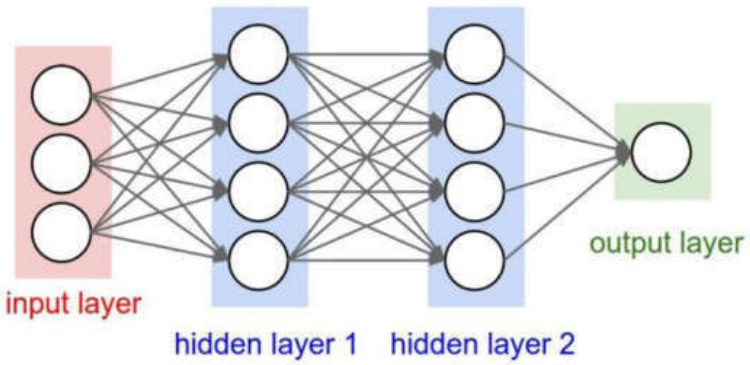
Gambar 2.7. Penerapan arsitektur SOM untuk klasifikasi

Pada Gambar 2.7 diatas, X_1 sampai X_4 yang merupakan layer input berupa 4 hasil nilai vektor dari proses GLCM seperti X_1 untuk nilai kontras, X_2 untuk nilai korelasi, X_3 untuk nilai energi, dan X_4 untuk nilai homogenitas. Y_1 sampai Y_5 merupakan layer output untuk masing penyakit retina seperti *Central Retinal Vein Occlusion, Retinitis, Proliverative Diabetic Retinopathy, Background Diabetic Retinopathy* dan *Coats*. K adalah jumlah neuron *cluster*.

2.6.2. Metode Multi Layer Perceptrons (MLP)

Multi-Layer Perceptrons merupakan Jaringan Syaraf Tiruan (JST*) feed-forward* yang terdiri dari sejumlah neuron yang terhubung oleh bobot-bobot penghubung. Neuron-neuron ini disusun dalam beberapa lapisan yang terdiri dari satu lapisan *input (input layer),* satu atau lebih lapisan terse mbunyi *(hidden layer)* dan satu lapis *output (output layer).* (Suartika, Wijaya, & Soelaiman, 2016) Proses pelatihan jaringan pada dasarnya merupakan proses penyesuaian bobot-bobot untuk masing–masing simpul antara lapisan input, lapisan tersembunyi, dan lapisan output. Penyesuaian bobot dilakukan secara terus-menerus sampai dicapai error yang paling minimum. Fungsi aktivasi yang digunakan adalah fungsi aktivasi sigmoid.

Adapun contoh arsitektur sederhana dari MLP seperti yang disajikan pada Gambar 2.8 sebagai berikut: (Suartika et al., 2016)



Gambar 2.8. Arsitektur MLP Sederhana

a. Algoritma *Multi-Layer Perceptrons*

Algoritma proses *Multi-Layer Perceptrons* dapat dilihat pada Gambar 2.9 di bawah ini:

1. Unit input (Xi, i=1,2,3.....n)

1. Menerima input Xi

2. dikirimkan ke semua Hidden Layer

2. Unit hidden (Zj,j=1,2,3....n)

1. menghitung semua vektor input beserta bobot:

Z_inj = Voj + Σ Xi Vij

2. Menghitung fungsi aktivasi setiap unit hidden sebagai output unit hidden dengan akivasi sigmoid.

Zj = f(z_inj)

f(z_inj= $\frac{1}{1+e^{-z_inj}}$

3. Memasukkan nilai aktivasi sebagai input untuk output

4. Unit Output (Yk, k = 1,2,3,....,n)

1. Menghitung semua input dengan bobot:

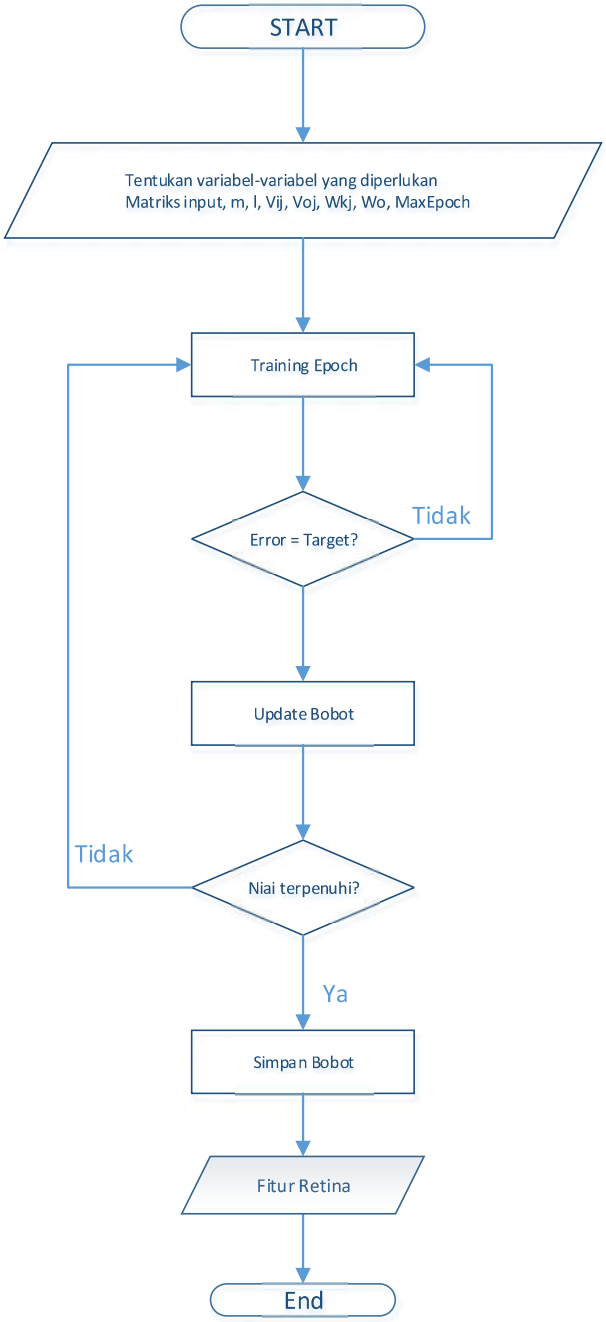
y_ink = wok + Σ

Zj Wjk

2. Menghitung nilai aktivasi setiap output sebagai output jaringan

f(y_ink= $\frac{1}{1+e^{-y_inj}}$

yk = f(y_ink)



Gambar 2.9. Flowchart Algoritma Multi-Layer Perceptrons

2.6.3. Metode Back Propagation(BP)

Klasifikasi citra merupakan suatu pengelompokan suatu pixel pada suatu citra ke dalam kelompok sehingga dapat diinterpretasikan sebagai suatu property yang spesifik. Kelasifikasi citra bertujuan untuk mengenali suatu objek dari suatu citra dan mengklasifikasikan citra ke salah satu yang telah ditentukan kategori (Zhao, Xu, Liu, Tian, & Jiang, 2018). Klasifikasi terbagi menjadi dua bagian yaitu klasifikasi terbimbing dan klasifikasi tidak terbimbing.

a. Klasifikasi Terbimbing yaitu klasifikasi yang dilakukan dengan arahan analis dimana kriteria pengelompokkan kelas ditetapkan berdasarkan penciri kelas yang diperoleh melalui pembuatan area seperti training area

b. Klasifikasi Tidak Terbimbing merupakan suatu klasifikasi dengan pembentukan kelasnya sebagian besar dikerjakan oleh komputer. Penggunaan Klasifikasi sangat membutuhkan suatu metode yang digunakan untuk mempermudah dalam mengklasifikasi suatu objek atau citra salah satu metode yang digunakan yaitu Jaringan Saraf Tiruan merupakan suatu metode yang melakukan pemerosesan data atau informasi dengan tujuan untuk menirukan cara kerja otak manusia atau berdasarkan prinsip-prinsip kerja otak manusia sehingga mirip dengan jaringan saraf manusia. Dan BP merupakan algoritma yang menggunakan pembelajaran pelatihan untuk mendapatkan output layer dengan akurasi yang baik. BP merupakan metode yang cerdas dan dianggap teknologi yang praktis, dan metode BP melakukan pembelajaran untuk memperkecil tingkat error dengan cara menyesuaikan bobotnya berdasarkan perbedaan output dan target yang diinginkan. BP dikatakan sebagai algoritma pelatihan multilayer karena BP memiliki tiga layer dalam proses pelatihannya, yaitu input layer, hidden layer dan output layer Dengan adanya hidden layer pada BP dapat menyebabkan besarnya tingkat error pada BP lebih kecil dibanding tingkat error pada single layer network. Hal tersebut dikarenakan hidden layer pada BP berfungsi sebagai tempat untuk mengupdate dan menyesuaikan bobot, sehingga didapatkan nilai bobot yang baru yang bisa diarahkan mendekati dengan target output yang diinginkan dengan akurasi

yang baik. Sehingga dengan metode BP banyak digunakan dalam menentukan akurasi dan BP banyak digunakan dalam mengidentifikasi suatu masalah. Pada backpropagatioan terdapat pemilihan bobot jaringan secara acak yang digunakan untuk menghitung koreksi yang diperlukan. Adapun langkah langkah yang dilakukan dalam pelatihan jaringan saaraf tiruan menggunakan *Back Propagation* yaitu langkah Tahap pertama yaitu inisialisasi bobot keterhubungan antara neuron menggunakan bilangan agak kecil, Lalu feedforward dimana Setiap neuron masukan (Xi i= 1…n) menerima sinyal masukan xi dan menyebarkannya ke semua neuron pada lapis tersembunyi.

a. Setiap neuron pada lapis tersembunyi (Zj, j=1…p) menjumlahkan bobot-bobot sinyal masukan dengan persamaan (2.46):

$$z_in_j \sum_{i=1}^{p} x_j v_{ij} \tag{2.46}$$

dan menerapkannya pada fungsi aktivasi untuk menghitung sinyal keluarannya, $z_j = f(z_in_j)$ dan mengirim sinyal ini ke semua neuron pada lapis keluaran.

b. Setiap *neuron* keluaran (Yk, k=1…m) menjumlahkan bobot sinyal yang masuk, persamaan (2.47)

$$y_in_k = \sum_{j=1}^{p} z_j w_{jk} \tag{2.47}$$

dan mengaplikasikan fungsi aktivasinya untuk menghitung sinyal yang dikeluarkannya dengan persamaan (2.48),

$$y_k = f(y_in_k) \tag{2.48}$$

Adapun Langkah backporpagaiton error sebagai berikut :

a. Setiap *neuron* keluaran (Yk, k=1…m) menerima sebuah pola target yang berhubungan dengan pola masukan pelatihan dan menghitung kesalahan

informasi dengan mengalikan dengan turunan fungsi aktivasinya, persamaan (2.49)

$$\delta_k = (t_k - y_k) f'(y_in_k) \tag{2.49}$$

menghitung koreksi bobot (yang digunakan untuk memperbaiki Wjk nanti), dengan persamaan (2.50)

$$\Delta w_{jk} = (\alpha \delta_k z_j) \tag{2.50}$$

dan mengirim k ke *neuron* lapis tersembunyi

b. Setiap neuron tersembunyi (Zj, j=1…p) menjumlahkan bobot setiap neuron yang yang telah dikali dengan kesalahan informasinya, menggunakan persamaan (2.51)

$$\delta_in_j = \sum_{k=1}^{m} \delta_k w_{jk} \tag{2.51}$$

c. mengalikan dengan turunan fungsi aktivasinya untuk menghitung kesalahan informasinya, dengan persamaan (2.52)

$$\delta_j = \delta_in_j f'(z_in_j) \tag{2.52}$$

menghitung koreksi bobotnya (yang digunakan untuk memperbaiki Vij nanti) dengan mengguankan persamaan (2.53) dibawah ini.

$$\Delta v_{ij} = (\alpha \delta_j x_i) \tag{2.53}$$

Selanjutnya yaitu perbaiki bobot untuk proses selanjutnya

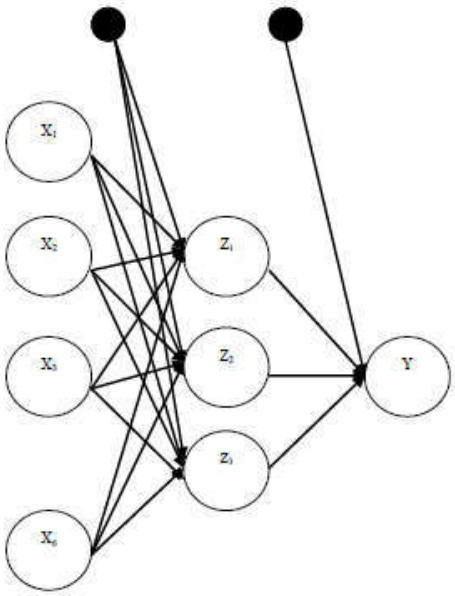
d. Setiap *neuron* keluaran (Yk, k=1…m) memperbaiki bobotnya (j=0…p), dengan persamaan (2.54)

$$w_{jk}(new) = w_{jk}(old) + \Delta w_{jk} \tag{2.54}$$

Setiap *neuron* tersembunyi (Zj, j=1…p) memperbaiki bobotnya dengan persamaan (2.55)

$$v_{ij}(new) = v_{ij}(old) + \Delta v_{ij} \tag{2.55}$$

Adapun arsitektur *Back Propagation* seperti Gambar 2.10. dibawah ini.



Gambar 2.10 Arsitektur Back Propagation

Pada peneliti (Antwi et al., 2017) yang menggunakan metode BP untuk menestimasi hasil biogas dan metana dalam UASB, sehingga BP dapat digunakan untuk memperkirakan hasil biogas dan metana dalam anaerobic naik reactor slude blanket (UASB) hasil dari akurai mencapai 98,72% yang menunjukan kinerja yang signifikan dalam mengontrol proses perencanaan anaeroik

2.6.4. Metode Convolutional Neural Network (CNN)

Convolutional Neural Networks adalah jenis jaringan saraf yang secara biologis terinspirasi dari sistem visual manusia yang menggabungkan tahap

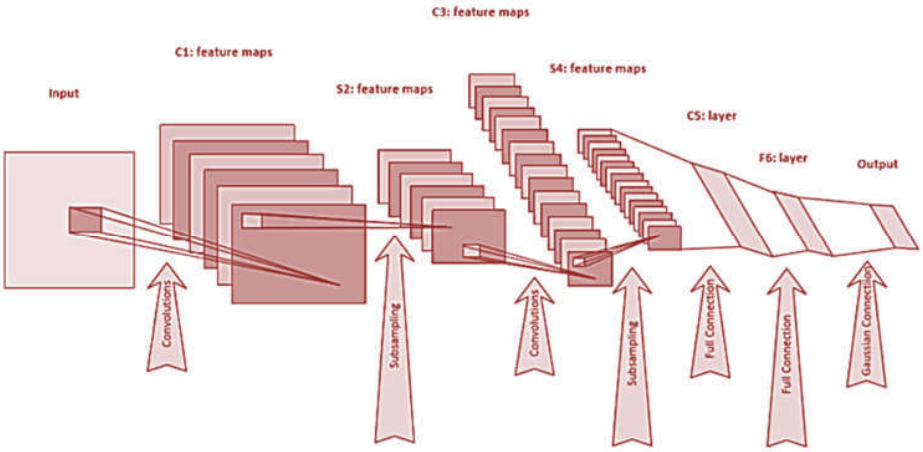
ekstraksi fitur dan klasifikasi. Jaringan saraf konvolusional adalah metode khusus pembelajaran jenis jaringan saraf khusus yang sangat baik disesuaikan dengan aplikasi visi komputer karena kemampuan mereka untuk representasi abstrak hierarkis dengan operasi lokal (Cengil, Cinar, & Guler, 2017). CNN memiliki struktur khusus untuk ekstraksi fitur. Masukan CNN adalah fitur 2-D (gambar), dan hasil klasifikasi dinyatakan dalam bentuk probabilitas. CNN bukan jaringan saraf yang terhubung penuh. Secara luas CNN digunakan dalam pemrosesan gambar, pengenalan pola dan klasifikasi (M. Zhang, Diao, & Guo, 2017). Metode ini memberikan tingkat invarians untuk bergeser, skala dan rotasi sebagai bidang reseptif lokal memungkinkan akses unit neuron atau pemrosesan ke fitur dasar seperti tepi atau sudut yang berorientasi.

Ada 3 jenis lapisan dalam jaringan konvolusional yaitu :

- Convolution (CONV)
- Pooling (POOL)
- Fully connected (FC)

1. LeNet-5

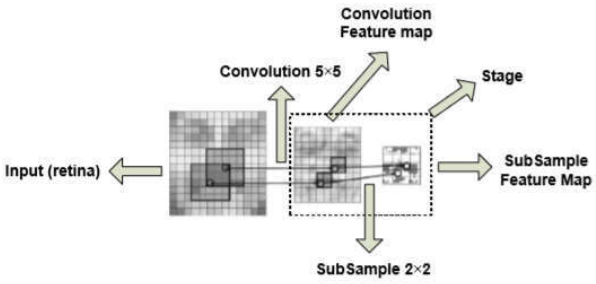
LeNet adalah salah satu jaringan saraf convolutional pertama yang membantu mendorong bidang Deep Learning. Lenet juga pertama kali di perkenalkan oleh (LeCun, Bottou, Bengio, & Haffner, 1998). Arsitektur LeNet disajikan pada Gambar 2.11 berikut(Mohammadi & Mehrolhassani, 1990):



Gambar 2.11 Arsitektur Lenet 5

CNN mengekstrak fitur dalam hierarki dan menciptakan spektrum pola input pada lapisan output. LeNet-5 terdiri dari tujuh lapisan mengecualikan lapisan input. Pad lenet-5 terdapat lima lapisan pertama, C1, S2, C3, S4, C5, merupakan konvolusi yang secara bergantian dan terdapat lapisan sub-sampling dengan 5x5 kernel konvolusi. Lapisan C3 mengikuti skema koneksi non-simetrik khusus. Layer F6 adalah lapisan neuron tambahan yang, sepenuhnya terhubung ke lapisan C5 .

Pada gambar 2.12 dibawah menunjukkan dua lapisan awal yang terkait dengan domain input yang disebut retina. Secara umum, pada akhir setiap lapisan konvolusi lapisan subsampel terpasang.



Gambar 2.12 Peta konvolusi dan subsampel LeNet-5

2. Konvolusi

Konvolusi adalah salah satu blok bangunan dari jaringan Konvolusional. Tujuan utama dari konvolusi dalam kasus CNN adalah untuk mengekstrak fitur dari gambar input. Setiap gambar dapat direpresentasikan sebagai matriks dari nilai piksel. Gambar dari kamera digital standar memiliki tiga saluran yaitu merah, hijau dan biru yang masing-masing memiliki nilai piksel dalam rentang 0 hingga 255. Gambar di bawah menunjukkan penerapan terhadap filter (kernel) untuk mendeteksi tepi vertikal dari gambar asli 2D.

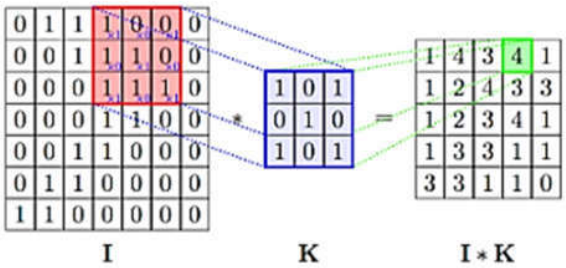
Masukkan dari operator konvolusi diberikan dengan gambar 2 dimensi. Dengan matriks I dan K merupakan yag merupakan kernel konvolusi. Hubungan antara Inputan gambar dan kernel dapat dihitung dengan persamaan (2.56) (Stutz, 2014) :

$$(I * K)_{xy} = \sum_{i=1}^{h} \sum_{j=1}^{w} K_{ij} \cdot I_{x+i-1,y+j-1} \tag{2.56}$$

Dimana filter K diberikan dengan persamaan (2.57):

$$K = \begin{pmatrix} K_{-h_1,-h_2} & \cdots & K_{-h_1,h_2} \\ \vdots & K_{0,0} & \vdots \\ K_{h_1-h_2} & \cdots & K_{h_1,h_2} \end{pmatrix} \tag{2.57}$$

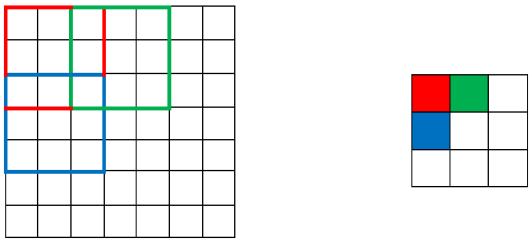
Hasil dari penerapan konvolusinya dapat di lihat pada Gambar 2.13. di bawah ini :



Gambar 2.13 Ilustrasi dari proses konvolusi

Sama seperti Jaringan Saraf lainnya, pada konvolusi juga menggunakan fungsi aktivasi untuk membuat output menjadi non-linear. Dalam kasus Jaringan Saraf Konvolusional, output konvolusi dilewatkan melalui fungsi aktivasi. Fungsi aktivasi yang digunakan yaitu aktivasi ReLU.

Setelah memilih ukuran filter, kita juga harus memilih *Stride* dan *padding*. Stride mengontrol bagaimana filter berputar di sekitar volume input. Gambar 2.14. di bawah merupakan contoh inputan gambar yang memiliki 2 stride, Jumlah yang digunakan untuk menggeser filter adalah Stride. Artinya filter menggeser piksel demi piksel.



Gambar 2.14 Pergeseran stride pada konvolusi

Karena ukuran peta fitur selalu lebih kecil daripada input, kita harus melakukan sesuatu untuk mencegah peta fitur kita menyusut. Di sinilah penerapan paadding dilakukan. Padding menghasilkan konvolusi yang sama seperti pada Gambar 2.15

0	0	0	0	0	0	0	0
0	3	1	1	2	8	4	0
0	1	0	7	3	2	6	0
0	2	3	5	1	1	3	0
0	1	4	1	2	6	5	0
0	3	2	1	3	7	2	0
0	9	2	6	2	5	1	0
0	0	0	0	0	0	0	0

Gambar 2.15 Padding pada konvolusi

Setelah melakukan stride dan padding. Maka output dihasilkan. Untuk menghitung ukuran output dengan cepat caranya mengetahui mengetahui ukuran filter (f), stride (s), padding (p), dan ukuran input (n) dapat dilakukan dengan persamaan (2.58) :

$$Output = \frac{(W-K+2P)}{S} + 1 \qquad (2.58)$$

Dimana,

W : input tinggi/panjang

K : ukuran filter

P : padding

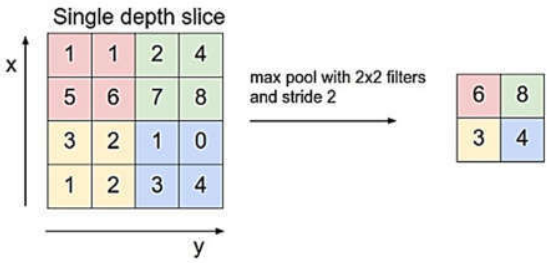
S : stride

3. Pooling

Setelah lapisan konvolusi selanjutnya melakukan pooling. Fungsi pooling adalah untuk terus mengurangi dimensi, mengurangi jumlah parameter dan perhitungan dalam jaringan. Ini berguna untuk memperpendek proses pelatihan dan mengontrol overfitting. Pooling beroperasi dengan menempatkan jendela pada posisi yang tidak tumpang tindih di setiap peta fitur dan menyimpan satu nilai per jendela sedemikian rupa sehingga peta fitur di-subsampel. Pooling di bagi menjadi 2 tipe yaitu :

- Average Pooling yaitu Saat menggunakan filter kotak$_{22}$, operasi ini disebut penggabungan rata-rata dan lapisan yang dilambangkan oleh *PA*.
- Max Pooling yaitu Untuk penggabungan maksimal, nilai maksimum setiap jendela diambil. Lapisan dilambangkan oleh *PM.*

Jenis penggabungan yang paling sering adalah penggabungan max, yang mengambil nilai maksimum di setiap jendela. Ukuran jendela ini perlu ditentukan terlebih dahulu. Ini mengurangi ukuran peta fitur sementara pada saat yang sama menjaga informasi penting seperti yang disajikan pada Gambar 2.16 di bawah ini:



Gambar 2.16 Max Pooling

4. Fully Connected

Lapisan yang terhubung sepenuhnya disemua dengan Fully Connected. Fully Connected berfungsi seperti lapisan jaringan neural tunggal standar, di mana

lapisan tersebut memiliki matriks berat W dan bias b. Jika tidak terhubung secara sepenuhnya, layer diberikan $m_1^{(l-1)}$ feature map dengan ukuran $m_2^{(l-1)} x\, m_3^{(l-1)}$ sebagai input dan unit i^{th} pada layer dapat dihitung dengan persamaan (2.59) :

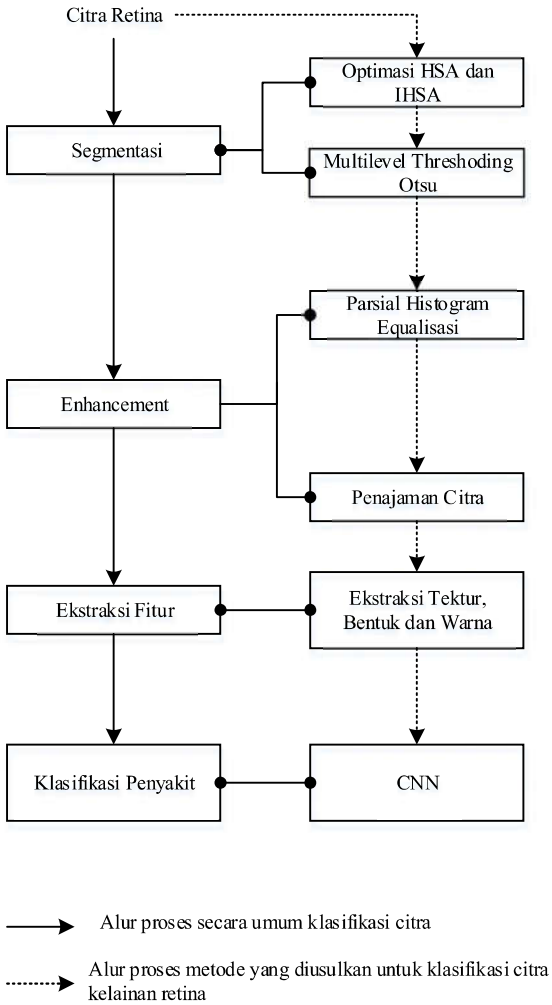
$$m_1^{(l)} = f z_i^{(l)} \qquad with\ z_i^{(j)} = \sum_{j=1}^{m_1^{(l-1)}} \sum_{r=1}^{m_2^{(l-1)}} \sum_{s=1}^{m_3^{(l-1)}} w_{i,j,k}^{(l)} \left(Y_j^{(l-1)}\right)_{r,s} \qquad (2.59)$$

Dimana $w_{i,j,k}^{(l)}$ menunjukkan berat yang menghubungkan unit pada posisi (r,s).

BAB III. METODOLOGI PENELITIAN

3.1. Pendahuluan

Secara garis besar bagan alur penelitian untuk membangun model optimasi teknik klasifikasi penyakit kelainan retina dengan harmony search algorithm dapat diperlihatkan pada Gambar 3.1.



Gambar 3.1 Bagan Alur Teknik Klasifikasi Penyakit Kelainan Retina

Pendekatan metode segmentasi yang dikembangkan akhir-akhir ini menunjukan peningkatan, walaupun masih ada kelemahaan. Teknik baru yang diusulkan

menggunakan pendekatan optimasi, dan multilevel untuk memisahkan foreground sebagai pembuluh darah, exudates, fovea dan mokula. Setelah foreground telah dapat dipisahkan melalui proses segmentasi, langkah berikutnya peningkatan foreground untuk proses enhancement. Setelah foreground disegmentasi dari background melalui proses segmentasi, nilai intensitas dilakukan penyesuaian secara otomatis secara parsial dengan menggunakan pemerataan histogram kemudian teknik penajaman citra dilakukan untuk meningkatkan aspek frekuensi tinggi pada citra, seperti tepi di sekitar fitur utama citra, sangat diharapkan untuk dipertajam untuk memperbaiki tampilan visual citra.

Selanjutnya, teknik ekstraksi fitur diterapkan untuk memperoleh ciri-ciri yang berkaitan dengan tesktur, bentuk dan warna dari pembuluh darah, exudates, fovea dan mokula. Akhirnya, proses klasifikasi penyakit secara otomatis dilakukan dengan metode CNN.

3.2. Dataset Citra Retina

DRIVE singkatan dari Digital Retinal Images for Vessel Extraction merupakan dataset segmentasi pembuluh darah di citra retina merupakan publikasi dari Image Sciences Institute (ISI) Univeritas Utrech. Pengamatan terhadap 400 pasien diabetes menghasilkan 40 citra pembuluh darah di retina yang dibagi menjadi 20 citra latih dan 20 citra uji.

STARE singkatan dari Structured Analysis of the Retina merupakan proyek penelitian yang dimulai tahun 1975 di University of California. Proyek ini dilakukan oleh ahli ophthalmologist yaitu dokter spesialis medis bidang struktur, fungsi dan penyakit pada mata manusia. Salah satu hasil pengujian secara klinik yaitu mempelajari pembuluh darah di retina dan diagnosa penyakit pasien. Dataset yang tersedia terdiri dari 400 citra retina, daftar kode diagnose, diagnose masing-masing citra dan hasil segmentasi pembuluh darah dan optic disk. Secara ringkas disajikan pada Tabel 3.1.

Tabel 3 1. Dataset Sekunder yang Digunakan

No.	Nama Dataset	Sumber	Jumlah Citra
1.	DRIVE	https://www.isi.uu.nl/Research/Databases/DRIVE/	40 Citra pembuluh darah di retina dari 400 pasien diabetes
2.	STARE	http://cecas.clemson.edu/~ahoover/stare/	400 Citra retina

Citra digital yang diambil oleh kamera retina Con-TRV-50 dengan sudut pandang 35 derajat. Setiap slide didigitalkan untuk membuat citra 605x700 *pixel* dengan 24 bit per *pixel*. Ada 13 kelas penyakit untuk tiap-tiap sampel citra retina dalam dataset. Namun, hanya 5 kelas penyakit untuk 149 sampel retina yang digunakan oleh penulis dalam penelitian ini. Sampel citra retina yang diberikan menunjukkan retina dengan kondisi pembuluh darah abnormal, berikut sampel retina STARE yang disajikan pada Gambar 3.2. a. Central Retinal Vein Occlusion, b. Retinitis, c. Background Diabetic Retinopathy, c. Proliverative Diabetic Retinopathy dan e. Coats

a **b** **c** **d** **e**

Gambar 3 2. Citra Retina dengan Penyakit Kelainan Retina

Tabel 3.2 dan 3.3 merupakan pembagian dataset yang digunakan sebagai data latih dan data uji untuk setiap penyakit. Ada sebanyak 112 data retina yang digunakan sebagai data latih dan data pengujian sebanyak 5 data untuk setiap masing-masing kelas. Tabel 3.2 merupakan pembagian dataset Stare untuk data latih dan data uji untuk masing-masing kelas penyakit citra untuk Ekstraksi Fitur Berdasarkan Tekstur, Warna dan Klasifikasi dengan SOM, BP dan CNN. Tabel 3.3 merupakan pembagian dataset Stare untuk data latih dan data uji untuk masing-masing kelas

penyakit citra untuk Multitrhresholding HSA, IHSA, Ekstraksi Fitur Berdasarkan Bentuk dan Klasifikasi MLP

Tabel 3 2. Pembagian dataset Stare untuk data latih dan data uji

No	Kelas	Banyak Data Latih	Banyak Data Uji
1	BDR	42	5
2	CRVO	11	5
3	Coats	8	5
4	Normal	35	5
5	PDR	16	5
Jumlah		112	25

Tabel 3 3. Pembagian dataset Stare untuk data latih dan data uji

No	Kelas	Banyak Data Latih	Banyak Data Uji
1	CNV	26	3
2	BRVO	9	2
3	Histoplasmosis	10	2
Jumlah		45	7

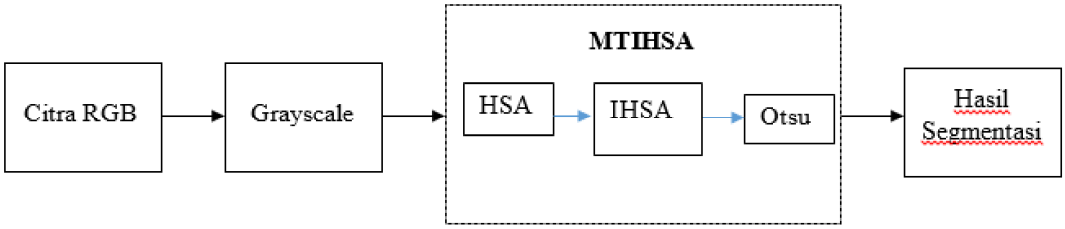
3.3. Metode Segmentasi

Segmentasi adalah proses untuk memisahkan citra menjadi beberapa bagian, perubahan ini digunakan untuk mendapatkan informasi baru yang lebih bermanfaat dan lebih mudah dalam analisis berikutnya(Dass & Devi, 2012). Proses segmentasi merupakan tahapan yang penting dalam keberhasilan analisis citra. Tingkat akurasi yang tinggi pada tahap segmentasi menghasilkan hasil analisis yang lebih baik (Gonzalez & Woods, 2007).

Segmentasi menggunakan teknik multilevel tresholding merupakan metode yang bekerja sangat baik dalam menentukan segmentasi dengan background yang kompleks (Yourui & Shuang, 2008). Teknik ini membagi citra menjadi beberapa bagian sehingga cocok diterapkan pada citra retina yang memiliki background yang komplek seperti pembuluh darah yang halus, dan titik titik sebagai ciri kelainan retina.

Segmentasi dengan multilevel thresholding memerlukan waktu dan melibatkan perhitungan yang besar, untuk mengatasi masalah tersebut, beberapa metode pengoptimalan diterapkan (Chen et al., 2016). Selain itu, alasan utama penggunaan pendekatan optimasi dikarenakan citra yang diteliti merupakan citra yang kompleks dan memerlukan akurasi yang tinggi sehingga dengan melakukan proses yang berulang-ulang dalam menemukan foreground secara optimal. Dalam penelitian terdahulu, multilevel thresholding dengan HSA digunakan dalam mensegmentasi citra dengan dua metode thresholding, yakni Otsu dan Kapur. Multilevel thresholding dengan Otsu yang telah dioptimalkan menggunakan HSA menunjukkan hasil yang lebih baik dibandingkan dengan Kapur (Oliva et al., 2013). Penelitian ini menggunakan teknik baru yaitu hybrid multilevel thresholding Otsu dan IHSA yang dinamakan MTIHSA untuk mengsegmentasi citra kelainan retina.

Perancangan metode yang dilakukan dalam penelitian ini terdiri dari beberapa tahap sehingga pergerjaannya menjadi terstuktur. Tahapan kerja disajikan dalam bentuk blok diagram yang dapat dilihat pada Gambar 3.3. Misalkan fungsi intensitas piksel citra RGB dinyatakan sebagai $f(x, y)$, kemudian dilakukan transformasi menjadi citra grayscale dinyatakan sebagai $f'(x, y)$ dan hasil segmentasi dengan MTIHSA dinyatakan sebagai $f''(x, y)$.



Gambar 3.3. Blok Diagram Proses Segmentasi

Bentuk umum masalah optimasi dinyatakan persamaan (3.1) sebagai berikut:

$$fungsi\ objektif: maks \quad f(x), x = [x(1), x(2), \dots, x(n)] \in R^n, \tag{3.1}$$

$$fungsi\ kendala\ x(j) \in [l(j), u(j)], j = 1,2, \dots, n$$

Dengan *f(x)* merupakan fungsi objektif, x_j adalah himpunan variabel acak yang merupakan solusi dari fungsi objektif, *n* merupakan jumlah variabel, sedangkan l_i dan u_i merupakan batas bawah dan batas atas variabel acak.

Parameter-parameter dalam IHSA yaitu vector solusi dinyatakan sebagai harmoni memori(HM), harmoni memori consideration rate(HMCR), pitch adjusting rate (PAR), distance bandwidth(BW) dan jumlah iterasi/generasi(NI).

Misalkan $x_i = [x_i(1), x_i(2), \dots, x_i(n)]$ merupakan vector acak dari harmoni memori yang diperoleh dari $x_i(j) = l(j) + (u(j) - l(j).rand(0,1)$, untuk $j = 1,2,\dots,n$ dan $i = 1,2,\dots,HMS$ serta *rand*(0,1) merupakan bilangan acak berdistribusi uniform, bernilai antara 0 dan 1. HM merupakan vector solusi dengan bentuk matrik persamaan (3.2) sebagai berikut:

$$HM = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{HMS} \end{bmatrix} \tag{3.2}$$

Perhitungan x_{baru} dengan model persamaan (3.3) yang dinyatakan dengan:

$$x_{baru}(j) = \begin{cases} x_i(j) \in [x_1(j), x_2(j), \dots, x_{HMS}(j)], & dengan\ peluang\ HMCR \\ l(j) + (u(j) - l(j).rand(0,1), & dengan\ peluang\ 1-HMCR \end{cases} \tag{3.3}$$

Setiap komponen dari HMCR disesuaikan dengan menghitung PAR dan BW sehingga nilai x_{baru} menjadi persamaan (3.4):

$$x_{baru}(j) = \begin{cases} x_{baru}(j) = x_{baru}(j) \pm rand(0,1).BW, & dengan\ peluang\ PAR \\ x_{baru}(j), & dengan\ peluang\ 1-PAR \end{cases} \tag{3.4}$$

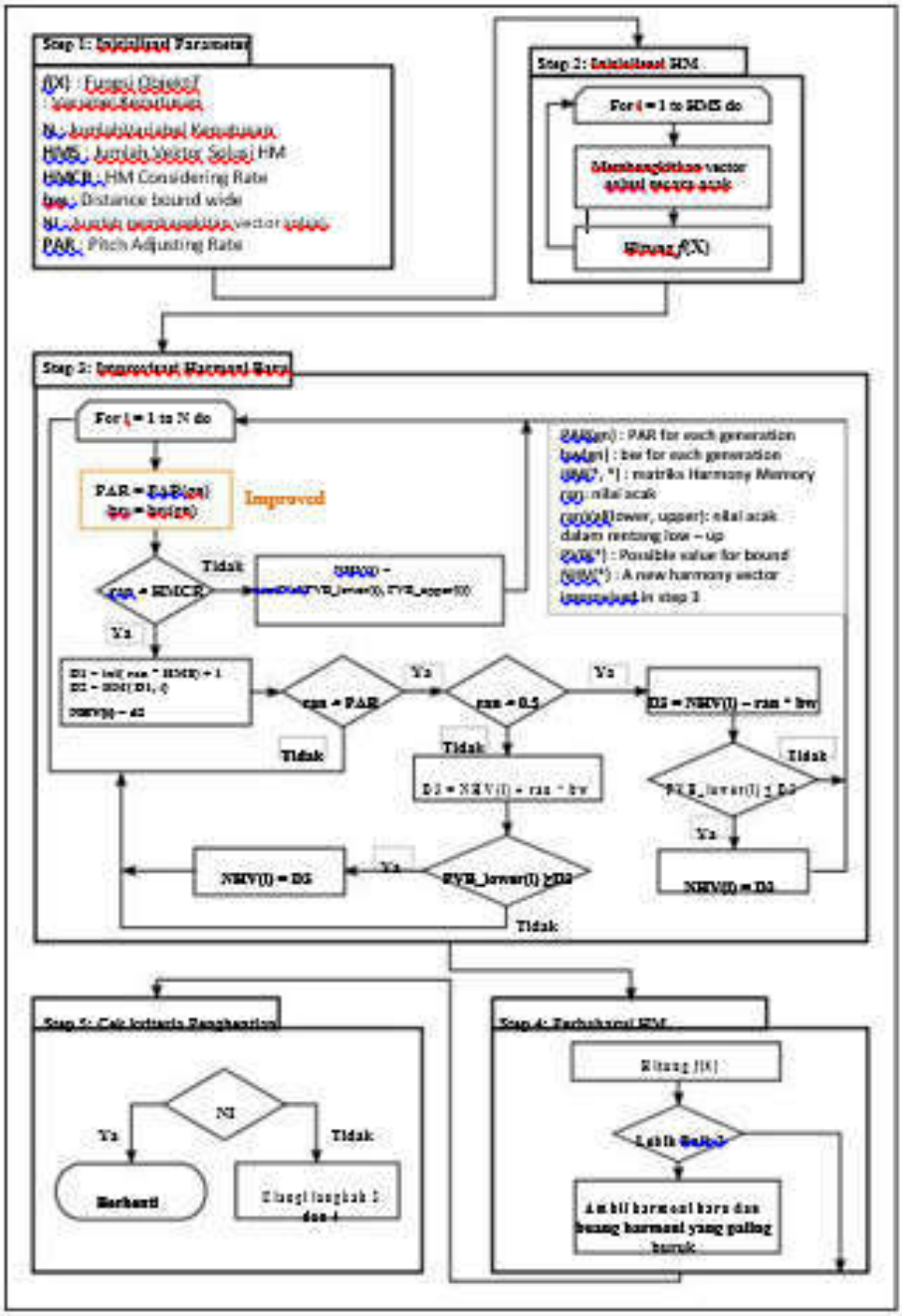
3.3.1. Teknik Hybrid Multilevel Thresholding Otsu dan Harmony Search Algorithm (MTHSA)

Hybrid metode multithresholding dan harmoni search ditunjukkan pada Gambar 3.4. Fase awal adalah membaca dan menyimpan citra dalam skala abu-abu. Prosedur lebih lanjut algoritma harmoni search dijalankan menjadi nilai thresholding Otsu, kemudian nilai harmonisasi algoritma Otsu dioptimalkan dengan mencari dan mengembalikan nilai optimasi untuk multithresholding.

Thresholding merupakan suatu proses di mana piksel dari citra abu-abu dibagi dalam set atau kelas tergantung pada tingkat intensitasnya (L). Misalkan nilai ambang (th) dengan multilevel dengan himpunan persamaan (3.5) sebagai berikut:

$$\begin{aligned} C_1 &\leftarrow p \quad if\ 0 \leq p < th_1, \\ C_2 &\leftarrow p \quad if\ th_1 \leq p < th_2, \\ C_i &\leftarrow p \quad if\ th_i \leq p < th_{i+1}, \\ C_n &\leftarrow p \quad if\ th_n \leq p < L-1, \end{aligned} \tag{3.5}$$

di mana $\{th_1, th_2, \dots, th_i, th_{i+1}, th_k\}$ mewakili ambang berbeda. Otsu merupakan varians maksimum antar klaster. Kriteria seleksi didapatkan dari histogram citra sebagai basis dan varians maksimum antara target dan latar belakang.



Gambar 3 4. Diagram Alur Metode MTHSA

3.3.2. Teknik Hybrid Multilevel Thresholding Otsu dan IHSA(MTIHSA)

Metode ini merupakan gabungan antara dua metode multilevel thresholding dengan Otsu dan algoritma pengembangan harmony search algorithm (MTIHSA). Metode yang diusulkan mencari sampel acak dalam histogram sebagai kandidat, kemudian mengevaluasi kualitasnya dengan menggunakan fungsi objektif oleh Otsu. Selanjutnya, operator IHSA melakukan evolusi pada kandidat sampai solusi yang paling optimal ditemukan.

Tabel 3.4 di bawah ini merupakan parameter IHSA yang digunakan dalam penelitan ini:

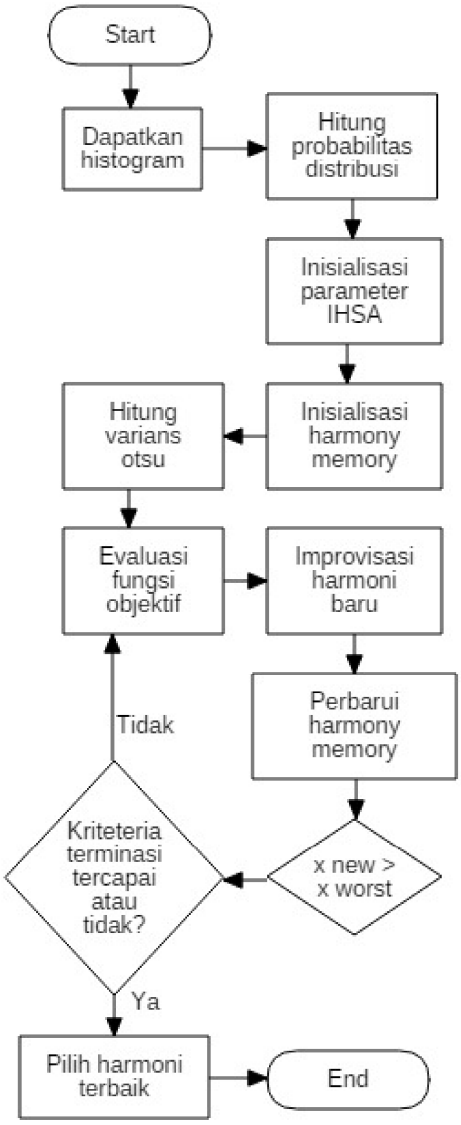
Tabel 3.4. Parameter Yang Digunakan

NI	HMS	HMCR	PAR Min	PAR Max	BW Min	BW Max
2.000	5	0,9	0,01	0,99	0,001	0,1

Kode Semu untuk Algoritma MTIHSA

$Step$ 1.Obtain histograms.
$Step$ 2.Calculate the probability distribution.
$Step$ 3.Initialize the IHSA parameters: HMS, HMCR, c, PAR_{max}, BW_{min}, BW_{max} , and NI.
$Step$ 4.Initialize a Harmony Memory (HM).
Step 5. Calculate Otsu variance.
$Step$ 6.Evaluate objective function
$Step$ 7.Improvise a new harmony.
$Step$ 8.Update the HM
$Step$ 9.If NI is completed or the stop criteria is satisfied, then jump to Step 10, otherwise go back to Step 6.
$Step$ 10.Select the harmony that has the best objective function value.
$Step$ 11.Apply the best thresholds values to the image.

Diagram alir *multilevel thersholding* berbasis IHSA dapat dijelaskan pada Gambar 3.5 berikut:



Gambar 3 5. Diagram Alur MTIHSA

3.3.3. Kriteria Pengukuran Peak Signal Noise Ratio (PSNR)

Peak Signal to Noise Ratio (PSNR) adalah perbandingan antara nilai maksimum dari sinyal yang diukur dengan besarnya derau yang berpengaruh pada sinyal tersebut.

PSNR biasanya diukur dalam satuan decibel (db). PSNR digunakan untuk mengetahui perbandingan kualitas citra sebelum di segmentasi dan sesudah di segmentasi. Untuk menentukan PSNR, terlebih dahulu harus ditentukan nilai MSE (Mean Square Error).

MSE adalah nilai error kuadrat rata-rata antara citra sebelum di segmentasi dengan citra sesudah di segmentasi. PSNR dan MSE didefinisikan dalam persamaan (3.6)-(3.8) berikut.

$$MSE = \frac{\sum_{x=1}^{M}\sum_{y=1}^{M}(S_{xy}-C_{xy})^2}{MN} \quad (3.6)$$

$$RMSE = \sqrt{MSE} \quad (3.7)$$

$$PSNR = 20 \times log_{10}\left(\frac{255}{RMSE}\right) \quad (3.8)$$

Keterangan :
x dan y = koordinat
M dan N = dimensi
Sxy = Citra asli
Cxy = Citra segmentasi

PSNR umumnya memiliki satuan decibel karena dinyatakan dalam skala logaritmik. Jika nilai PSNR berada dibawah 30 dB maka diindikasikan sebagai kualitas rendah sedangkan nilai diatas 40 dB dinyatakan sebagai kualitas tinggi (Cheddad, Condell, Curran, & McKevitt, 2010). Nilai PSNR umumnya antara 30 dB sampai 50 dB untuk perbaikan citra dan kompresi video sedangkan pada kualitas transmisi pada jaringan nirkabel antara 20 dB sampai 25 dB.

3.4. Metode Enhancement: Parsial Foreground Histogram Equalization

Proses penyesuaian nilai intensitas dapat dilakukan secara otomatis dengan menggunakan pemerataan histogram. Pemerataan histogram melibatkan perubahan

nilai intensitas sehingga histogram citra keluaran kira-kira sesuai dengan histogram yang ditentukan.

Teknik ini sering digunakan untuk mengolah histogram yaitu histogram equalization (HE) yang menghasilkan histogram yang seragam atau merata sehingga sering juga disebut histogram merata (Gonzalez & Woods, 2007). Teknik ini dapat dilakukan untuk keseluruhan daerah citra(*global histogram equalization*) atau dengan beberapa kali diulang untuk setiap blok gambar (sub-image). Teknik yang diusulkan melakukan pemerataan histogram secara parsial untuk foreground yang menjadi fokus ciri suatu kelas penyakit yang dinamakan Parsial Foreground Histogram Equalization.

Persamaan (3.9) digunakan untuk melakukan proses pemerataan histogram.

$$H(b_k) = \sum_{i}^{k} \frac{s_i}{s} = \sum_{i}^{k} p_{k(b_k)}, \quad \begin{array}{l} 0 \le b_k \le 1 \\ k = 0,1,\ldots, L-1 \end{array}$$

$$b_k = \frac{k}{L-1}, 0 \le k \le L-1 \tag{3.9}$$

Langkah-langkah Penggunaan Parsial Foreground Histogram Equalization

1. Hitung Nilai citra tingkat abu-abu pada foreground.
2. Hitung nilai maksimum hasil tingkat abu-abu citra. Histogram menentukan pemerataan histogram.
3. Dari histogram citra asli, Simpan derajat frekuensi abu-abu dalam array (vektor). Set array (vektor) yang ukurannya tergantung pada nilai derajat maksimum. Lalu, menghitung derajat abu-abu pada frekuensi vektor posisi.
4. Buat pemerataan histogram. Pemerataan histogram diperoleh dengan menghitung derajat kemunculan abu-abu foreground yang dikalikan dengan tingkat abu-abu maksimum dari citra asli.
5. Temukan nilai gray level dari hasil citra baru dengan menggunakan histogram equalization.
6. Untuk memetakan pemerataan histogram citra menjadi yang baru, matriks nol sama dengan ukuran citra aslinya. Nilai matriks berdasarkan citra.

3.5. Metode Ekstraksi Fitur

3.5.1. Ekstraksi Fitur Retina menggunakan Tekstur

Ekstraksi fitur adalah bagian paling penting dari suatu aplikasi pengenalan pola. Proses ekstraksi fitur dilakukan untuk mendapatkan nilai yang merupakan ciri dari citra retina tersebut. Tahap ekstraksi fitur tekstur pada penelitian ini adalah *Gray Level Co-occurrence Matric (GLCM)*. Berikut merupakan tahapan yang digunakan dalam pengambilan fitur tekstur GLCM dari citra retina.

1. Nilai pixel pada sebuah citra retina yang sudah di perbaiki kualitasnya (*image enhancement*), terlebih dahulu di quantization untuk diproses co–occurrence matrix. Quantization merupakan pengelompokan nilai pixel pada tingkatan level yang telah ditentukan. Pada sebuah citra terdiri dari 256 variasi warna yang di bagi dalam 8 variasi seperti ditunjukkan pada Tabel 3.5 sebagai berikut :

Tabel 3.5. Kuantisasi 8 variasi

Level	Nilai
0	0 – 31
1	32 – 63
2	64 – 95
3	96 – 127
4	128 – 159
5	160 – 191
6	192 – 223
7	224 – 256

2. Buatlah 4 matriks GLCM untuk 4 orientasi berbeda masing-masing untuk $\theta = 0°$, $\theta = 45°$, $\theta = 90°$, dan $\theta = 135°$, dengan radius $\delta = 1$ (d adalah jarak antara dua piksel). Dengan demikian, dapat diperoleh $GLCM_{0,}$ $GLCM_{45,}$ $GLCM_{90,}$ $GLCM_{135.}$
3. Dari setiap GLCM matriks yang dikonstruksi pada langkah 2 ekstraksi sifat-sifat seperti kontras (C), korelasi (CO), energi (E), dan homogeneitas (H).
4. Buat fitur vektor dari properti ini dengan mengambil rata-rata setiap properti yang ditunjukkan pada Tabel 3.6.

Tabel 3.6. Fitur tekstur hasil GLCM

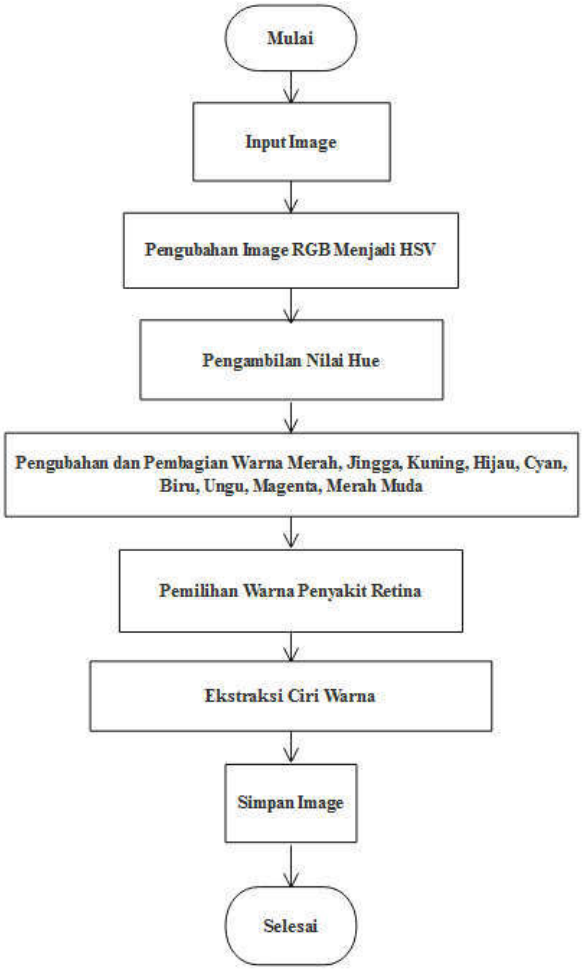
C_0	CO_0	H_0	E_0
C_{45}	CO_{45}	H_{45}	E_{45}
C_{90}	CO_{90}	H_{90}	E_{90}
C_{135}	CO_{135}	H_{135}	E_{135}
Avg C	Avg CO	Avg H	Avg E

3.5.2. Ekstraksi Fitur Berdasarkan Bentuk

Ekstraksi fitur bentuk dilakukan dengan cara mengambil nilai dari ciri yang tampak pada retina secara kasat mata, mampu dilihat dengan menggunakan mata manusia atau biasa juga disebut ciri morfologi retina. Tahap ekstraksi fitur kali ini bukan terbatas pada ciri morfologi retina saja tetapi memanfaatkan ciri morfologi tersebut untuk melakukan perhitungan agar bisa mendapatkan fitur kelainan retina yang mampu membuat perbedaan dari kelainan retina yang lainnya.

3.5.3. Ekstraksi Fitur Berdasarkan Warna

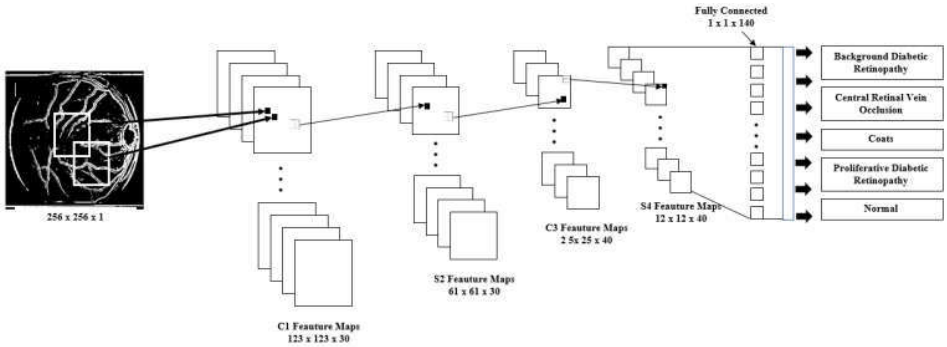
Flowchart untuk ekstraksi fitur berdasarkan warna disajikan pada Gambar 3.6 di bawah ini:



Gambar 3 6. Diagram Alir Klasifikasi Ciri Warna

3.6. Klasifikasi Penyakit Kelainan Retina dengan CNN

Gambar 3.7 di bawah ini merupakan arsitektur lenet-5 untuk membuat model dari klasifikasi CNN.

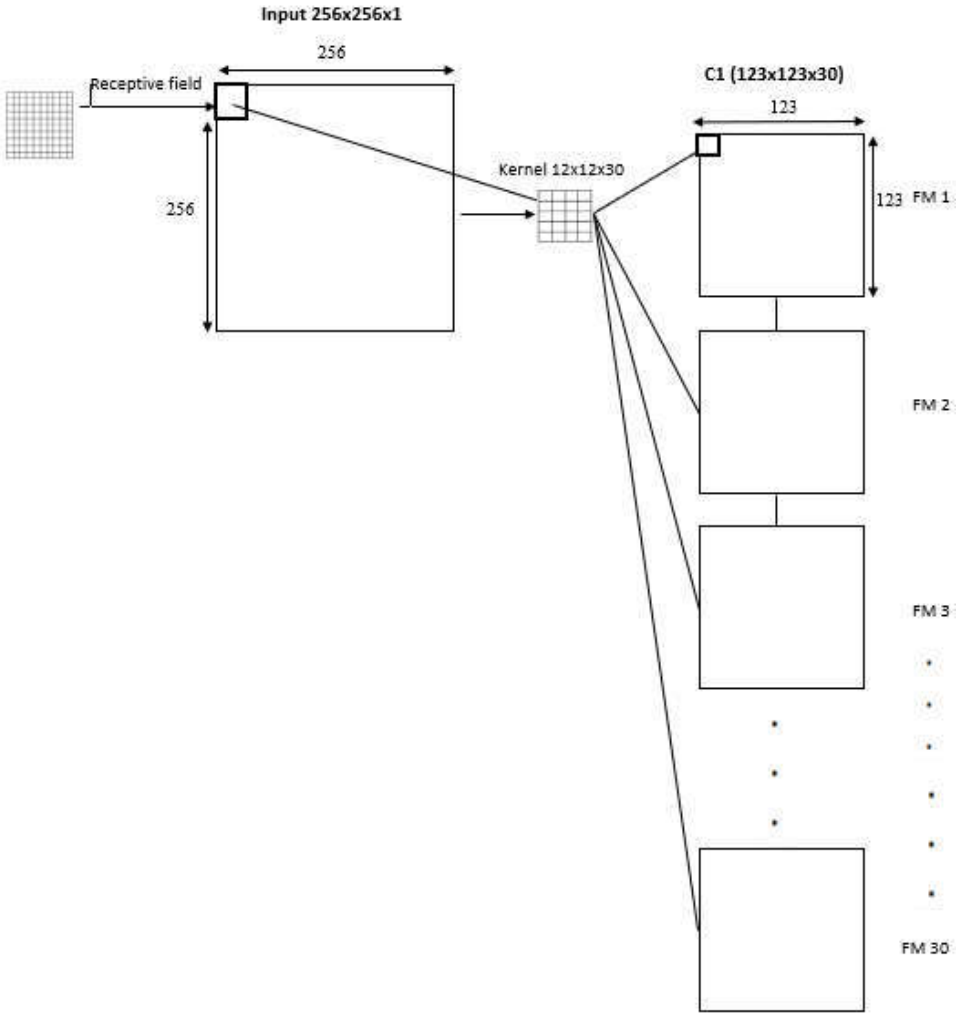


Gambar 3.7. Arsitektur CNN dengan menggunakan Lenet-5

Arsitektur lenet-5 yang dibuat terdapat 5 lapisan dan 1 fully connected. Secara detail berikut penjelasan dari setiap lapisan dari lenet-5 :

a. Lapisan konvolusi 1 (C1)

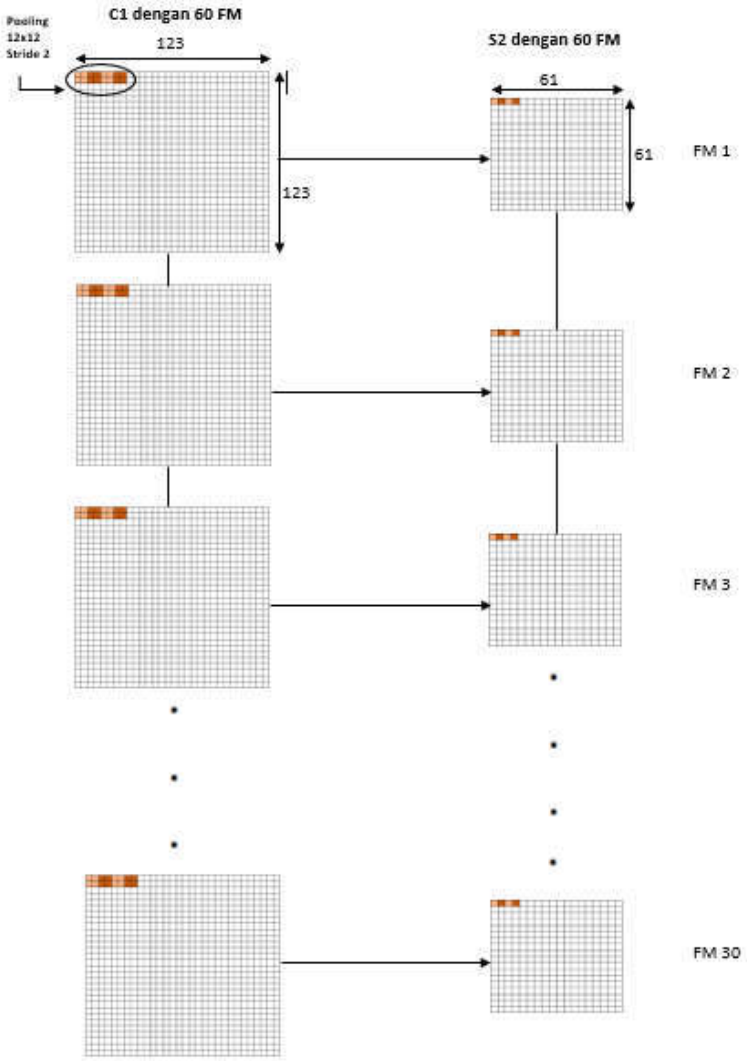
Pada lapisan C1 pixel yang dimasukkan adalah 256x256x1. Setiap feature map memiliki 12x12 kernel untuk dimasukkan pada citra 256x256 secara tumpang tindih dengan 1 bias. Pada lapisan C1 menghasilkan layer konvolusi dengan pixel 123x123 dengan 2 stride. Pada lapisan C1 jumlah feature mapnya ada 30. 1 feature map terdapat 12x12 kernel. Sehingga paramater yang dapat dilatih dapat dihitung (12x12+1bias)x30FM menghasilkan 4350 parameter yang dapat dilath. Koneksi yang terhubung juga dapat dihitung dengan 123x123x(12x12+1bias)x30 menghasilkan 65.811.150 koneksi seperti yang disajikan pada Gambar 3.8.



Gambar 3.8. Visualisasi C1

b. Lapisan Subsampling (Pooling) S2

Pada lapisan subsampling (pooling layer) feature map yang diberikan juga sama yaitu 30 feature map. Pada s1 area receptive field tidak saling tumpang tindih. Karena dilakukan stride (jumlah pergeseran) sebanyak 2 sedangkan untuk pooling layernya menggunakan max pooling 12x12. Sehingga menghasilkan pixel 61x61. Setiap feature map pada lapisan subsampling terdapat 1 bias dan juga 1 bobot. Parameter yang dapat dilatih pada S2 dapat dihitung 30(1+1) = 60 parameter yang dapat dilatih dan memiliki 61x61x(12x12+1)x30 = 16.186.350 koneksi seperti yang disajikan pada Gambar 3.9.

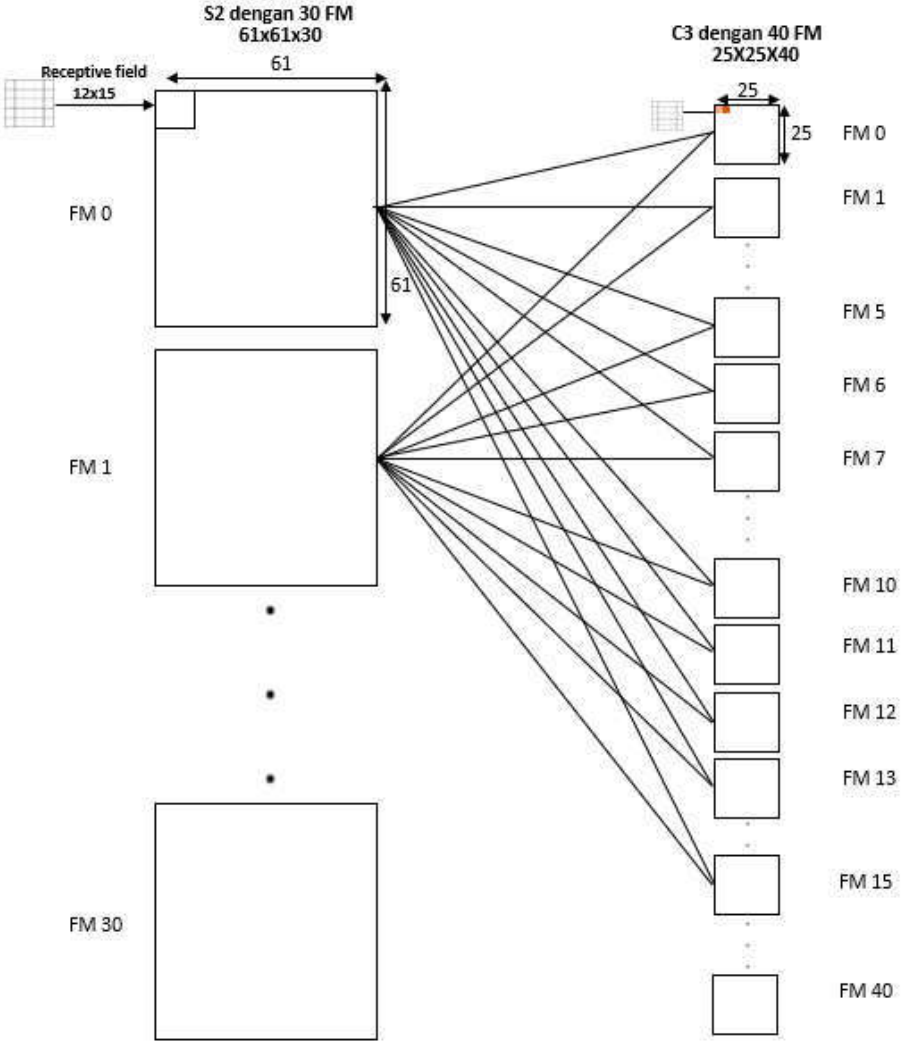


Gambar 3.9. Visualisasi Sum Sampling (Pooling) S2

c. Lapisan konvolusi 3 (C3)

Lapisan berikutnya yaitu lapisan konvolusi dengan 40 kernel yang berbeda. setap neuron dari FM terhubung kelapisan subsampling S2 kebeberapa area. Karena skema koneksi yang tdak lengkap menjaga jumlah koneksi yang saling berhubungan. Yang lebih pentingnya lagi, Peta fitur yang berbeda dipaksa untuk mengekstraksi fitur yang berbeda (komplementer) karena mereka mendapatkan set

input yang berbeda. jadi, setiap lapisan S2 menuju kelapisan S3 dengan 10 koneksi saja seperti yang disajikan pada Gambar 3.10.



Gambar 3.10. Visualisasi Konvolusi C3

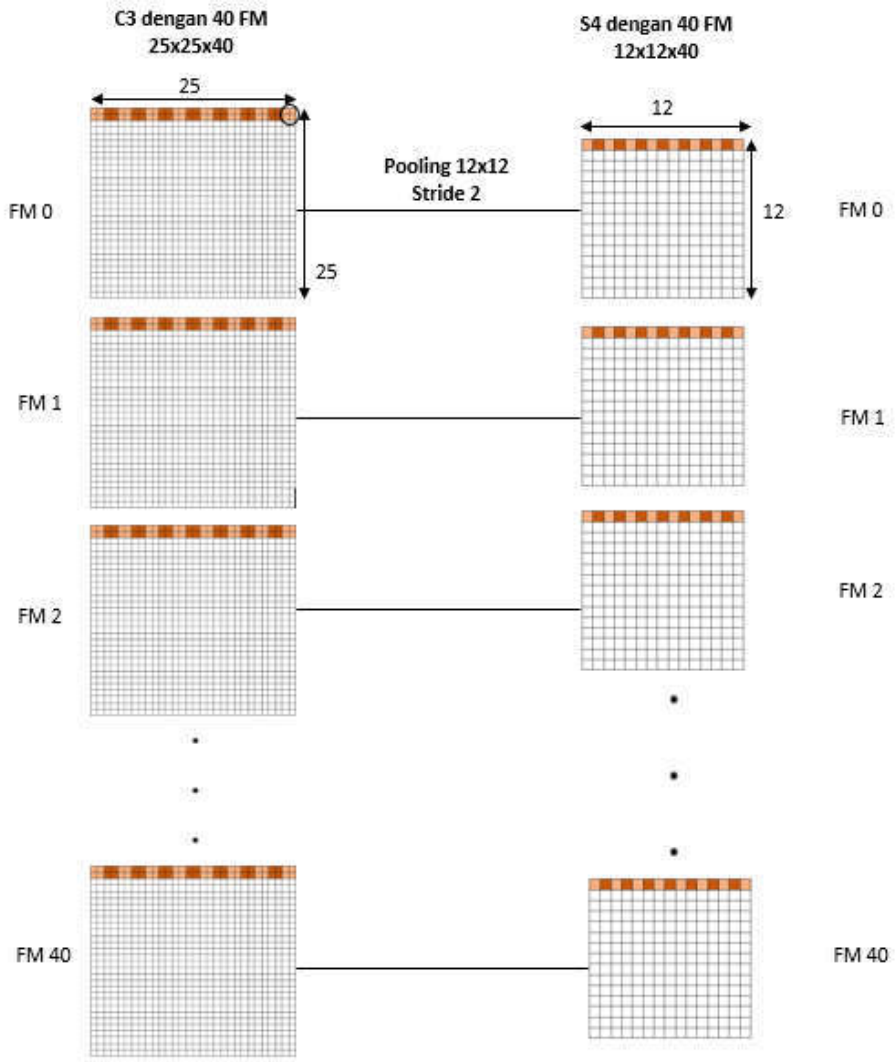
Tabel 3.7 menunjukkan bagaimana peta fitur lapisan C3 terhubung ke peta fitur lapisan S2.

Tabel 3.7. Feature Maps Connection Antara C3 Dan S2

Features Maps in S2	Features Maps in C3															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	x	-	-	-	x	x	x	-	-	x	x	x	x	-	x	X
1	x	x	-	-	-	x	x	x	-	-	x	x	x	x	-	x
2	x	x	x	-	-	-	x	x	x	-	-	x	-	x	x	x
3	-	x	x	x	-	-	x	x	x	x	-	-	x	-	x	x
4	-	-	x	x	x	-	-	x	x	x	x	-	x	x	-	x
5	-	-	-	x	x	x	-	-	x	x	x	x	-	x	x	x

d. Lapisan Subsampling (Pooling) S4

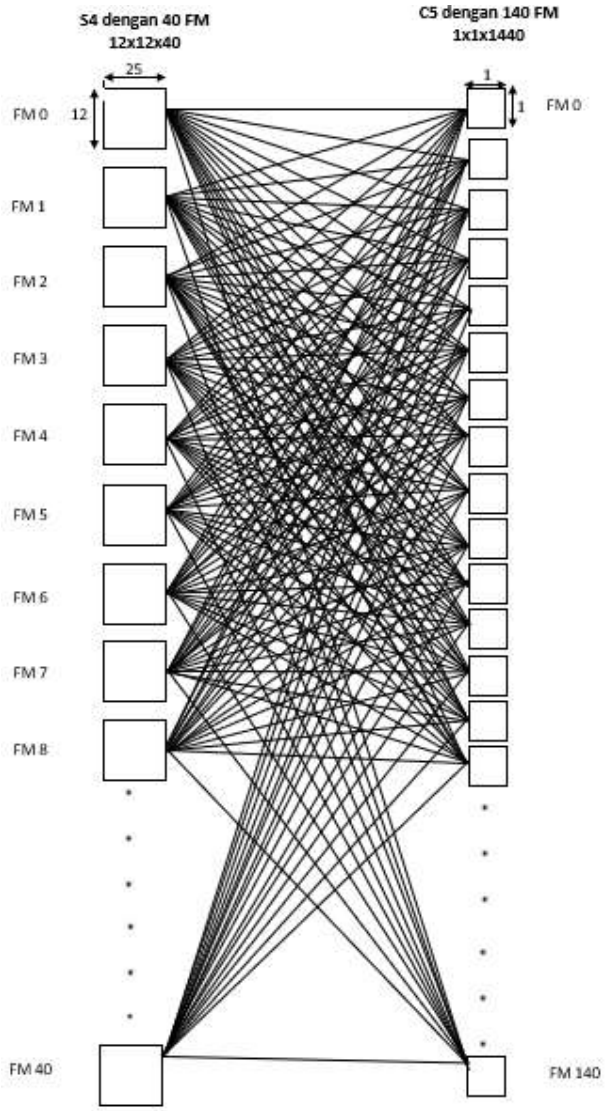
Lapisan berikutnya adalah S4 yang terdiri dari 40 FM juga dengan ukuran 12x12 kernel. Layer s4 memiliki 1 bobot dan 1 bias. Sehingga secara keseluruhan s4 memiliki 40(1+1) = 80 parameter yang dalat dilatih. Pooling ayng diberikan juga sama yaitu 2x2 dengan stride 2. Koneksi pada S4 dapat dihitung dengan 12x12x(12x12+1)x40 sehingga koneksi yang dimiliki sebanyak 835200 koneksi, seperti yang disajikan pada Gambar 3.11.



Gambar 3.11. Visualisasi Subsampling (Pooling) S4

e. Lapisan Konvolusi 5 (C5)

Lapisan yang terhubung penuh ini memiliki 1x1x140 neuron degan total keseluruhan sebanyak 140 neuron. Parameter dapat dihitung dengan 140x(40x12x12+1) sehingga menghasilkan 5744 paramter yang dapat dilatih dengan koneksi penuh, seperti yang disajikan pada Gambar 3.12.



Gambar 3.12. Visualisasi Konvolusi C5

f. Fully Connected (F6)

Layer F6 merupakan lapisan terakhir yang terhubung secara penuh yang memiliki 84 neuron. Oleh karena itu jumlah paremeternya dapat dihitung 84x(140+1) sehingga menghasilkan 11844 parameter pelatihan dan koneksi.

BAB IV. HASIL DAN PEMBAHASAN

4.1. Pendahuluan

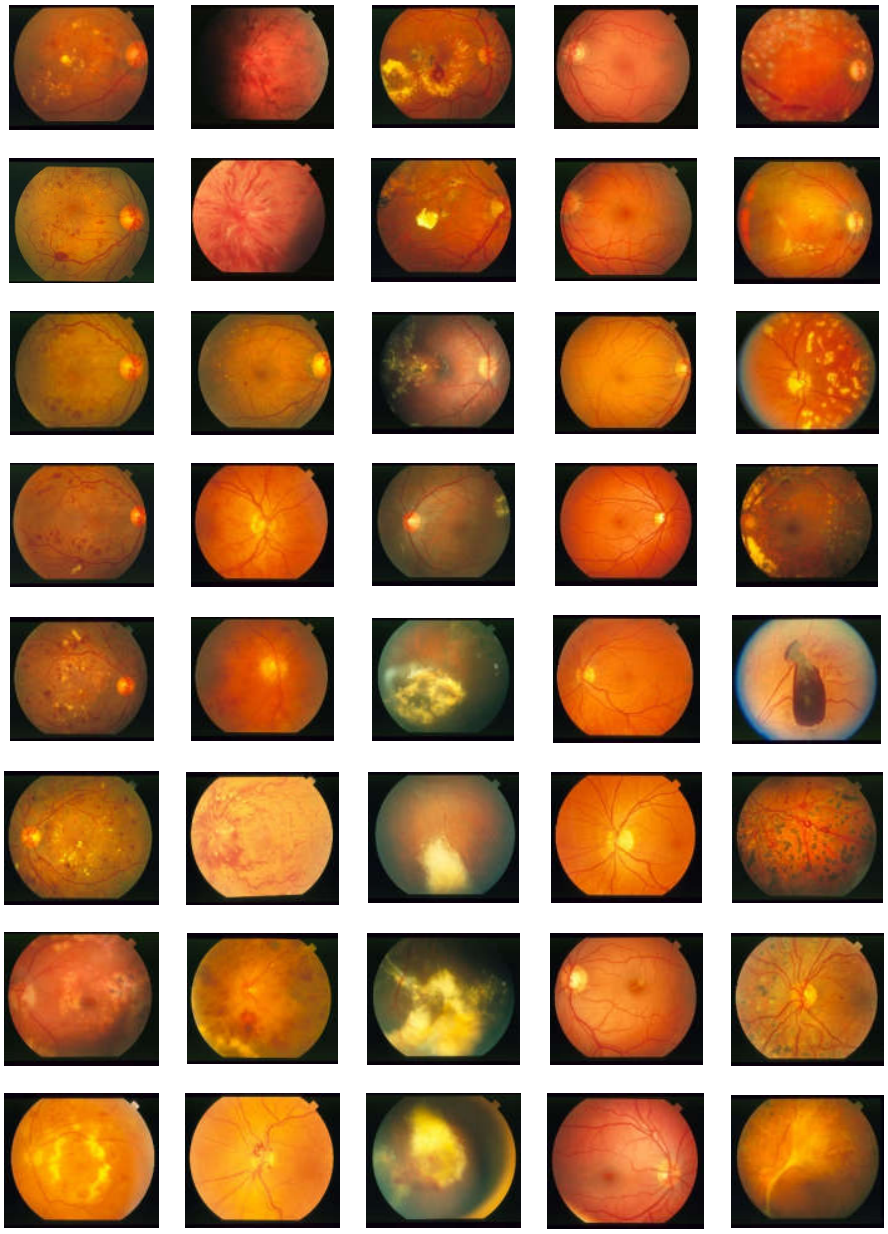
Bab ini dimulai dengan membahas dataset yang digunakan, kemudian menyajikan lingkungan yang digunakan dalam melakukan penelitian, dan diakhiri dengan hasil dan pembahasan. Penyajian hasil dibahas dengan mengikuti langkah-langkah berurutan dimulai dari proses segmentasi, enhancement, ekstraksi fitur dan klasifikasi.

4.2. Dataset

STARE (STstrured Analisis of the Retina) adalah database yang tersedia untuk umum. Database berisi dua set segmentasi manual yang dibuat oleh dua pengamat yang berbeda. Kinerja dihitung dengan segmentasi pengamat pertama sebagai ground truth.

Ketersediaan data terdiri dari 400 citra asli, 40 citra ground truth oleh 2(dua) ahli untuk segmentasi pembuluh darah manual dan hasil segmentasi sistem, hasil pelabelan pembuluh arteri dan vena oleh 2(dua) ahli dan 80 citra hasil deteksi optik disk dan ground truth manual.

Pada penelitian ini, dataset yang digunakan adalah citra retina sebanyak 107 data latih dan 25 data uji yang dibagi menjadi 5 kelas. Adapun jenis penyakit yang digunakan yaitu penyakit Background Diabetic Retinopathy (BDR), Central Retinal Vein Occlusion (CRVO), Coats, Proliferative Diabetic Retinopathy (PDR) dan kelas Normal. Gambar 4.1 merupakan contoh citra berdasarkan kelas penyakit dan kelas Normal yang digunakan pada penelitian ini. BDR mempunyai ciri adanya bintik kecil merah, CRVO mempunyai ciri adanya pembengkakan di pembuluh darah, Coats mempunyai ciri awal yang terdiri dari area kekuningan besar di dalam dan di bawah retina, Normal mempunyai ciri Sel-sel saraf fotoreseptor hadir di makula dan terkonsentrasi di fovea, dan PDR mempuyai ciri pendarahan ke dalam substansi seperti jelly di tengah mata (humor vitreous)



(a) (b) (c) (d) (e)

Gambar 4. 1. Contoh Citra untuk Penyakit dan Normal a. Background Diabetic Retinopathy (BDR), b. Central Retinal Vein Occlusion (CRVO), c. Coats, d. Normal dan e. Proliferative Diabetic Retinopathy (PDR)

4.3. Lingkungan Penelitian

Penelitian dilakukan untuk menguji kinerja metode yang diusulkan pada bab 3. Untuk mencapai tujuan tersebut, perangkat lunak dibangun dengan menggunakan Bahasa pemograman JAVA dan Lingkungan Pengembangan Terintegrasi IDE(*Integrated Development Environment*) NetBeans 8.2 dengan spesifikasi sebagai berikut:

Product Version: NetBeans IDE 8.2 (Build 201609300101)
Updates: Updates available to version NetBeans 8.2 Patch 2
Java: 1.8.0_131; Java HotSpot(TM) 64-Bit Server VM 25.131-b11
Runtime: Java(TM) SE Runtime Environment 1.8.0_131-b11
System: Windows 10 version 10.0 running on amd64; Cp1252; en_US (nb)
User directory: C:\Users\erwin\AppData\Roaming\NetBeans\8.2
Cache directory: C:\Users\erwin\AppData\Local\NetBeans\Cache\8.2

Pembangunan perangkat menggunakan metode Rational Unified Process (RUP). Rational Unified Process (RUP) merupakan suatu metode rekayasa perangkat lunak yang dikembangkan dengan mengumpulkan berbagai best practises yang terdapat dalam industri pengembangan perangkat lunak. Ciri utama metode ini adalah menggunakan use-case driven dan pendekatan iteratif untuk siklus pengembangan perankat lunak. RUP menggunakan konsep object oriented, dengan aktifitas yang berfokus pada pengembangan model dengan menggunakan Unified Model Language (UML).

Fase RUP terdiri dari:

1. Inception/insepsi
2. Elaboration/elaborasi
3. Construction/konstruksi
4. Transition/transisi

Perangkat lunak dijalankan dengan prosesor Intel Core 2 Duo, CPU T8100@2.10 GHz, dengan memori 2 GB. Berikut ini langkah-langkah dalam membangun aplikasi:

1. Membangun program utama, prosedur, fungsi dan tampilan antar muka untuk modul-modul preprosesing, enhancement, segmentasi, ekstraksi fitur dan deteksi.

2. Membuat dataset dengan field yang terdiri atas nama direktori, nama file, image file, jenis penyakit dan Normal, dua kelas(secara manual oleh ahli dari dataset Stare dan Drive), dan kelas pengujian(berdasarkan hasil deteksi dari metode yang diusulkan).
3. Akuisisi citra retina dari dataset Stare dan Drive dengan format file (.ppm), dan di konversi ke format (.jpg). Selanjutnya dilakukan poses prapengolahan citra dengan melakukan Resize, dari citra original yang berukuran 600 x 705 pixel menjadi 255 x 255 pixel. Langkah ini digunakan untuk mempercepat proses deteksi.
4. Menganalisis kinerja setiap metode yang diusulkan dimulai dari praprosesing, enhancement, segmentasi, ekstraksi ciri dan deteksi. Tabel 4.1 menyajikan pengukuran kinerja dari metode yang diusulkan.

Tabel 4.1. Pengukuran kinerja

Metode	Pengukuran Kinerja	
	Kualitatif	Kuantitatif
Segmentasi	Manual dengan mata	PSNR
Enhancement	Manual dengan mata	Distribusi histogram
Ekstraksi fitur bentuk	Manual dengan mata	Area, Perimeter, Metric, Eccentricity
Ekstraksi fitur tektur	Manual dengan mata	Kontras (C), Korelasi (CO), Energi (E), dan Homogeneitas (H)
Ekstraksi fitur warna	Manual dengan mata	Nilai H, S dan V
Klasifikasi Citra Retina Penyakit dan Normal	Manual dengan mata	Spesifikasi, Sensitivitas, Akurasi dengan confusion matrik dan QEF dengan Connectivity, Area dan Length

4.4. Hasil Penelitian dan Pembahasan

Hasil penelitian disajikan dan dibahas secara berurutan berdasarkan tahapan proses dimulai dari proses segmentasi, enhancement, ekstraksi fitur dan klasifikasi. Kinerja hasil klasifikasi diperoleh dengan menerapkan hasil yang diperoleh dari

ekstraksi fitur. Begitu juga, hasil ekstraksi fitur diperoleh dari hasil enhancement dan segmentasi.

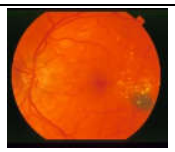
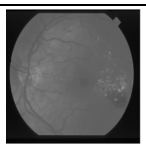
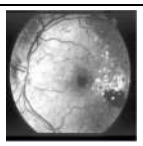
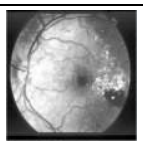
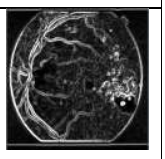
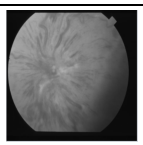
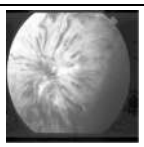
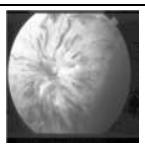
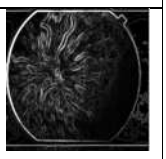
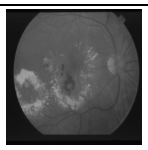
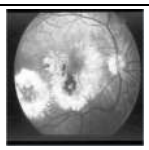
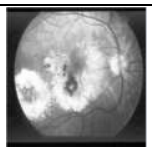
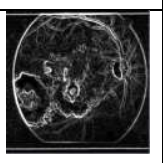
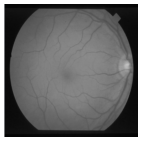
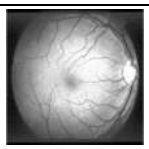
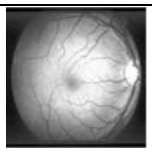
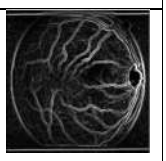
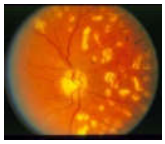
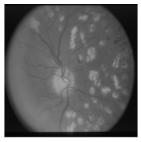
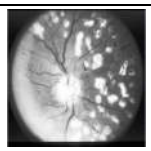
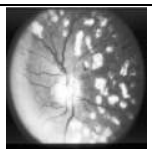
4.4.1. Performance Hasil Pra Pengolahan dan Enhancement

Proses preprocessing dilakukan dengan menggunakan teknik grayscale untuk mengubah kanal menjadi 1 nilai saja. Output dari grayscale memiliki warna keabuan. Selanjutnya, teknik enhancement menggunakan Histogram Equalization (HE) pada foreground yaitu peningkatan warna kontras agar urat dari retina mata semakin terlihat, dilanjutkan dengan gaussian filter untuk mengeblurkan citra retina. Lalu, citra retina dilakukan teknik deteksi sobel untuk memisahkan antara garis tepi yaitu berupa urat pada retina mata. Ukuran pixel asli dari citra retina adalah 600x705. Dengan mempertimbangkan kecepatan untuk mendapatkan informasi pada saat melakukan deteksi dilakukan resize yaitu kompresi terhadap pixel retina menjadi 256x256. Selajutnya, proses grayscale yang dilanjutkan dengan pencerahan kontras dengan menggunakan HE. Kontras yang telah dipertajam diambil cirinya dengan menggunakan deteksi tepi sobel.

a. Tombol grayscale : untuk mengubah citra retina menjadi skala abu-abu agar citra retina memiliki 1 kanal saja. Proses ini berpengaruh pada deteksi sobel.
b. Histogram Ekualisasi : untuk meningkatkan kontras pada skasla abu-abu pada foreground pembuluh, untuk menonjolkan urat yang terdapat pada proses grayscale sebelumnya digunakan persamaan 2.8.
c. Gaussian Filter : untuk memberikan efek blur pada gambar citra dengan menggunakan persamaan 2.12.
d. Deteksi Sobel : untuk menampilkan garis dan ciri dari retina yang telah terdeteksi dengan menggunakan persamaan 2.18.

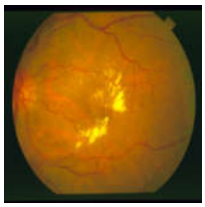
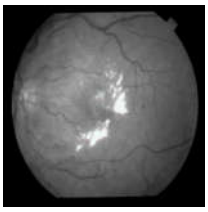
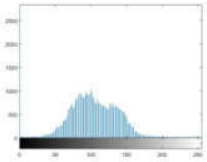
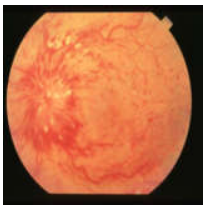
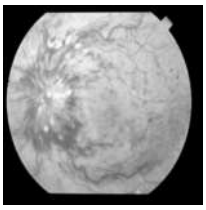
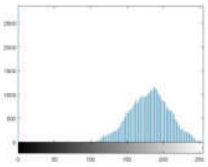
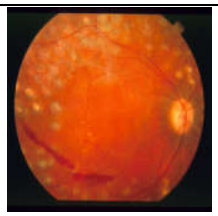
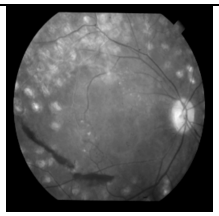
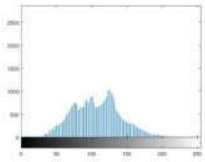
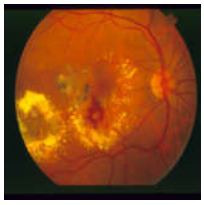
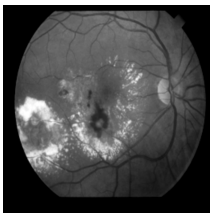
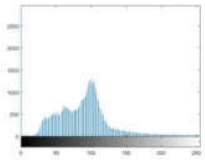
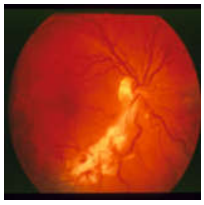
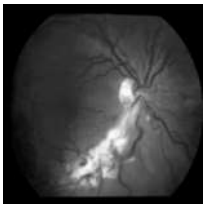
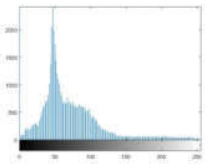
Tabel 4.2 merupakan hasil preprocessing untuk setiap kelas BDR, CRVO, Coats, Normal dan PDR. Sehingga hasil enhancement dengan menggunakan preprocessing dapat dilihat pada Tabel 4.3 :

Tabel 4.2. Hasil Praprocessing kelas BDR, CRVO, Coats, Normal dan PDR

Kelas	Retina	Grayscale	HE	Gaussian Filter	Deteksi Sobel
BDR					
CRVO					
Coats					
Normal					
PDR					

Tabel 4.3. Hasil dari proses enhancement citra retina

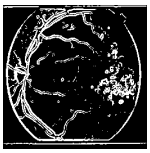
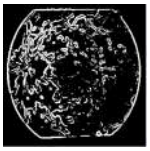
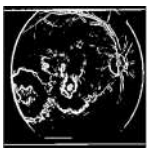
Nama Penyakit Retina	Citra RGB	Hasil Prapengolahan	
		Enhancement	Histogram

Background Diabetic Retinopathy			
Central Retinal Vein Occlusion			
Proliverative Diabetic Retinopathy			
Coats			
Retinitis			

4.4.2. Performance Hasil Segmentasi

Hasil yang telah didapatkan dari deteksi tepi sobel dipisahkan lagi untuk menentukan background dan foregroundnya dengan menggunakan Multilevel Thresholding. Tabel 4.4 adalah tabel hasil pengujian data training dengan menggunakan multilevel thresholding :

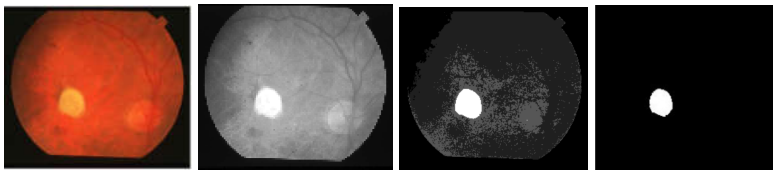
Tabel 4. 4. Hasil Pengujian data training dengan Multilevel Thresholding.

No	Kelas	Retina	Th Level	PSNR
1	BDR		2	20.05
			3	19.93
			4	20.28
			5	20.14
2	CRVO		2	20.07
			3	19.95
			4	20.01
			5	20.12
3	Coats		2	20.05
			3	20.05
			4	19.93
			5	20.09
4	Normal		2	20.08
			3	20.18
			4	20.16
			5	20.20
5	PDR		2	20.11

			3	19.93
			4	20.15
			5	20.02

5. Pengujian Tahap Segmentasi MTIHSA

Pengujian proses segmentasi citra dilakukan untuk mengetahui apakah citra RGB yang ditangkap kamera telah memenuhi kebutuhan untuk melakukan pengekstraksian fitur. Didalam proses segmentasi citra, citra RGB diubah menjadi citra *greyscale* lalu dari citra *greyscale* diubah menjadi citra *binner*. Tahap segmentasi yang menggunakan metode otsu tresholding, karena teknik ini mampu memilih nilai ambang secara otomatis, sehingga pemisahan foreground sempurna seperti yang disajikan pada Gambar 4.2.



Gambar 4.2. Proses Segmentasi Citra MTIHSA

Algoritma *harmony search* menghasilkan nilai ambang batas probabilitas distribusi kelas pada algoritma otsu. Setelah ambang batas dimasukan, algoritma otsu menjalankan proses sampai mencari varian otsu. Hasil varian otsu merupakan harmoni memori pada *harmony memory size*. Banyaknya harmoni memori bergantung dengan *harmony memory size*. Kemudian harmony *memory considering rate* memilih harmoni memori yang harus diganti dengan perbandingan nilai *random*. Nilai kemudian diperbaharui mengikuti petunjuk parameter PAR dan BW. Nilai PAR mengakibatkan seberapa persen nilai berubah sedangkan nilai BW mengakibatkan seberapa jauh nilai intensitas bertambah (nilai intensitas membersar) atau berkurang (nilai intensitas mengecil). Perubahan intensitas bertambah jika nilai BW positif, sebaliknya perubahan intensitas berkurang jika nilai BW negatif. Setelah nilai diperbaharui, nilai tersebut disimpan di harmoni memori sebanyak perulangan yang ditetapkan. Nilai yang terbaik pada harmoni

memori diambil sebagai nilai ambang batas. Nilai ambang batas tersebut mempengaruhi intensitas piksel yang dimasukan ke dalam beberapa kelas menurut teknik multithresholding. Tabel 4.5 merupakan contoh hasil perhitungan MTIHSA untuk data training dan Tabel 4.6. menyajikan contoh hasil perhitungan MTIHSA untuk data uji. Hasil perhitungan untuk seluruh citra training dan citra uji disajikan pada lampiran 1.

Tabel 4.5. Hasil Multithresholding Improved Harmony Search pada Data Training

No	Image	k	Thresholds	STD	PSNR	MEAN
1	1	2	54	0,36	17,08	785.82
		3	11; 57	0,66	21,53	1.033, 90
		4	12; 34; 98	5,37	21,45	875,35
		5	12; 28; 58;73	0,63	29,24	914,46
2	2	2	49	0,60	19,92	662,10
		3	51; 174	2.0581e-11	21,16	758,02
		4	14; 35; 124	5,37	23,31	755,04
		5	8; 27; 50; 175	3,02	24,05	861,14
3	3	2	58	5,9129e-11	13,37	2,4645e+03
		3	42; 125	1,86	16,79	2,6074e+03
		4	53; 101; 147	0,29	20,26	2,6530e+03
		5	50; 101; 147; 197	0,99	19,87	2,6617e+03

Tabel 4. 6. Hasil Multithresholding Improved Harmony Search pada Data Uji

No	Image	k	Thresholds	STD	PSNR	MEAN
1	1	2	208	2.5471e-11	19,29	785,82
		3	12; 87	0,18	21,53	1.0339e+03
		4	12; 34; 98	5,37	21,453	875,35

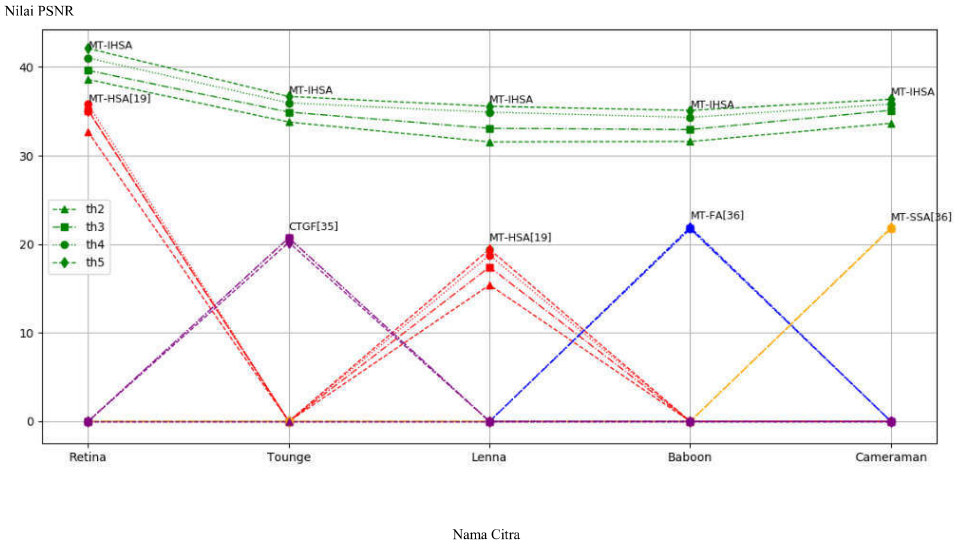
		5	12; 28; 58; 73	0,63	29,24	914,46
2	2	2	65	3,172e-11	22,82	662,10
		3	12; 49	2,058e-11	21,16	758,02
		4	12; 75; 102	1,14	29,12	-
		5	8; 27; 50; 175	3,02	24,05	861,14
3	3	2	58	5,913e-11	13,37	2,465e+03
		3	47; 141	7,960e-11	17,95	2,607e+03
		4	53; 111; 147	0,29	21,26	2,653e+03
		5	51; 101; 146; 197	0,99	19,97	2,662e+03

Analisis Hasil Segmentasi dengan MTIHSA

Pengukuran keberhasilan metode yang diusulkan menggunakan nilai PSNR. Hasil pengujian menggunakan dataset retina STARE diperoleh nilai PSNR lebih tinggi dibandingkan citra lidah, lena, babon dan kameraman. Rata-rata nilai PSNR dari citra retina Stare adalah 40,34 dB yang berarti proses segmentasi berhasil dengan baik sehingga penggunaan metode yang diusulkan ini meningkatkan akurasi pada proses klasifikasi. Hasil percobaan dari metode yang diusulkan ini dibandingkan dengan metode-metode lainnya yaitu (B. Zhang, Kumar, & Zhang, 2013) yang menggunakan segmentasi warna, tekstur dan fiur geometri(CTGF), (Oliva et al., 2013) menggunakan Multilevel Thresholding harmony search algorithm (MT-HSA), dan (R. Singh, Agarwal, Kashyap, & Bhattacharya, 2016) yang menggunakan Multilevel Thresholding Firefly Algorithm(MT-FA) dan Multilevel Thresholding Social Spider Algorithm (MT-SSA) disajikan pada Tabel 4.7 dan Gambar 4.3 di bawah ini:

Tabel 4.7. Perbandingan nilai PSNR antara metode MT-IHSA dengan metode lainnya

Citra	MT-HSA				MT-IHSA				MT-FA				MT-SSA				CTGF			
	Thersholding				Thersholding				Thersholding				Thersholding				Thersholding			
	2	3	4	5	2	3	4	5	2	3	4	5	2	3	4	5	2	3	4	5
Retina	32.69	35.121	35.826	35.121	38.602	39.628	41.032	42.107	0	0	0	0	0	0	0	0	0	0	0	0
Tounge	0	0	0	0	33.796	34.923	35.957	36.683	0	0	0	0	0	0	0	0	0	20.721	20.724	20.207
Lenna	15.401	17.427	18.763	19.443	31.559	33.079	34.909	35.578	0	0	0	0	0	0	0	0	0	0	0	0
Baboon	0	0	0	0	31.599	32.946	34.329	35.121	0	21.744	21.759	21.924	0	0	0	0	0	0	0	0
Cameraman	0	0	0	0	33.659	35.121	35.826	36.37	0	0	0	0	0	21.783	21.791	21.869	0	0	0	0



Gambar 4.3. Grafik Perbandingan Nilai PSNR

4. 4.3. Performance Proses Ekstraksi Fitur

4.4.3.1. Performance Ekstraksi Fitur berdasarkan Tekstur

Nilai ekstraksi fitur tekstur diperoleh dari metode Gray Level Co-occurrence Matric (GLCM). Tahap pertama dalam metode ini yakni nilai pixel pada sebuah citra retina yang sudah di perbaiki kualitasnya, terlebih dahulu di quantization untuk diproses co–occurrence matrix. Pada sebuah citra terdiri dari 256 variasi warna yang di bagi dalam 8 variasi.

Tahap kedua yakni menghitung dari keempat matriks GLCM untuk 4 orientasi berbeda masing-masing untuk $\theta = 0°$, $\theta = 45°$, $\theta = 90°$, dan $\theta = 135°$, dengan radius $\delta = 1$ (d adalah jarak antara dua piksel). Dengan demikian, dapat diperoleh $GLCM_0$, $GLCM_{45}$, $GLCM_{90}$, $GLCM_{135}$.

Dari setiap GLCM matriks yang dikonstruksi pada tahap kedua, kemudian dilakukan ekstraksi ciri-ciri seperti kontras (C), korelasi (CO), energi (E), dan homogeneitas (H) dengan menggunakan persamaan (2.23) hingga (2.27). Tahap selanjutnya membuat fitur vektor dari properti tersebut dengan mengambil rata-rata setiap ciri dengan keempat orientasi θ yang ditunjukkan pada Tabel 4.8 berikut ini.

Tabel 4.8 Hasil perhitungan rata-rata setiap fitur

Ciri-ciri	Orientasi θ				Rata-rata
	0°	45°	90°	135°	
Contrast	0,42	0,32	0,08	0,42	0,31
Correlation	1,89	2,63	2,72	2,82	2,52
Energy	0,35	0,49	0,50	0,36	0,43
Homogeneity	0,78	0,82	0,95	0,76	0,83

Untuk hasil proses ekstraksi fitur tekstur citra retina dengan GLCM secara keseluruhan ditunjukkan pada Tabel 4.9 dan seluruh proses perhitungan ekstraksi fitur ini disajikan dalam lampiran 2 dan listing program disajikan pada lampiran 3

Tabel 4.9. Hasil proses ekstraksi fitur tekstur citra retina

No.	Nama Penyakit Retina	Ciri GLCM	Orientasi θ				
			0°	45°	90°	135°	Rata-rata
1	*Background Diabetic Retinopathy*	Contrast	0,10	0,15	0,10	0,15	0,13
		Correlation	0,98	0,98	0,99	0,98	0,98
		Energy	0,17	0,16	0,17	0,16	0,16
		Homogeneity	0,96	0,94	0,95	0,94	0,95
2	*Central Retinal Vein Occlusion*	Contrast	0,26	0,38	0,22	0,39	0,31
		Correlation	0,98	0,97	0,98	0,97	0.98
		Energy	0,20	0,19	0,21	0,19	0.19
		Homogeneity	0,94	0,92	0,92	0,92	0.93
3	*Proliverative Diabetic Retinopathy*	Contrast	0,18	0,25	0,16	0,26	0.21
		Correlation	0,98	0,98	0,99	0,98	0.98
		Energy	0,15	0,14	0,15	0,14	0.15
		Homogeneity	0,94	0,92	0,94	0,92	0.93
4	*Coats*	Contrast	0,16	0,24	0,16	0,23	0.20

		Correlation	0,98	0,96	0,98	0,97	0.97
		Energy	0,17	0,16	0,17	0,16	0.16
		Homogeneity	0,93	0,91	0,93	0,91	0.92
		Contrast	0,09	0,13	0,10	0,14	0.12
		Correlation	0,98	0,97	0,98	0,97	0.98
5	*Retinitis*	Energy	0,17	0,16	0,17	0,16	0.17
		Homogeneity	0,95	0,94	0,95	0,94	0.95

Jika dianalisis dari Tabel 4.9, hasil dari proses ekstraksi fitur menunjukkan bahwa nilai kontras korelasi, energi dan homogenitas secara independen memainkan peran penting dalam penentuan directionality citra retina dalam proses ekstraksi fitur tekstur. Nilai Kontras minimum sedangkan korelasi, energi dan homogenitas maksimum pada sudut 0°. Untuk tiga sudut lainnya, keempat fitur yang dihitung hampir sama.

Sementara pasda sudut 45° nilai Kontras dan Energi minimum sedangkan nilai Korelasi dan Homogenitas maksimum, hal ini karena nilai abu-abu konstan pada sudut 45° itulah sebabnya Kontras dan Energi minimum sepanjang sudut itu dan Korelasi serta Homogenitas masing-masing maksimum.

Nilai-nilai Kontras lagi minimum sementara Korelasi, Energi, dan Homogenitas maksimum untuk sudut 90°. Sudut ini juga dapat dianggap sebagai sudut arah utama. Hal ini karena nilai abu-abu tidak konstan pada sudut 90°, itulah sebabnya Kontras keluar menjadi minimum sementara Korelasi, Energi, dan Homogenitas masing-masing keluar menjadi maksimal.

Pada sudut 135° juga sama seperti sudut 45° nilai Kontras dan Energi minimum sementara Korelasi, dan Homogenitas adalah maksimum. Jadi ketika mengenali fitur seperti itu, sudut-sudut ini dapat secara khusus memainkan peran penting dalam memberi hasil yang lebih akurat.

4.4.3.2. Performance Ekstraksi Fitur berdasarkan Bentuk

Pada ekstraksi fitur berdasarkan bentuk ini menggunakan 45 citra retina. Nilai ekstraksi fitur dari masing-masing jenis retina yaitu, Matrik, Eccentricity, Perimeter dan Area. Sedangkan target yang ingin dicapai berjumlah 3 kelas disajikan pada lampiran 4.

Hasil dari ekstraksi fitur bentuk inilah yang menjadi nilai input untuk pelatihan MLP disajikan pada Tabel 4.10.

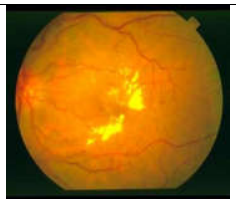
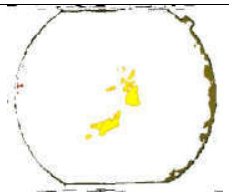
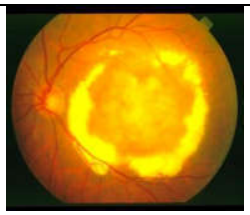
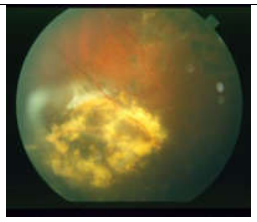
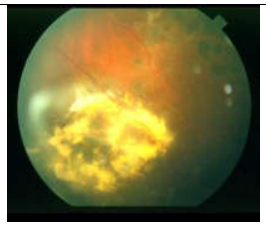
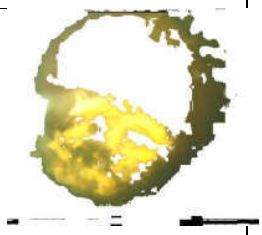
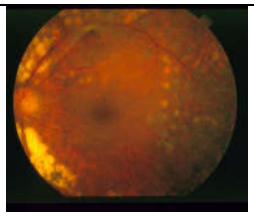
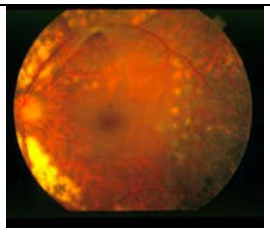
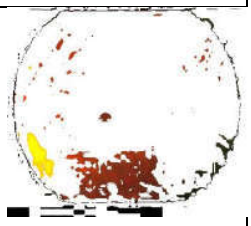
Tabel 4.10. Hasil Ekstraksi Bentuk

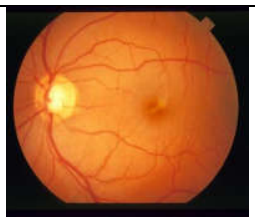
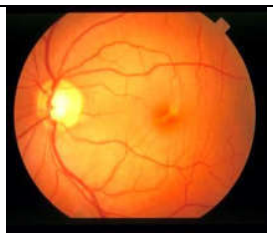
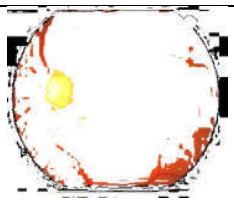
No	Metric	Eccentricity	Area	Perimeter
Choroidal Neovasscularization				
1	0,97	0,47	14,97	441,38
2	0,98	0,49	8,65	333,59
3	0,98	0,49	14,75	435,96
Branch Retinal Vein Occlussion				
4	6,54	0,87	2	1,96
5	0,65	0,87	2,49	219,99
Histoplasmosis				
6	2,17	0,35	7	6,37
7	2,17	0,35	7	6,37

4.4.3.3. Performance Ekstraksi Fitur Berdasarkan Warna

Setelah melakukan segmentasi warna yaitu melakukan pembagian warna citra berdasarkan nilai hue yang dikelompokan menjadi Sembilan bagian warna yaitu merah, jingga, kuning, hijau, cyan, biru, ungu, magenta, dan merah muda, maka pada tahap ekstraksi fitur yaitu melakukan pengambilan warna yang merupakan warna dari ciri penyakit retina seperti pada pada penelitian (Cui et al., 2018) (Ghorbanian, Jaulim, & Chatziralli, 2012) (Abdelraouf, Sonbol, & Mohamed, 2018) (Wykoff et al., 2018) yang menjelaskan bahwa pembuluh darah yang bocor dicirikan dengan bercak merah dan penumpukan protein, lalu terdiri dari area kekuningan besar di dalam dan di bawah retina sehingga dilakukan ekstraksi fitur warna merah dan kuning pada citra retina. Hasil ekstraksi berdasarkan warna disajikan pada Tabel 4.11 di bawah ini:

Tabel 4.11. Hasil Ekstraksi Fitur Warna

No	Nama Penyakit Retina dan Normal	Citra Normal	Citra Hasil Perbaikan Citra	Citra Hasil Segmentasi Warna HSV
1	BDR			
2	CNV			
3	Coats			
4	PDR			

5	Normal			

Analisis Hasil Ekstraksi Fitur

Perbaikan akurasi untuk klasifikasi penyakit kelainan retina dilakukan pada proses ekstraksi fitur. Penelitian ini mengusulkan metode ekstraksi berdasarkan tektur, bentuk dan warna sesuai dengan ciri-ciri setiap penyakit kelainan retina. Penyakit kelainan retina ada yang memiliki ciri bentuk seperti perubahan bentuk seperti mokula, pembuluh darah dan fovea, untuk ciri tektur seperti tektur pembuluh darah yang lebar dan sempit sebagai suatu kelainan, sedangkan ciri warna seperti ada bintik dan pembuluh darah yang pecah yang menimbulkan perubahan warna. Umumnya pengukuran keberhasihan metode menggunakan nilai sensitivity, specificity dan akuran, namun untuk pengujian hasil ekstraksi ini mengunakan fungsi yang baru dari (Gegúndez-Arias, Aquino, Bravo, & Marín, 2012) yang dinamakan Quality Evaluation Function (QEF). Fungsi ini menghitung nilai connectivity, area, dan length dari hasil ekstraksi fitur pembuluh darah dengan menggunakan persamaan (4.1).

$$f(C, A, L) = C * A * L \tag{4.1}$$

Pengukuran keberhasilan diperoleh dengan membandingkan nilai piksel antara ekstraksi fitur secara otomatis dengan manual(ground-truth). Hasil perhitungan QEF untuk ekstraksi pembuluh darah menggunakan dataset STARE dan dibandingkan dengan hasil yang diperoleh (B. Zhang, Zhang, Zhang, & Karray, 2010), (Nguyen, Bhuiyan, Park, & Ramamohanarao, 2013) dan (Yin et al., 2015) disajikan pada Tabel 4.12. menunjukkan hasil bahwa metode yang diusulkan mendapatkan nilai QEF yang paling tinggi dibandingkan metode lainnya.

Tabel 4.12. Perbandingan Kinerja Ekstraksi Pembuluh Darah

Metode	Connectivity(C)	Area(A)	Length(L)	C*A*L
manual	0.99	0.85	0.86	0.73
(B. Zhang et al., 2010)	0.99	0.78	0.80	0.63
(Nguyen et al., 2013)	0.99	0.79	0.81	0.63
(Yin et al., 2015)	0.99	0.82	0.84	0.69
Metode yang diusulkan	0.96	0.83	0.81	0.71

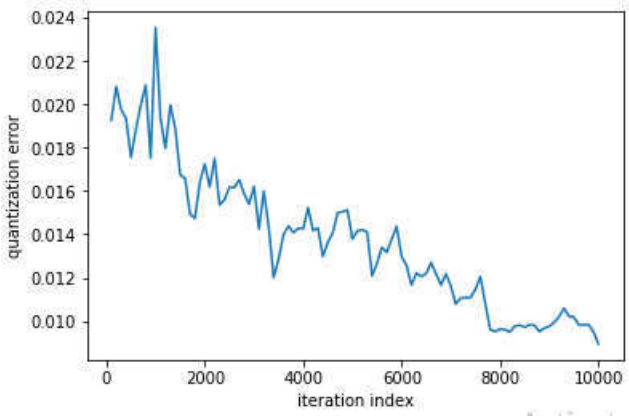
Tabel 4.12 di atas menunjukkan bahwa hasil kinerja dari metode yang diusulkan dengan pengujian menggunakan dataset STARE. Perhitungan dengan fungsi CAL memperoleh nilai 71,32 % lebih tinggi dibandingkan dengan metode lainnya dan mendekati nilai ground truth.

4.4.4. Performance Hasil Klasifikasi Penyakit Kelainan Retina

4.4.4.1. Performance Hasil Klasifikasi Penyakit dengan SOM

SOM mengambil data input dari GLCM berupa fitur vektor dan kemudian SOM mencari prototipe yang mirip dengan permintaan citra input dari retina. Jumlah sampel tekstur yang paling mirip dengan citra retina diperoleh sebagai output. Data training dan data testing yang dijadikan nilai input terdiri dari 149 citra dengan jumlah citra tiap kelas yang tidak sama, hal ini berdasarkan jumlah citra retina terbanyak dibandingkan dengan kelas penyakit yang lainnya karena hasil output dari ekstraksi fitur GLCM bukan berupa citra melainkan nilai vektor. Data input berupa file CSV (Comma Separated Value) yang berisi nilai vektor dari ekstraksi fitur tekstur meliputi empat properti yaitu kontras, korelasi, energi dan homogenitas. Selama proses berlangsung sistem menampilkan *quantization error* untuk setiap iterasinya, dan berhenti ketika telah mencapai *quantization error* terkecil. Nilai akhir bobot neuron output dijadikan acuan dalam proses pemetaan data input ke dalam *cluster*.

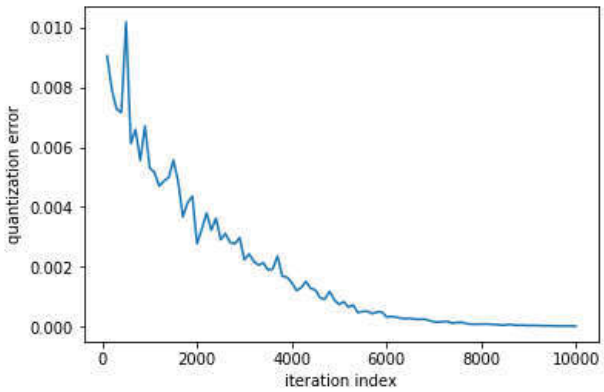
Proses perhitungan klasifikasi dengan SOM disajikan dalam lampiran 5, Gambar 4.4 merupakan perhitungan nilai error untuk setiap iterasi yang dilakukan dengan mengunakan data training.



Gambar 4.4 Grafik hasil quantization error nilai iterasi dari data training

Jika dianalisis dari hasil uji coba data training dan data testing diatas, pada Gambar 4.4 dari uji coba data training tersebut dapat dilihat yang menjadi neuron pemenang pada peta lebih dominan ke penyakit *Background Diabetic Retinopathy*, hal ini karena jumlah citranya lebih dominan terdiri dari 66 citra input dibandingkan dengan keempat kelas penyakit lainnya. Dapat dilihat juga pada Gambar 4.4, hasil dari data training dengan *quantization error* terendah mencapai 0.009. Pada tahap awal pembelajaran, peta cenderung mengatur dirinya sendiri pada vektor input. Oleh karena itu, *quantization error* rata-rata berkurang dengan cepat. Tahap ini berlangsung hingga iterasi 1000. Pada akhir fase fine-tunning (disini dari iterasi 1000 hingga 10000) *quantization error* rata-rata memiliki tren yang stabil dan semua unit peta dipesan sebaik mungkin melaui vektor input. Pada tahap ini, jaringan harus tertata dengan baik dan unit peta tersebar di seluruh vektor input.

Data testing yang dijadikan nilai input terdiri 2 citra dari tiap kelas. Hasil penurunan *quantization error* dengan meningkatkan jumlah iterasi ditunjukkan pada Gambar 4.5.



Gambar 4.5. Grafik quantization error nilai iterasi dari data testing

Dari Gambar 4.3 uji coba data testing diatas, yang menjadi neuron pemenang pada peta mengalami konstanta terhadap lima kelas penyakit retina tersebut, hal ini karena jumlah citranya yang sama masing-masing 2 citra input. Dapat dilihat juga pada Gambar 4.5, hasil dari data testing dengan *quantization error* terendah mencapai 1,473e-05 paling cocok dengan data input citra retina.

Analisa dan Pembahasan

Berikut adalah sampel 10 data penyakit citra retina yang dilakukan proses training data berupa perhitungan manual dengan empat variabel input yaitu kontras, korelasi, energi, dan homogenitas. Untuk data sampel, diambil dari dataset testing dilampiran dan kemudian dinormalisasikan, ditunjukkan pada Tabel 4.13 berikut ini.

Tabel 4.13. Data testing setelah di normalisasi

No.	Nama Penyakit	Kontras	Korelasi	Energi	Homogenitas
X_1	BDR	0,19	0,98	0,15	0,94
X_2	BDR	0,16	0,98	0,18	0,94
X_3	Coats	0,11	0,98	0,14	0,96
X_4	Coats	0,09	0,98	0,17	0,96
X_5	CRVO	0,14	0,98	0,18	0,94
X_6	CRVO	0,19	0,98	0,22	0,96
X_7	PDR	0,19	0,98	0,17	0,94
X_8	PDR	0,14	0,98	0,16	0,94
X_9	Retinitis	0,16	0,98	0,20	0,95
X_{10}	Retinitis	0,09	0,98	0,22	0,97

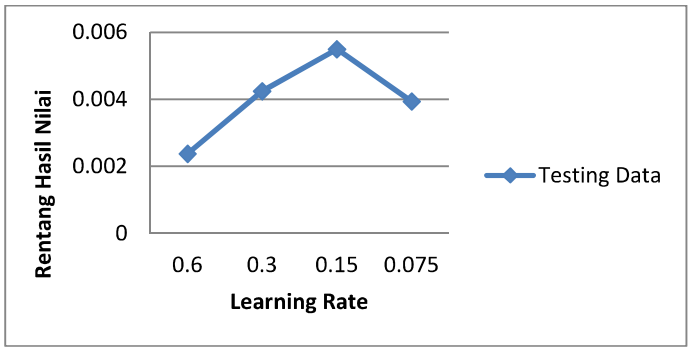
Berdasarkan input data normalisasi pada tabel diatas kemudian dilakukan training data dengan maksimum 5 iterasi. Berikut adalah hasil *clustering* 10 data dengan maksimum 5 iterasi dengan *learning rate* awal 0,6 dan bobot awal (0,1 – 1). Hasil klasifikasi dari masing-masing data input dengan maksimum 5 iterasi dinyatakan index 0 untuk *kelas* 1, index 1 untuk *kelas* 2, index 2 untuk *kelas* 3 dan index 3 untuk *kelas* 4.

Tabel 4.14. Hasil Klasifikasi 5 iterasi

No.	Variabel Input				Klasifikasi			
	Kontras	Korelasi	Energi	Homogenitas	1	2	3	4
X_1	0,19	0,98	0,15	0,94	*			
X_2	0,16	0,98	0,18	0,94	*			
X_3	0,11	0,98	0,14	0,96	*			
X_4	0,09	0,98	0,17	0,96	*			
X_5	0,14	0,98	0,18	0,94	*			
X_6	0,19	0,98	0,22	0,96	*			
X_7	0,19	0,98	0,17	0,94	*			
X_8	0,14	0,98	0,16	0,94	*			
X_9	0,16	0,98	0,20	0,95	*			
X_{10}	0,09	0,98	0,22	0,97	*			

Dari Tabel 4.14 setelah melalui 5 iterasi didapatkan hasil berupa pengelompokkan dari 10 data dengan jumlah *kelas* 0 sebanyak 10 buah. Dapat dilihat juga hasil training data dengan *learning rate* 0.6 dan *learning rate* 0,038 menghasilkan *cluster* dengan anggota semakin kecil angka indeksnya. Keakuratan hasil klasifikasi dengan semakin kecil nilai *learning rate* $\frac{10}{10} \times 100\%$ yaitu mencapai 100%.

Berikut ini pengujian data testing menggunakan pemrograman python untuk mengetahui pengaruh dari nilai *learning rate* nilai quantization error yang dapat dilihat pada Gambar 4.6 dibawah ini.



Gambar 4.6. Grafik Quantization Error nilai Learning Rate

Pada grafik Gambar 4.6 diatas merupakan hasil program data dengan pemrograman python menggunakan metode *Self-Organizing Map*. Dari grafik tersebut dapat diketahui bahwa semakin rendah nilai dari *learning rate* maka nilai keakuratan semakin besar. Dengan rentang nilai 0 sampai 0,006 dan nilai max_iterasi sebesar 100, pada garis warna biru yang merupakan nilai *quantization error* dari data testing, dapat dilihat bahwa pada *learning rate* 0.6 nilainya sangatlah rendah yaitu 0,002, *learning rate* 0,3 nilainya meningkat yaitu 0,004, *learning rate* 0,15 nilainya juga meningkat yaitu 0,006 dan *learning rate* 0,075 nilainya menurun kembali yaitu 0,004. Semakin kecil nilai *learning rate* maka semakin kecil nilai quantization error.

4.4.4.2. Performance Hasil Klasifikasi Penyakit dengan MLP

A. Analisis Hasil Pengujian Data Traning

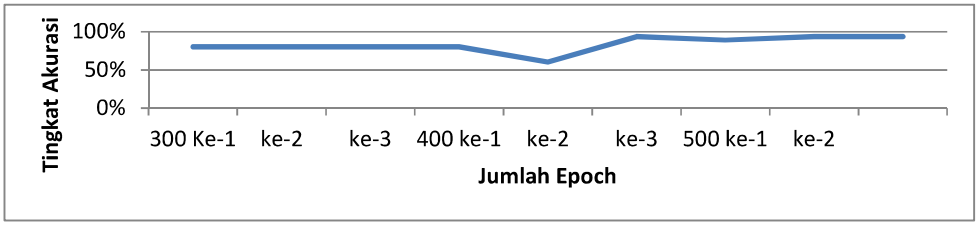
Pelatihan dilakukan dengan perbedaan jumlah epoch pada 45 data citra retina. Adapun hasil pengujian dengan data latih ini dilakukan dengan tiga skenario yaitu 300, 400 dan 500 Epoch. Proses perhitungan hasil data training dan data uji disajikan dalam lampiran 6.

Hasil pelatihan yang telah dilakukan dengan dengan perbedaan jumlah epoch dirangkum pada Tabel 4.15 di bawah ini:

Tabel 4.15. Perbandingan Hasil Pelatihan Berdasarkan Epoch

No.	Epoch	Pelatihan ke	Jumlah Target Hasil	Presentase(%)
1.	300	1	36	80%
		2	36	80%
		3	36	80%
2.	400	1	36	80%
		2	25	60%
		3	42	93,33%
3.	500	1	40	88,89%
		2	42	93,33%
		3	42	93,33%
Average				83,21%

Dari ke 45 data citra retina dapat dilihat hasil perbedaan epoch nya seperti Tabel 4.15 di atas, dari epoch terendah hingga epoch tertinggi berbeda nilai presentase akurasinya. Hasil masing-masing presentase tersebut dihitung rata-ratanya yaitu sebesar 83,210 % yang bisa diartikan bahwa hasil pelatihan algoritma tersebut sudah baik. Untuk grafik presentase tersebut bisa dilihat pada Gambar 4.7 di bawah ini.



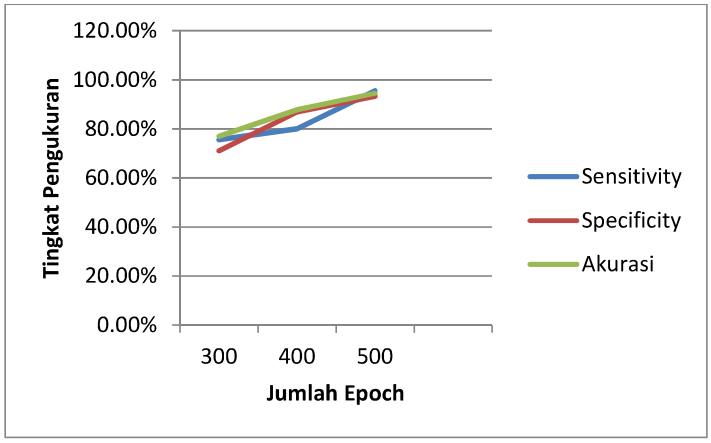
Gambar 4.7. Grafik Perbandingan Tingkat Akurasi Hasil Pelatihan

Tabel 4.16. Perbandingan Hasil Pengukuran Pelatihan Berdasarkan Epoch

No.	Epoch	Sensitivity	Specificity	Akurasi
1	300	75,56%	71,11%	76,94%
2	400	80%	86,89%	87,78%
3	500	95,56%	93,33%	94,44%

Pada Tabel 4.16. dilihat hasil pengukuran nilai *sensitivity, specificity* dan akurasinya berkaitan dengan nilai epochnya. Semakin tinggi nilai epochnya,maka nilai pengukuran *sensitivity, specificity* dan akurasinya juga semakin tinggi, karena hasil dari pengukuran tersebut diperoleh dari nilai keluaran epoch tersebut. Hasil

pengukuran pelatihan tersebut bisa dibuktikan dengan grafik pada Gambar 4.8 di bawah ini.



Gambar 4.8. Perbandingan Tingkat Pengukuran Hasil Pelatihan

B. Analisis Hasil Data Testing

Hasil pengujian yang telah dilakukan dengan dengan perbedaan jumlah epoch telah dijelaskan pada subbab sebelumnya, dan telah dirangkum pada Tabel 4.17 di bawah ini:

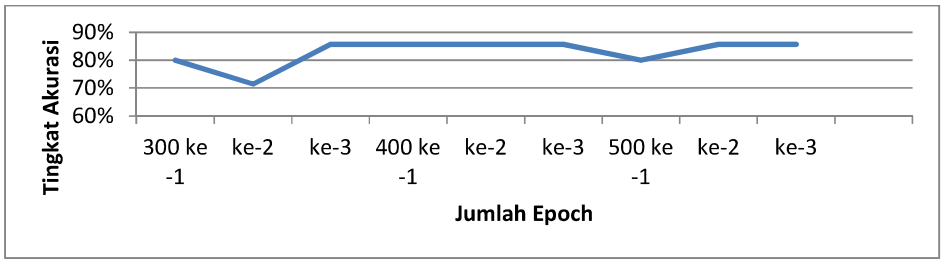
Tabel 4.17. Perbandingan Hasil Pengujian

No.	Epoch	Pengujian ke	Jumlah Target Hasil	Presentase(%)
1.	300	1	6	80%
		2	5	71,43%
		3	6	85,71%
2.	400	1	6	85,71%
		2	6	85,71%
		3	6	85,71%
3.	500	1	6	80%
		2	6	85,71%
		3	6	85,71%
Average				82,86%

Dari ke 7 data citra retina dapat dilihat hasil perbedaan epoch nya seperti Tabel 4.17 di atas, dari epoch terendah hingga epoch tertinggi berbeda nilai presentase akurasinya. Hasil masing-masing presentase tersebut dihitung rata-ratanya yaitu sebesar 82,86% yang bisa diartikan bahwa hasil pelatihan algoritma

MLP tersebut sudah baik. Dengan pengujian yang telah dilakukan tersebut dapat diimpulkan bahwa data yang digunakan pada proses pelatihan sangat menentukan keakuratan dalam dalam mengenali data yang diuji.

Adapun grafik terhadap Tabel 4.17 Perbandingan Hasil Pelatihan di atas terdapat pada Gambar 4.9. di bawah ini.

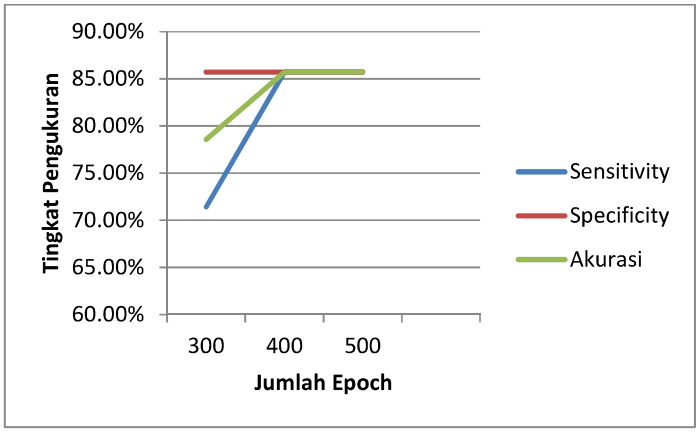


Gambar 4.9. Perbandingan Tingkat Akurasi Hasil Pengujian

Tabel 4.18. Perbandingan Hasil Pengukuran Pengujian Berdasarkan Epoch

	Epoch	Sensitivity	Specificity	Akurasi
1	300	71,43%	85,71%	78,57%
2	400	85,71%	85,71%	85,71%
3	500	85,71%	85,71%	85,71%

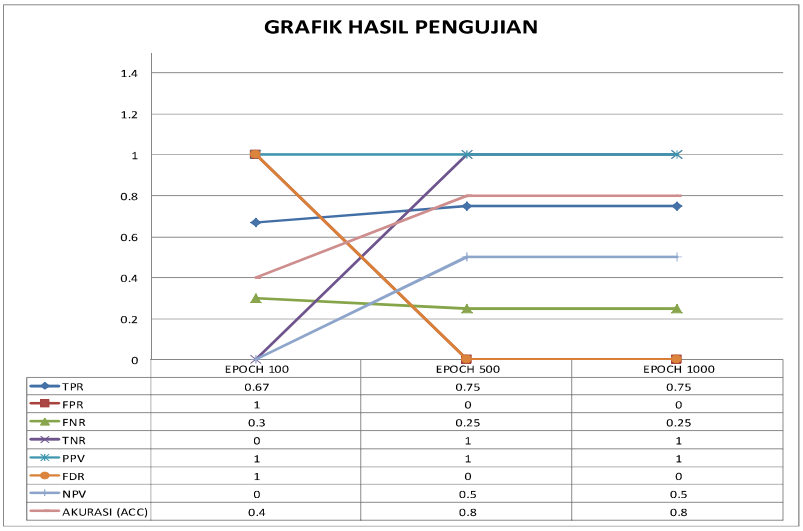
Pada Tabel 4.18. dilihat hasil pengukuran nilai *sensitivity, specificity* dan akurasinya berkaitan dengan nilai epochnya. Semakin tinggi nilai epochnya,maka nilai pengukuran *sensitivity, specificity* dan akurasinya juga semakin tinggi, karena hasil dari pengukuran tersebut diperoleh dari nilai keluaran epoch tersebut. Hasil pengukuran pengujian tersebut bisa dibuktikan dengan grafik pada Gambar 4.10 di bawah ini.



Gambar 4.10. Perbandingan Tingkat Pengukuran Hasil Pengujian

4.4.4.2. Performance Hasil Klasifikasi Penyakit dengan BP

Penelitian ini dilakukan tiga kali percobaan yaitu dengan menggunakan epoch yaitu 100, 500, dan 1000 epoch. Proses perhitungan secara lengkap disajikan pada lampiran 7. Hasil pengujian dapat dilihat pada Gambar 4.11 dibawah ini.



	EPOCH 100	EPOCH 500	EPOCH 1000
TPR	0.67	0.75	0.75
FPR	1	0	0
FNR	0.3	0.25	0.25
TNR	0	1	1
PPV	1	1	1
FDR	1	0	0
NPV	0	0.5	0.5
AKURASI (ACC)	0.4	0.8	0.8

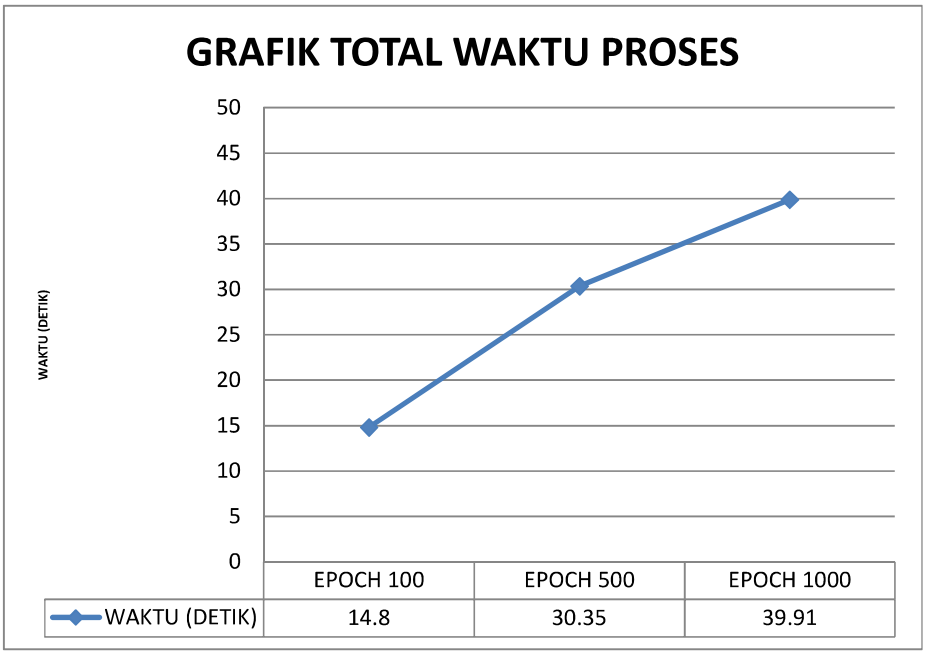
Gambar 4.11 Grafik Perbandingan nilai pengukuran kinerja

Dapat dilihat pada nilai nilai True Positive Rate (TPR) yang mengalami kenaikan dari epoch 100 ke epoch 500 yaitu dari nilai 0,67 menjadi 0,75 sedangkan pada epoch 500 ke 1000 tidak mengalami perubahan yaitu dari 0,75 menjadi 0,75. Pada False Positive Rate (FPR) terjadi penurunan dari epoch 100 hingga ke epoch 1000 yaitu dari nilai 1 menjadi 0. Pada nilai False negative rate (FNR) juga terjadi penurunan dari 0,3 menjadi 0,25. Sedangkan nilai True Negative Rate (TNR) terdapat peningkatan yaitu dari nilai 0 menjadi 1. Pada nilai Positive Predictive Value (PPV) tidak terjadi perubahan yaitu tetap pada nilai 1 pada epoch 100 hingga epoch 1000. False Discovery Rate (FDR) mengalami penurunan nilai yaitu dari 1 menjadi 0 pada epoch 100 hingga epoch 1000. Negative Predictive Value (NPV) mengalami kenaikan yaitu dari 0 menjadi 0.5 dan terakhir yaitu Accuracy (ACC) yang mengalami peningkatan akurasi pada epoch 100 ke epoch 500 yaitu dari nilai

0,4 menjadi 0,8 sedangkan pada epoch 500 ke epoch 1000 tidak terjadi perubahan yaitu tetap pada nilai 0,8.

A. Waktu Proses Klasifikasi

Penelitian pada pengujian 1 dengan penggunaan epoch 100 memiliki waktu proses 14,8 detik. Pengujian 2 dengan penggunaan epoch 500 memiliki waktu proses 30,35 detik. Pengujian 3 dengan penggunaan epoch 1000 memiliki waktu proses 39,91 detik. Pada pengujian yang telah dilakukan pada klasifikasi pengujian 1 dan pengujian 2 dengan menggunakan data yang berbeda terdapat peningkatan akurasi pada pengujian 2 hal ini dapat dilihat pada Gambar 4.12 dibawah ini.



Gambar 4.12 Grafik Akurasi dengan 100 Epoch

Pada Gambar 4.12 yaitu gambar grafik waktu proses klasifikasi yang dilakukan pada pengujian 1 yaitu dengan epoch 100 dan pengujian 2 dengan epoch 500 dan pengujian 3 dengan 1000 epoch. Dan dapat dilihat terjadi kenaikan pada waktu proses yang dilakukan yaitu pada pengujian epoch 100 waktu akhir proses klasifikasi yaitu selama 14,8 detik sedangkan pada epoch 500 mendapatkan waktu

proses selama 30,35 detik dan pada epoch 1000 meningkat menjadi 39,91 detik. Sehingga semakin banyak epoch maka proses klasifikasi menghasilkan waktu yang lebih lama.

4.4.4.4. Performance Hasil Klasifikasi Penyakit dengan CNN

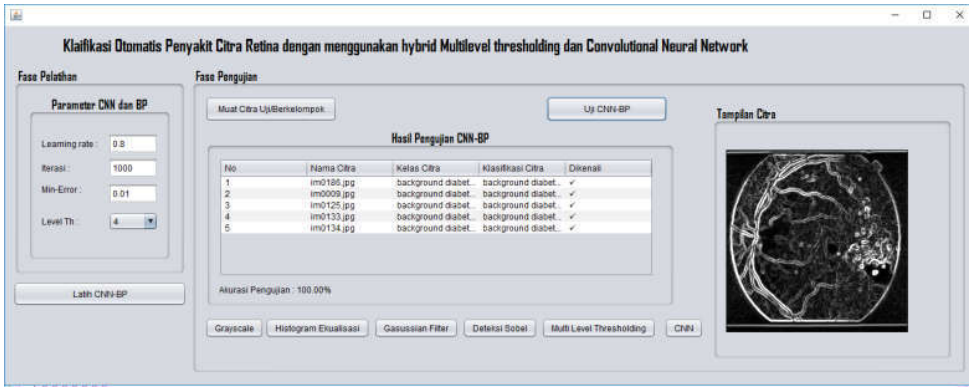
Pada hasil yang telah didapatkan pada multilevel thresholding dilakukan proses klasifikasi dengan menggunakan CNN. Pada layer inputan yang digunakan, gambar multilevel thresholding diubah kedalam bentuk matriks secara otomatis dari program yang telah di buat.

Model klasifikasi pada CNN yang telah dibuat memiliki 6 layer yaitu :

1. Layer konvolusi : pada lapisan layer konvolusi terdapat beberapa kumpulan kernel dari citra retina 256x256 pixel. Setiap filter di konvolusi terhadap gambar inputan dan fitur tersebut di ektrak dengan membentuk suatu layer baru.
2. Layer Max Pooling (subsampling) : max pooling seperti yang telah dijelaskan pada bab 3, max pooling merupakan proses diskretisasi berbasis sample. Max pooling digunakan untuk mengedown sample representasi input pada gambar-, matriks output yang tersembunyi. Ukuran kernek yang digunakan yaitu 2x2 dengan stride 2.
3. Layer aktivasi : fungsi aktivasi yang digunakan adalah fungsi aktivasi ReLu. ReLu digunakan untuk melatih jaringan syaraf secara efesien.
4. Dropout layer : layer ini digunakan untuk mencegah overtting dengan mencegah jaringan agar tidak saling bergantung ke banyaknya node pada suatu layer.
5. Fully Connected Layer (layer yang sepenuhnya terhubung): lapisan ini mengambil semua neuron dari penyatuan sebelumnya. Pada layer konvolusi terhubung sepenuhnya pada setiap neuron. Mereka tidak berlokasi secara spasial karena tidak ada lapisan konvolusional setelah lapisan tersebut terhubung secara penuh.

6. Layer Klasifikasi : pada layer klasifikasi, jaringan syaraf yang digunakan yaitu meggunakan backpropagasi untuk lapisan akhir dari pengklasifikasian untuk klasifikasi penyakit citra retina dengan mengubah matrik retina menjadi vektor. Vektor dilatih untuk mendapatkan bobot dari citra retina. Bobot tersebut digunakan untuk hasil dari klasifikasi penyakit retina.

Untuk hasil pengujian menggunakan interfasce aplikasi bisa dilihat pada Gambar 4.13, dimana hasil dari proses klasifikasi ditampilkan pada tabel yang tertera pada perangkat lunak. Akurasi dari pengujian perangkat lunak juga dapat dilihat pada hasil pengujian perangkat lunak.



Gambar 4.13. Hasil Uji CNN

A. Hasil Pengujian Perangkat Lunak

Pengujian data dilakukan dengan menggunakan data sekunder sebanyak 25 citra penyakit retina. Dimana terdapat 5 citra retina yang merupakan penyakit yang di klasifikasi dan 20 data bukan merupakan penyakit retina yang di klasifikasi. Hasil dari pengujian dengan data sekunder dilakukan dengan menghitung :

1. Melakukan pengujian dengan th 2
2. Melakukan pengujian dengan th 3
3. Melakukan pengujian dengan th 4
4. Melakukan pengujian dengan th 5

Dimana hasil dari setiap th di ukur meliputi :

a. Hasil klasfikasi diuji dengan menggunakan semua kelas. Data uji dimasukkan bukan hanya satu kelas saja, tetapi dengan memasukkan 5 kelas, jadi data yang digunakan sebanyak 25 citra retina yang menguji hasil klasifikasi tersebut. Sehingga dapat dijelaskan :
    - 5 Data uji BDR, CRVO, Coats, Normal, PDR
    - 20 Data uji yang bukan BDR, CRVO, Coats, Normal, PDR
b. Hasil klasifikasi diuji dengan mengukur kinerja dari perangkat lunak saja dengan menggunakan 25 citra penyakit retina tanpa menghitung nilai kesalahan (FP).

    Sehingga dapat dijelaskan :
    - 5 Data uji BDR, CRVO, Coats, Normal, PDR
    - 20 Data uji yang bukan BDR, CRVO, Coats, Normal, PDR

Hasil dari pengujian data penyakit retina dilakukan pengukuran akurasi secara manual dengan menggunakan confusion matrix yang telah dijelaskan pada bab 2.

B. Hasil pengujian klasifikasi dengan menghitung nilai kesalahan pada perangkat lunak

Pada sub bab ini dijelaskan hasil dari klasifikasi dengan menghitung nilai kesalahan yang dilakukan oleh perangat lunak dengan menggunakna Th 2, Th 3, Th 4 dan Th 5. Pengujian dilakukan dengan menggunakan 25 data. Dimana :

- 5 Data uji BDR, CRVO, Coats, Normal, PDR
- 20 Data uji yang bukan BDR, CRVO, Coats, Normal, PDR

1. Hasil pengujian dengan menggunakan th 2

Pengujian dengan data sekunder dilakukan dengan menggunakan bobot yang sudah dihasilkan dengan menggunakan th 2. Hasil dari th 2 dapat ditunjukkan dengan Tabel 4.19.

Tabel 4.19. Nilai Performa klasifikasi dengan Confusion Matriks untuk th 2

No	Retina	TP	TN	FP	FN	Spesifikasi	Sensitivitas	Akurasi
1	BDR	3	7	13	2	0,35	0,6	0,4
2	CRVO	5	5	15	0	0,25	1	0,48
3	Coats	3	10	10	2	0,5	0,6	0,52
4	Normal	2	12	8	3	0,6	0,4	0,56
5	PDR	0	20	5	0	0.8	0	0,8

Berdasarkan hasil dari performa klasifikasi pada Tabel 4.19, maka dapat dihitung kinerja dari teknik yang digunakan. Teknik kinerja yang dihitung berdasarkan dari Tabel 4.19 adalah sebagai berikut :

$$Spesifikasi\ (BDR) = \frac{7}{7+13} = \frac{7}{20} = 0{,}35$$

$$Sensitivitas\ (BDR) = \frac{3}{3+2} = \frac{3}{5} = 0{,}6$$

$$Akurasi\ (BDR) = \frac{3+7}{5+7+13+0} = \frac{10}{25} = 0{,}4$$

Hasil dari Tabel 4.19 menunjukkan bahwa melakukan pengujian terhadap 5 kelas penyakit retina dengan memasukkan 25 data uji yaitu 5 citra penyakit BDR, 5 citra penyakit CRVO, 5 citra penyakit Coats, 5 citra Normal dan 5 citra penyakit PDR. Salah satunya yaitu kelas penyakit BDR, pada kelas BDR untuk nilai TN yang berarti citra pada kelas BDR yang dipisahkan yang bukan kelas BDR pada Th 2 sebanyak 7 data citra penyakit retina saja yang menghasilkan spesifkasi sebesar 35%. Karena dari 25 data tersebut, FP yang berarti data tersebut sebenarnya bukan penyakit BDR tetapi data itu terklasifikasi sebagai BDR ada sebanyak 13 data. Pada FN yang berarti data yang diprediksi salah untuk kelas BDR pada perangkat lunak padahal ground truth menyatakan bahwa data tersbeut BDR ada sebanyak 2 citra dan TP yang artinya citra yang dipisahkan dengan benar pada kelas BDR ada 3 citra . Sehingga sensitivity menghasilkan nilai sebesar 60%. Sehingga akurasi yang didapatkan pada kelas BDR sebesar 40%

2. Hasil pengujian dengan menggunakan th 3

Pengujian dengan data sekunder dilakukan dengan menggunakan bobot yang sudah di hasilkan dengan menggunakan th 3. Hasil dari th 3 dapat ditunjukkan dengan Tabel 4.20.

Tabel 4.20 Nilai Performa klasifikasi dengan Confusion Matriks untuk th 3

No	Retina	TP	TN	FP	FN	Spesifikasi	Sensitivitas	Akurasi
1	BDR	5	4	16	0	0,2	1	0,36
2	CRVO	4	6	14	1	0,3	0,8	0,4
3	Coats	3	13	7	2	0,65	0,6	0,64
4	Normal	3	9	11	2	0,45	0,6	0,48
5	PDR	0	20	5	0	0,8	0	0,8

Berdasarkan hasil dari performa klasifikasi pada Tabel 4.20, maka dapat dihitung kinerja dari teknik yang digunakan. Teknik kinerja yang dihitung berdasarkan dari Tabel 4.20 adalah sebagai berikut :

$$Spesifikasi\ (CRVO) = \frac{6}{6+14} = \frac{6}{20} = 0{,}3$$

$$Sensitivitas\ (CRVO) = \frac{4}{4+1} = \frac{4}{5} = 0{,}8$$

$$Akurasi\ (CRVO) = \frac{4+6}{4+6+14+1} = \frac{10}{25} = 0{,}4$$

Hasil dari Tabel 4.20 menunjukkan bahwa melakukan pengujian terhadap 5 kelas penyakit retina dengan memasukkan 25 data uji yaitu 5 citra penyakit BDR, 5 citra penyakit CRVO, 5 citra penyakit Coats, 5 citra Normal dan 5 citra penyakit PDR. Salah satunya yaitu kelas penyakit CRVO, pada kelas CRVO untuk nilai TN yang berarti citra pada kelas CRVO yang dipisahkan yang dipisahkan dengan benar bukan kelas CRVO Th 3 sebanyak 6 data citra penyakit retina yang menghasilkan spesifikasi sebesar 30%. Karena dari 25 data tersebut, FP yang berarti data tersebut sebenarnya bukan penyakit CRVO tetapi data itu terklasifikasi sebagai CRVO ada sebanyak 14 data. Pada FN yang berarti data yang diprediksi salah untuk kelas CRVO pada perangkat lunak padahal ground truth menyatakan bahwa data tersbeut CRVO ada sebanyak 1 citra. Sehingga sensitivity menghasilkan nilai sebesar 80%. Ada 6 data pada TN pada kelas CRVO yang artinya data dipisahkan dengan benar oleh perangkat lunak dan ground truth. TP yaitu citra CRVO yang diprediksi benar sebagai CRVO ada 4 citra. Jadi total keseluruhan akurasi dari kelas CRVO sebesar 40%.

3. Hasil pengujian dengan menggunakan th 4

Pengujian dengan data sekunder dilakukan dengan menggunakan bobot yang sudah di hasilkan dengan menggunakan th 4. Hasil dari th 4 dapat ditunjukkan dengan Tabel 4.21.

Tabel 4.21. Nilai Performa klasifikasi dengan Confusion Matriks untuk th 4

No	Retina	TP	TN	FP	FN	Spesifikasi	Sensitivitas	Akurasi
1	BDR	5	6	14	0	0,3	1	0,36
2	CRVO	5	2	18	0	0,1	1	0,28
3	Coats	3	10	10	2	0,5	0,6	0,52
4	Normal	3	12	8	2	0,6	0,6	0,6
5	PDR	0	20	0	5	0,8	0	0,8

Berdasarkan hasil dari performa klasifikasi pada Tabel 4.21, maka dapat dihitung kinerja dari teknik yang digunakan. Teknik kinerja yang dihitung berdasarkan dari Tabel 4.21 adalah sebagai berikut :

$$Spesifikasi\ (Coats) = \frac{10}{10+10} = \frac{10}{20} = 0{,}5$$

$$Sensitivitas\ (Coats) = \frac{3}{3+2} = \frac{3}{5} = 0{,}6$$

$$Akurasi\ (Coats) = \frac{3+10}{3+10+10+2} = \frac{13}{25} = 0{,}52$$

Hasil dari Tabel 4.21 menunjukkan bahwa melakukan pengujian terhadap 5 kelas penyakit retina dengan memasukkan 25 data uji yaitu 5 citra penyakit BDR, 5 citra penyakit CRVO, 5 citra penyakit Coats, 5 citra Normal dan 5 citra penyakit PDR. Salah satunya yaitu kelas penyakit Coats, pada kelas Coats untuk nilai TN yaitu citra pada kelas Coats yang dipisahkan yang bukan kelas Coats pada Th 4 sebanyak 10 citra penyakit retina yang menghasilkan spesifikasi sebesar 50%. Karena dari 25 data tersebut, FP yang berarti data tersebut sebenarnya bukan penyakit Coats tetapi data itu terklasifikasi sebagai Coats ada sebanyak 10 citra. Pada FN yang berarti data yang diprediksi salah untuk kelas Coats pada perangkat lunak padahal ground truth menyatakan bahwa data tersbeut Coats ada sebanyak 2 citra. Sehingga Sensitivity menghasilkan nilai sebesar 60%. Ada 10 citra untuk TN pada kelas Coats yang artinya data dipisahkan dengan benar oleh perangkat lunak dan ground

truth. Selain itu TP yaitu citra Coats yang diprediksi benar sebagai Coats ada 10 citra. Jadi total keseluruhan akurasi dari kelas Coats sebesar 52%

4. Hasil pengujian dengan menggunakan th 5

Pengujian dengan data sekunder dilakukan dengan menggunakan bobot yang sudah di hasilkan dengan menggunakan th 5. Hasil dari th 5 dapat ditunjukkan dengan Tabel 4.22.

Tabel 4.22. Nilai Performa klasifikasi dengan Confusion Matriks untuk th 5

No	Retina	TP	TN	FP	FN	Spesifikasi	Sensitivitas	Akurasi
1	BDR	5	3	17	0	0,15	1	0,32
2	CRVO	5	4	16	0	0,2	1	0,36
3	Coats	3	13	7	2	0,65	0,6	0,64
4	Normal	2	9	11	3	0,45	0,4	0,44
5	PDR	0	20	5	0	0,8	0	0,8

Berdasarkan hasil dari performa klasifikasi pada Tabel 4.22, maka dapat dihitung kinerja dari teknik yang digunakan. Teknik kinerja yang dihitung berdasarkan dari Tabel 4.22 adalah sebagai berikut :

$$Spesifikasi\ (Normal) = \frac{9}{9+11} = \frac{9}{20} = 0{,}45$$

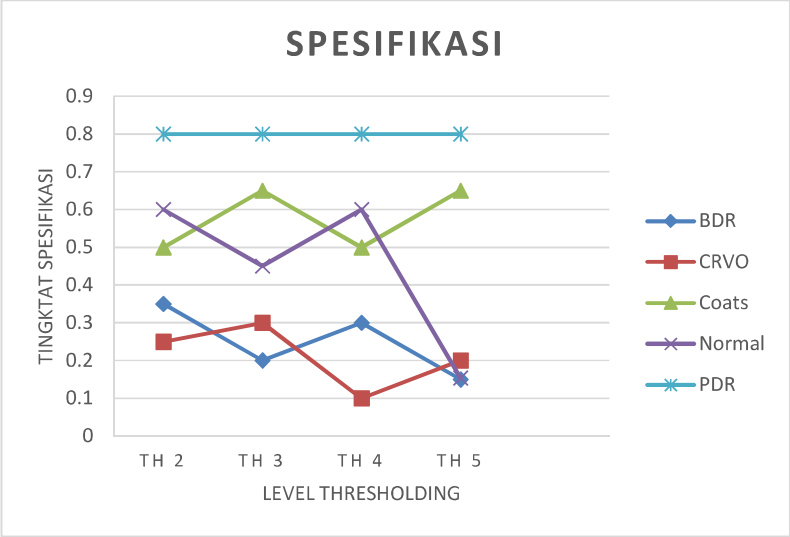
$$Sensitivitas\ (Normal) = \frac{2}{2+3} = \frac{2}{5} = 0{,}4$$

$$Akurasi\ (Normal) = \frac{2+9}{2+9+11+3} = \frac{11}{25} = 0{,}44$$

Hasil dari Tabel 4.22 menunjukkan bahwa melakukan pengujian terhadap 5 kelas penyakit retina dengan memasukkan 25 data uji yaitu 5 citra penyakit BDR, 5 citra penyakit CRVO, 5 citra penyakit Coats, 5 citra Normal dan 5 citra penyakit PDR. Salah satunya yaitu kelas Normal, pada kelas Normal untuk nilai TN yang artinya yang berarti citra pada kelas Normal yang dipisahkan yang bukan kelas Normal pada Th 5 sebanyak 9 citra penyakit retina yang menghasilkan spesifikasi sebesar 45%. Karena dari 25 data tersebut, FP yang berarti citra pada kelas Norma; yang dipisahkan yang bukan kelas Normal ada sebanyak 11 citra. Pada FN yang berarti data yang diprediksi salah untuk kelas Normal pada perangkat lunak padahal

ground truth menyatakan bahwa data tersbeut Nomal ada sebanyak 3 citra. Sehingga nilai Sensitivitas meghasilkan 40%. Ada 9 data pada TN pada kelas Normal yang artinya data dipisahkan dengan benar oleh perangkat lunak dan ground truth. TP yaitu citra Normal yang diprediksi benar sebagai Normal ada 2 citra. Jadi total keseluruhan akurasi dari kelas Normal sebesar 44%

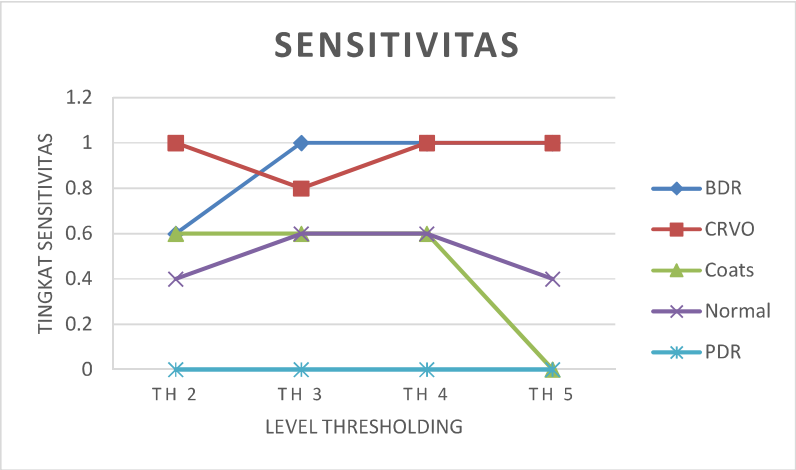
Untuk melihat perbandingan klasifikasi yang dilakukan dengan mengukur nilai Spesifikasi, Sensitivitas dan Akurasi yang menggunakan th 2, th 3, th 4, th 5, bisa dilihat pada Gambar 4.14, 4.15 dan 4.16 :



Gambar 4.14 Grafik Spesifikasi dari th 2, th 3, th 4, th 5 pada 5 kelas penyakit

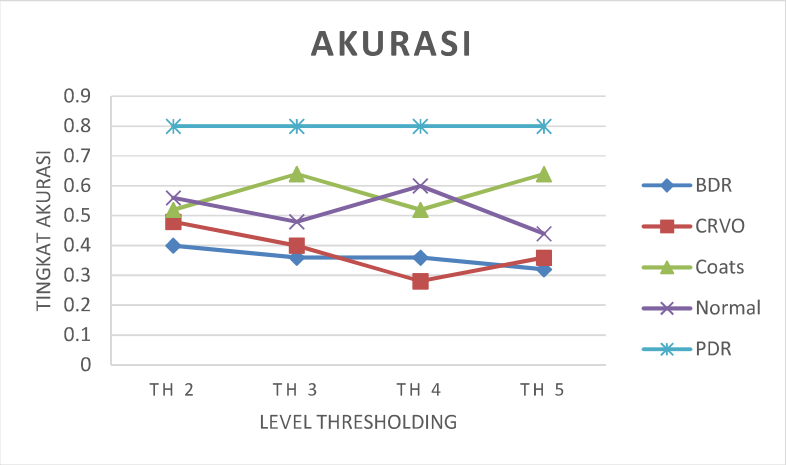
Pada hasil dari Gambar 4.14 yaitu grafik dari hasil nilai spesifikasi dari 5 kelas penyakit berdasarkan th 2, th 3, th 4 dan th5. Pada grafik spesifikasi, hasilnya dihitung berdasarkan citra diklasifikasi penyakit BDR padahal non BDR begitupun juga untuk kelas yang lain. Hasil spesifikasi terbaik pada kelas BDR adalah pada th 2. Karena pada th 2 nilai dari TN yang berarti citra pada kelas BDR yang dipisahkan yang bukan kelas BDR dpaat dilihat pada Tabel 4.7 . Pada FP yaitu citra tersebut sebenarnya bukan penyakit BDR tetapi citra tersebut terklasifikasi sebagai BDR sebanyak 13 citra. Sehingga mendapatkan hasil Spesifikasi sebesar 35%. Pada kelas CRVO spesifikasi terbaik terdapat pada th 3, karena pada TN yaitu citra pada

kelas CRVO yang dipisahkan yang bukan kelas CRVO pada Th 3 sebanyak 6 citra. Pada FP yaitu citra tersebut sebenarnya bukan penyakit CRVO tetapi citra tersebut terklasifikasi sebagai CRVO sebanyak 14 citra. Sehingga mendapatkan spesifikasi 30%. Pada kelas Coats, nilai spesifikasi tebaik terdapat pada th 3 dan 5. Karena citra Coats pada th 3 dan th 5 terdapat citra yang sebenarnya bukan penyakit Coats tetapi data itu terklasifikasi sebagai Coats sebanayk 13 citra. Pada FP yaitu citra tersebut sebenarnya bukan penyakit Coats tetapi citra tersebut terklasifikasi sebagai Coats sebanyak 7 citra. Sehingga menghasilkan spesifikasi sebesar 65%. Pada kelas Normal, nilai spesifikasi terbaik terdapat pada th 2 dan th 4, karena pada TN yaitu citra pada kelas Normal yang dipisahkan yang bukan kelas Normal sebanyak 12 citra dan FP sebanyak 7 citra, sehingga mendapatkan spesifikasi sebesar 60%. Pada kelas PDR, th terbaik yang dapat digunakan masih ambigu. Karena dari th 2, th 3, th 4 dan th 5 semua cita PDR maupun non PDR tidak dapat dikenali. Tetapi, TN yaitu citra pda kelas PDR yang bukan kelas PDR dapat dipisahkan dengan baik. Sehingga spesifikasi yang didapatkan sebesar 80%.



Gambar 4.15 Grafik Sensitivitas dari th 2, th 3, th 4, th 5 pada 5 kelas penyakit

Pada grafik sensitivitas dihitung jumlah citra perkelas yang diprediksi salah. Untuk kelas BDR, sensitivitas yang tebaaik yang dihasilkan adalah pada th 3, th 4 dan th 5. Hasil yang didapatkan adalah 100%. Pada kelas CRVO sensitivitas terbaik didapatkan pada th 2, th 4 dan th 5. Hasil didapatkan sebesar 100%. Untuk kelas Coats sensitivitas terbaik terdapat pada semua th sebesar 60%. Untuk kelas Normal, sensitivitas terbaik didapatkan pada th 3, th dan th 5 sebesar 60%. Sedangkan untuk kelas PDR sensitivitas yang didapatkan 0. Karena pada kelas PDR, tidak ada 1 citra pun yang dapat diklasifikasi.



Gambar 4.16 Grafik Hasil Akurasi dari th 2, th 3, th 4, th 5 pada 5 kelas penyakit

Pada Gambar 4.16, hasil akurasi dihitung dengan meghitung jumlah citra yang dipisahkan dengan benar. Untuk kelas BDR, akurasi terbaik didapatkan pada th 2 yaitu sebesar 40%. Sedangkan untuk kelas CRVO akurasi terbaik didapatkan pada th 2 juga yaitu sebesar 48%. Untuk kelas Coats akurasi terbaik terdapat pada th 3 dan th 5 yaitu sebesar 64%. Pada kelas Normal Akurasi terbaik terdapat pada th th 4 yaitu sebesar 60%. Untuk kelas PDR akurasi terbaik didapatkan pada seluruh th yaitu sebesar 80%. Pada kelas PDR, hasil akurasi yang didapatkan besar yaitu 80% karena, terdapat banyak citra yang dapat dipisahkan dengan benar.

C. **Hasil pengujian klasifikasi tanpa menghitung nilai kesalahan pada perangkat lunak**

Hasil klasifikasi diuji dengan mengukur kinerja dari perangkat lunak saja dengan menggunakan 25 citra penyakit retina tanpa menghitung nilai kesalahan (FP). Sehingga dapat dijelaskan :

- 5 Data uji BDR, CRVO, Coats, Normal, PDR
- 20 Data uji yang bukan BDR, CRVO, Coats, Normal, PDR

1. **Hasil pengujian dengan menggunakan th 2**

Pengujian dengan data sekunder dilakukan dengan menggunakan bobot yang sudah di hasilkan dengan menggunakan th 2. Hasil dari th 2 dapat ditunjukkan dengan Tabel 4.23.

Tabel 4.23. Nilai Performa klasifikasi dengan Confusion Matriks untuk th 2

No	Retina	TP	TN	FP	FN	Spesifikasi	Sensitivitas	Akurasi
1	BDR	3	20	0	2	1	0,6	0,92
2	CRVO	5	20	0	0	1	1	1
3	Coats	3	20	0	2	1	0,6	0,92
4	Normal	2	20	0	3	1	0,4	0,88
5	PDR	0	20	0	5	1	0	0,8

Dengan melihat hasil dari perhitungan yang hanya menghitung kinerja perangkat lunak saja. Dimana TN yaitu data yang dipisahkan dengan benar yang bukan merupakan kelas BDR sebesar 20 citra. Untuk FP nya 0 karena kesalahan perangkat lunak tidak dihitung terhadap ground truth. Sehingga menghasilkan Spesifikasi sebesar 100%. Pada FN yang berarti data yang diprediksi salah untuk kelas BDR pada perangkat lunak padahal ground truth menyatakan bahwa data tersbeut BDR ada sebanyak 2 citra. Nilai yang dihasilkan untuk Sensitivity sebesar 60%. Untuk TP pada kelas BDR yang artinya citra dipisahkan dengan benar oleh perangkat lunak dan ground truth sebanyak 3 citra. Jadi untuk akurasi yang didapatkan adalah 92%. Pada kasus ini menunjukkan kinerja dari perangkat lunak dapat memisahkan citra BDR lumayan baik. Tetapi, masih banyak citra BDR yang tidak terdeteksi atau lolos dari proses pengujian dan pemisahan.

2. Hasil pengujian dengan menggunakan th 3

Pengujian dengan data sekunder dilakukan dengan menggunakan bobot yang sudah di hasilkan dengan menggunakan th 3. Hasil dari th 3 dapat ditunjukkan dengan Tabel 4.24.

Tabel 4.24. Nilai Performa klasifikasi dengan Confusion Matriks untuk th 3

No	Retina	TP	TN	FP	FN	Spesifikasi	Sensitivitas	Akurasi
1	BDR	5	20	0	0	1	1	1
2	CRVO	4	20	0	1	1	0,8	0,96
3	Coats	3	20	0	2	1	0,6	0,92
4	Normal	3	20	0	2	1	0,6	0,92
5	PDR	0	20	0	5	1	0	0,8

Dengan melihat hasil dari perhitungan yang hanya menghitung kinerja perangkat lunak saja, Dimana TN yaitu data yang dipisahkan dengan benar yang bukan merupakan kelas CRVO sebesar 20 citra. Dimana dari data tersebut yaitu pada kelas CRVO terdapat 4 citra yang terklasifikasi dengan benar sebagai CRVO. FP yang berarti data tersebut sebenarnya bukan CRVO tetapi data itu terklasifikasi sebagai BDR ada 0. Karena yang dihitung hanya data yang benar terklasfikasi oleh perangkat lunak saja. Sehingga menghasilkan spesifikasi sebesar 100%. Pada FN yang berarti citra yang diprediksi salah untuk kelas CRVO pada perangkat lunak padahal ground truth menyatakan bahwa citra tersbeut CRVO ada sebanyak 1 citra. Sehingga menghasilkan sensitivitas sebesar 80%. Jadi untuk akurasi yang didapatkan adalah 96%.Pada kasus ini menunjukkan kinerja dari perangkat lunak dapat memisahkan citra CRVO dengan baik. Tetapi, masih ada citra CRVO yang tidak terdeteksi atau lolos dari proses pengujian dan pemisahan.

3. Hasil pengujian dengan menggunakan th 4

Pengujian dengan data sekunder dilakukan dengan menggunakan bobot yang sudah di hasilkan dengan menggunakan th 4. Hasil dari th 4 dapat ditunjukkan dengan tabel 4.25.

Tabel 4.25. Nilai Performa klasifikasi dengan Confusion Matriks untuk th 4

No	Retina	TP	TN	FP	FN	Spesifikasi	Sensitivitas	Akurasi
1	BDR	5	20	0	0	1	1	1
2	CRVO	5	20	0	0	1	1	1
3	Coats	3	20	0	2	1	0,6	0,8
4	Normal	3	20	0	2	1	0,6	0,8
5	PDR	0	20	0	5	1	0	0,8

Dengan melihat hasil dari perhitungan yang hanya menghitung kinerja perangkat lunak saja. Dimana TN yaitu data yang dipisahkan dengan benar yang bukan merupakan kelas Coats sebesar 20 citra. Untuk FP nya 0 karena kesalahan perangkat lunak tidak dihitung terhadap ground truth. Sehingga menghasilkan Spesifikasi sebesar 100%. Pada FN yang berarti data yang diprediksi salah untuk kelas Coats pada perangkat lunak padahal ground truth menyatakan bahwa citra tersbeut BDR ada sebanyak 2 citra. Nilai yang dihasilkan untuk Sensitivity sebesar 60%. Untuk TP pada kelas Coats yang artinya citra dipisahkan dengan benar oleh perangkat lunak dan ground truth sebanyak 3 citra. Jadi untuk akurasi yang didapatkan adalah 80%. Pada kasus ini menunjukkan kinerja dari perangkat lunak dapat memisahkan citra Coats lumayan baik. Tetapi, masih ada citra Coats yang tidak terdeteksi atau lolos dari proses pengujian dan pemisahan.

4. Hasil pengujian dengan menggunakan th 5

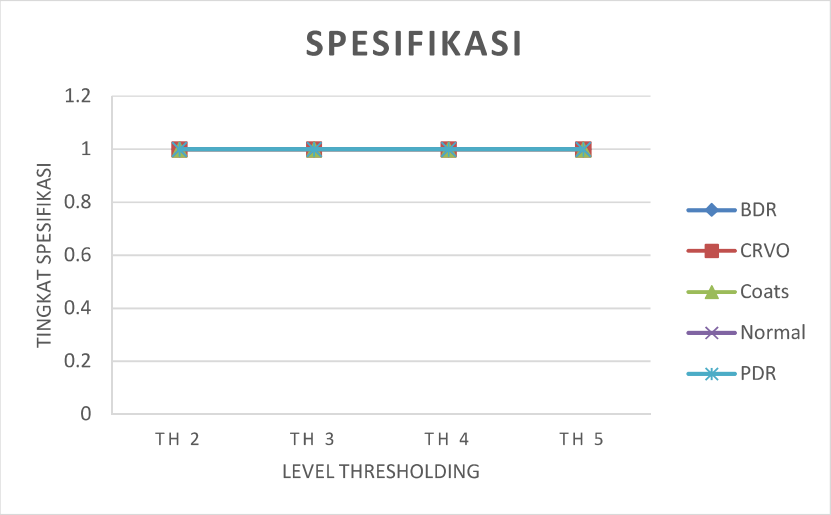
Pengujian dengan data sekunder dilakukan dengan menggunakan bobot yang sudah di hasilkan dengan menggunakan th 5. Hasil dari th 5 dapat ditunjukkan dengan tabel 4.26.

Tabel 4.26. Nilai Performa klasifikasi dengan Confusion Matriks untuk th 5

No	Retina	TP	TN	FP	FN	Spesifikasi	Sensitivitas	Akurasi
1	BDR	5	20	0	0	1	1	1
2	CRVO	5	20	0	0	1	1	1
3	Coats	3	20	0	2	1	0,6	0,92
4	Normal	2	20	0	3	1	0,4	0,88
5	PDR	0	20	0	0	1	0	0,8

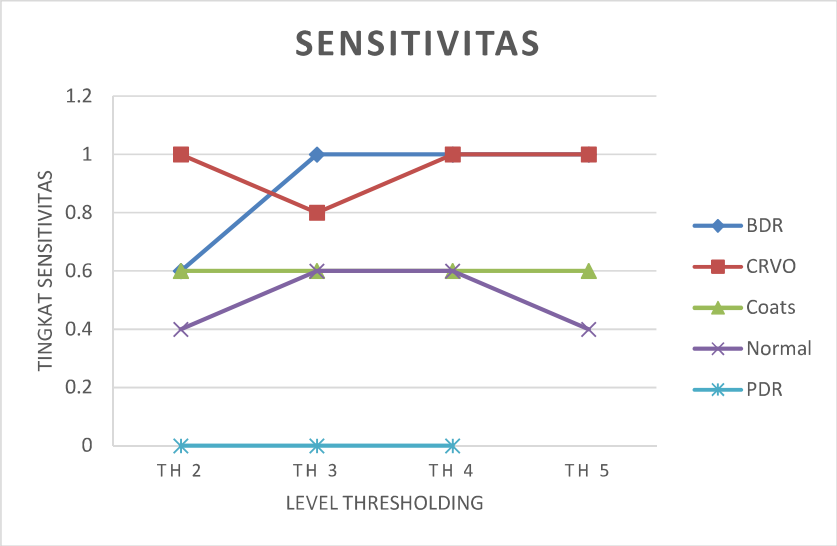
Dengan melihat hasil dari perhitungan yang hanya menghitung kinerja perangkat lunak saja. Dimana TN yaitu data yang dipisahkan dengan benar yang bukan merupakan kelas Normal sebesar 20 citra. Untuk FP nya 0 karena kesalahan perangkat lunak tidak dihitung terhadap ground truth. Sehingga menghasilkan Spesifikasi sebesar 100%. Pada FN yang berarti data yang diprediksi salah untuk kelas Normal pada perangkat lunak padahal ground truth menyatakan bahwa citra tersbeut Coats ada sebanyak 3 citra. Nilai yang dihasilkan untuk Sensitivity sebesar 40%. Untuk TP pada kelas Normal yang artinya citra dipisahkan dengan benar oleh perangkat lunak dan ground truth sebanyak 2 citra. Jadi untuk akurasi yang didapatkan adalah 88%. Pada kasus ini menunjukkan kinerja dari perangkat lunak dapat memisahkan citra Normal lumayan baik. Tetapi, masih ada citra BDR yang tidak terdeteksi atau lolos dari proses pengujian dan pemisahan.

Untuk melihat perbandingan klasifikasi yang dilakukan dengan mengukur nilai Spesifikasi, Sensitivitas dan Akurasi yang menggunakan th 2, th 3, th 4, th 5, bisa dilihat pada Gambar 4.17, 4.18 dan 4.19:



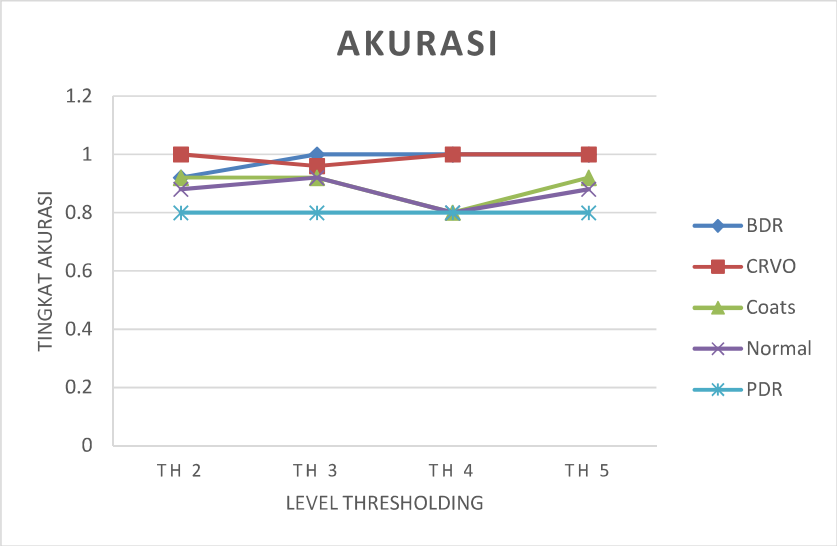
Gambar 4.17 Grafik Hasil Spesifikasi dari th 2, th 3, th 4, th 5 pada 5 kelas penyakit

Pada nilai spesifikasi tanpa menghitung jumlah kesalahan untuk semua th pada kelas penyakit BDR, CRVO, Coats, Normal dan PDR yaitu sebesar 100%.



Gambar 4.18 Grafik Hasil Sensitivitas dari th 2, th 3, th 4, th 5 pada 5 kelas penyakit

Pada nilai sensitivas, untuk kelas BDR sensitivitas terbaik di dapatkan pada th 3, th 4, dan th 5 yaitu sebesar 100%. Untuk kelas CRVO, sensitivitas terbaik didapatkan pada th 2, th 4 dan th 5 yaotu sebesar 100%. Untuk kelas Coats terdapat pada pada semua th yaitu sebesar 60%. Pada kelas Normal, sensitivitas terbaik dihasilkan pada th 3 dan th 5 sebesar 60%. Untuk kelas PDR hasil yang didaoatkan 0 pada setiap th sehingga sensitivitas yang didapatkan juga 0%.

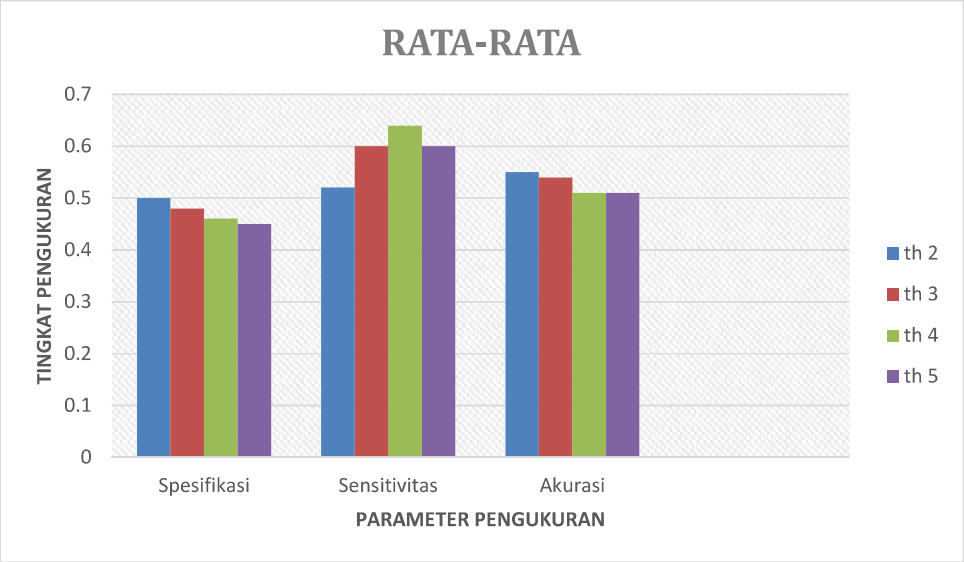


Gambar 4.19 Grafik Hasil Akurasi dari th 2, th 3, th 4, th 5 pada 5 kelas penyakit

Pada Gambar 4.19 akurasi terbaik yang didapatkan pada kelas BDR yaitu pada th 3, th 4 dan th 5 yaitu sebesar 100%. Untuk kelas CRVO akurasi terbaik yang dihasilkan pada th 4 dan th 5 yaitu sebesar 100%. Pada kelas Coats, akuras yang terbaik didapatkan semua th yaotu 92%. Untuk kelas normal akurasi terbaik didapatkan pada th 3 yaitu sebesar 88%. Pada kelas PDR hasil akurasi yang didapatkan pada semua th yaitu 80%.

D. Analisis Hasil Pengujian

Pada hasil rata-rata yang telah dihasilkan, nilai spesifikasi terbaik terdapat pada th 2 dengan hasil 50%. Untuk menghitung spesifikasi, misalkan BDR dihitung dengan TN yaitu data yang dipisahkan dengan benar yang bukan merupakan penyakit BDR. Hal tersebut menyebabkan nilai spesifikasi untuk nilai kesalahan perangkat lunak pada th 2 sedikit. Sehingga spesifikasi yang didapatkan lebih besar dari th lainnya Gambar 4.20.

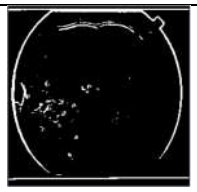
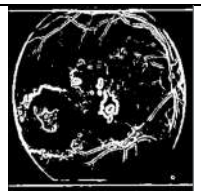
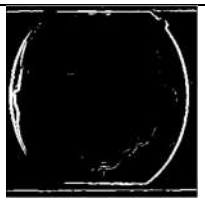
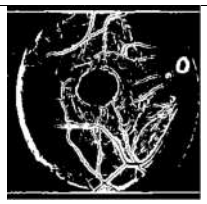
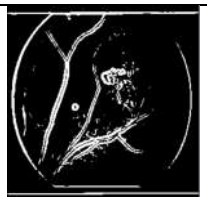


Gambar 4.20 Grafik Rata-rata spesifikasi, sensitivitas, dan akurasi

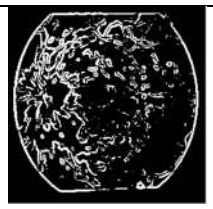
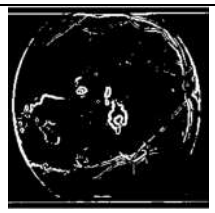
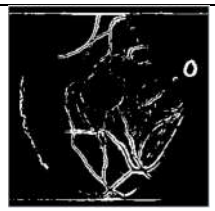
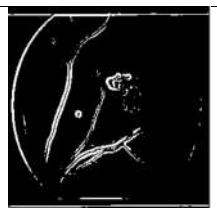
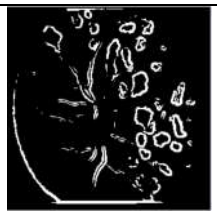
Dari perhitungan ini dapat disimpulkan semakin kecil nilai kesalahan pada perangkat lunak, semakin kecil pula nilai spesifikasinya. Begitu juga sebaliknya, semakin kecil nilai kesalahan pada perangkat lunak, semakin besar pula nilai spesifikasinya. Sedangkan untuk nilai sensitivitasnya juga terdapat pada th 4 dengan hasil 64%. Pada sensitivitas, dihitung jumlah data yang diprediksi salah saja. Misalkan ada 5 citra BDR, tetapi yang diklasifikasi hanya 3 saja. Hal ini menyebabkan nilai sensitivitas yang dihasilkan rata-rata menghasilkan nilai yang tinggi. Tetapi pada akurasi terdapat pada th 2 sebesar 55%. Karena pada saat menghitung nilai akurasi, dihitung jumlah citra yang salah diklasifikasi oleh perangkat lunak dengan benar. Sehingga, dapat disimpulkan bahwa, semakin besar data yang di pisahkan dengan benar maka semakin besar juga akurasi yang didapatkan. Sebaliknya, semakin kecil data yang dipisahkan dengan benar, semakin kecil juga hasil akurasi yang didapatkan.

Untuk melihat citra penyakit yang gagal dikenali pada setiap th, dapat dilihat pada Tabel 4.27 sampai Tabel 4.30.

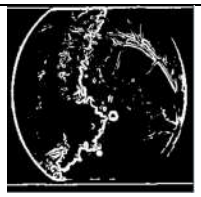
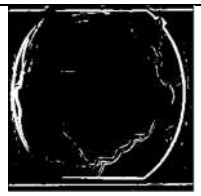
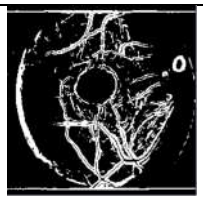
Tabel 4.27. Data citra penyakit retina gagal dengan Th 2

No	Citra Retina	Kelas	Dikenali Sebagai	Banyak Kegagalan
1		BDR	Normal	2
2		Coats	Normal	2
3		Normal	BDR	1
4		Normal	Coats	2
5		PDR	Normal	4
6		PDR	Coats	1

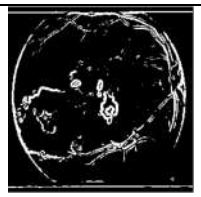
Tabel 4.28. Data citra penyakit retina gagal dengan Th 3

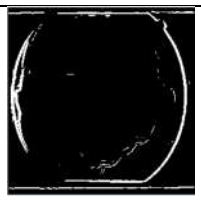
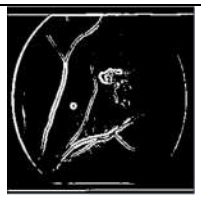
No	Citra Retina	Kelas	Dikenali Sebagai	Banyak Kegagalan
1		CRVO	Coats	1
2		Coats	Normal	2
3		Normal	BDR	2
4		PDR	Normal	4
5		PDR	BDR	1

Tabel 4.29. Data citra penyakit retina gagal dengan Th 4

No	Citra Retina	Kelas	Dikenali Sebagai	Banyak Kegagalan
1		Coats	Normal	2
2		Normal	BDR	1
3		Normal	Coats	1
4		PDR	Normal	5

Tabel 4.30. Data citra penyakit retina gagal dengan Th 5

No	Citra Retina	Kelas	Dikenali Sebagai	Banyak Kegagalan
1		Coats	Normal	2

2		Normal	BDR	2
3		Normal	Coats	1
4		PDR	Normal	4
5		PDR	BDR	1

Dari hasil data yang telah diuji, berdasarkan dari tabel kegagalan dengan menggunakan Th 2, Th 3, Th 4. Th 5 terdapat berbagai macam pola dan bentuk sehingga bobot yang dihasilkan tidak sesuai. Penyebab terjadi kegagalan misalnya untuk kelas PDR karena pola ciri dari penyakit PDR tidak sama, seperti pendarahan ke dalam substansi seperti jelly di tengah mata retina atau disamping kiri kanan retina. Pendarahan seperti jelly tidak selalu berada di tengah, kanan atau kiri retina. Selain itu pendarahan seperti jelly tersebut banyaknya tidak sama untuk data PDR 1 sama lainnya. Ada beberapa penyakit retina yang tidak memiliki pendarahan seperti jelly tetapi menurut ground truth data tersebut merupakan data penyakit PDR.

Analisis Hasil Klasifikasi

Tabel 4.31. menyajikan hasil penggunaan metode Multilevel Thresholding dan klasifikasi penyakit kelainan retina dengan CNN. Pada saat, Thresholding level 2(Th2), specificity mencapai 100% karena tidak terdapat kesalahan dalam hasil klasifikasi, sedangkan untuk sensitivity menghasilkan nilai 60% disebabkan ada 2 citra yang salah mendeteksi kelas BDR dan nilai akurasi mencapai 92%. Percobaan klasifikasi untuk 3 level sudah mencapai masing-masing 100% untuk specificity, sensitivity dan akurasi. Perbandingan kinerja antara metode yang diusulkan dilakukan dengan metode lain, yaitu Two-Field Mydriatic Fundus photography yang dilakukan (Kumar, Deepak, Sathar, Sahasranamam, & Kumar, 2016) menggunakan klasifikasi interpolasi bi-cubic dan model waterfall.

Tabel 4.31. Perbandingan Kinerja antara Metode yang diusulkan dengan metode lain

Metode	**Specificity(%)**	**Sensitivity(%)**	**Accuracy(%)**
Two-Field Mydriatic Fundus Photography (Kumar et al., 2016)	80%	50%	60%
Morphological filter (Kaur & Sinha, 2012)	86,39%	91,28%	NA
Artificial Neural Network (Kanth, Jaiswal, & Kakkar, 2013)	NA	NA	94.11%
Back propagation(Gardner, Keating, Williamson, & Elliott, 1996)	99%	69%	99%
Mathematical morphology methods (Sopharak, Uyyanonvara, Barman, & Williamson, 2008)	71,35%	98,95%	98%
Metode yang diusulkan Th 2	100%	60%	92%
Metode yang diusulkan Th 3	100%	100%	100%
Metode yang diusulkan Th 4	100%	100%	100%
Metode yang diusulkan Th 5	100%	100%	100%

BAB V. KESIMPULAN

5.1. Kesimpulan

1. Segmentasi yang dihasilkan oleh Multilevel thresholding memberikan output yang sangat baik. Semakin tinggi Th yang diberikan, maka nilai PSNR semakin tinggi yaitu untuk thresholding 2, 3, 4, dan 5 berturut-turut nilai PSNR adalah 38,60; 39,63; 41,03 dan 42,12.
2. GLCM dapat digunakan untuk mengumpulkan informasi vec\ktor. Efek dari perubahan parameter sudut θ di GLCM dianalisis bersama dengan parameter jarak d untuk pengenalan berbagai pola. Diamati bahwa sudut kontras dan energi minimum sedangkan sudut korelasi dan homogenitas maksimum, secara independen cukup memadai dalam menentukan arah suatu citra. Hal Ini bisa diterapkan pada citra tekstur lain yang memiliki sifat yang sangat terarah sehingga memberikan informasi tekstur yang lebih signifikan secara rotasional invarian. Nilai rata-rata empat properti yang lebih besar dari kelima penyakit setelah diuji yaitu penyakit *Central Retinal Vein Occlusion* dengan kontras = 0,31, korelasi = 0,98, energi = 0,19, serta homogenitas = 0,93. Dan nilai rata-rata empat properti yang lebih kecil yaitu penyakit *Proliverative Diabetic Retinopathy* dengan kontras = 0,21, korelasi = 0,98, energi = 0,15 serta homogenitas = 0,93.
3. Hasil klasifikasi menggunakan metode *Self-Organizing Map* didapatkan bahwa semakin rendah nilai dari *learning rate* maka nilai keakuratan semakin besar. Dengan rentang nilai 0 sampai 0,006 dan nilai max_iterasi sebesar 100, dapat dilihat bahwa pada *learning rate* 0.6 nilainya sangatlah rendah yaitu 0,002, *learning rate* 0,3 nilainya meningkat yaitu 0,004, *learning rate* 0,15 nilainya juga meningkat yaitu 0,006 dan *learning rate* 0,075 nilainya menurun kembali yaitu 0,004.
4. Algoritma Multi-Layer Perceptrons dapat diterapkan untuk identifikasi kelainan retina menggunakan ekstraksi fitur bentuk. Semakin tinggi nilai

epoch, maka nilai sensitivity, specificity dan akurasi yang diperoleh tinggi. Nilai rata-rata sensitivity, specificity dan akurasi berturut-turut yaitu 80,95%, 85,71% dan 83,33%.

5. Lamanya waktu proses klasifikasi penyakit menggunakan BP sangat berpengaruh pada jumlah epoch yang digunakan. Pada proses klasifikasi menggunakan 100 epoch didapatkan waktu proses klasifikasi 14,8 detik sedangkan pada epoch 500 mendapatkan waktu proses klasifikasi 30,35 dan pada kelasifikasi menggunakan 1000 epoch didapatkan waktu proses klasifikasi meningkat menjadi 39,91 detik. Berdasarkan pengujian yang jumlah epoch berpengaruh pada hasil akurasi klasifikasi penyakit menggunakan BP yaitu pada pengujian dengan menggunakan epoch 100 akurasi yang dihasilkan yaitu 0,4 sedangakan pada epoch 500 dan 1000 meningkat menjadi 0,8. Berdasarkan pengujian yang dilakukan dengan menggunakan learning rate 0,01 dan 0,001 dapat disimpulkan bahwa penggunaan learning rate 0,01 menghasilkan nilai loss atau kesalahan yang lebih kecil dibandingkan penggunaan learning rate 0,001 yaitu learning rate dengan nilai 0,01 mendapatkan nilai yang kecil yaitu 5,132e-04 sedangkan learning rate dengan nilai 0,01 mendapatkan nilai yang kecil yaitu 0,008. sehingga hasil loss yang terbaik yaitu dengan menggunakan learning rate 0,01.
6. Perhitungan klasifikasi dengan metode CNN, diperoleh rata-rata akurasi yang didapatkan relatif tinggi untuk Th 2, Th 3, Th 4 dan Th 5, sebesar 90%, 92%, 88% dan 92%, sedangkan untuk sensitivity sebesar 52%, 60%, 64% dan 60%, walaupun nilai rata-rata specificity masing-masing 100%.

5.2. Kontribusi

Ada 3(tiga) kontribusi yang diperoleh dari penelitian ini yaitu pada proses segmentasi, ekstraksi dan klasifikasi dengan uraian sebagai berikut:

1. Metode baru yang menggabungkan penggunaan metode multi level thresholding Otsu dan algoritma improved harmonic search algorithm, yang disingkat

menjadi MT-IHSA. Pengukuran keberhasilan metode yang diusulkan menggunakan nilai PSNR. Hasil pengujian menggunakan dataset retina STARE diperoleh nilai PSNR lebih tinggi dibandingkan citra lidah, lena, babon dan kameraman. Rata-rata nilai PSNR dari citra retina Stare adalah 40,34 dB. Hasil percobaan dari metode yang diusulkan ini dibandingkan dengan metode-metode terkini lainnya yaitu (B. Zhang, Kumar, & Zhang, 2013) yang menggunakan segmentasi warna, tekstur dan fiur geometri(CTGF), (Oliva et al., 2013) menggunakan Multilevel Thresholding harmony search algorithm (MT-HSA), dan (R. Singh, Agarwal, Kashyap, & Bhattacharya, 2016) yang menggunakan Multilevel Thresholding Firefly Algorithm(MT-FA) dan Multilevel Thresholding Social Spider Algorithm (MT-SSA) menghasilkan bahwa teknik baru yang diusulkan yaitu MIHSA mendapatkan nilai PSNR tertinggi yaitu 40,342 dB.

2. Perbaikan akurasi untuk klasifikasi penyakit kelainan retina dilakukan pada proses ekstraksi fitur. Penelitian ini mengusulkan metode ekstraksi berdasarkan tektur, bentuk dan warna sesuai dengan ciri-ciri setiap penyakit kelainan retina. Penyakit kelainan retina ada yang memiliki ciri bentuk seperti perubahan bentuk seperti mokula, pembuluh darah dan fovea, untuk ciri tektur seperti tektur pembuluh darah yang lebar dan sempit sebagai suatu kelainan, sedangkan ciri warna seperti ada bintik dan pembuluh darah yang pecah yang menimbulkan perubahan warna. Hasil perhitungan QEF untuk ekstraksi pembuluh darah menggunakan dataset STARE dan dibandingkan dengan hasil yang diperoleh (B. Zhang et al., 2010), (Nguyen et al., 2013) dan (Yin et al., 2015) menunjukkan hasil bahwa metode yang diusulkan mendapatkan nilai QEF yang paling tinggi dibandingkan metode lainnya. Perhitungan dengan fungsi CAL memperoleh nilai 71,32 % lebih tinggi dibandingkan dengan metode lainnya.
3. Proses klasifikasi penyakit kelainan retina diusulkan dengan menggunakan metode CNN. Hasil penggunaan metode Multilevel Thresholding dan klasifikasi penyakit kelainan retina dengan CNN. Pada saat, Thresholding level 2(Th2), specificity mencapai 100% karena tidak terdapat kesalahan dalam hasil klasifikasi, sedangkan untuk sensitivity menghasilkan nilai 60% disebabkan ada

2 citra yang salah mendeteksi kelas BDR dan nilai akurasi mencapai 92%. Percobaan klasifikasi untuk 3 level sudah mencapai masing-masing 100% untuk specificity, sensitivity dan akurasi. Perbandingan kinerja antara metode yang diusulkan dilakukan dengan metode lain, yaitu Two-Field Mydriatic Fundus photography yang dilakukan (Kumar et al., 2016) menggunakan klasifikasi interpolasi bi-cubic dan model waterfall menghasilkan hasil yang terbaik.

5.3. Saran

Untuk penelitian selanjutnya, pengembangan lebih lanjut mengenai metode Hyrid Multilevel Thresholding dan Convolutional Neural Network dapat ditambahkan ektraksi fitur yang dapat disesuaikan dengan rotasi dan translasi yang berbeda bergantung dari posisi pengambilan citra dan kamera. Intensitas cahaya di ruang pengambilan citra menjadi penyebab kegagalan klasifikasi yang dilakukan sehingga dibutuhkan teknik khusus untuk mengatasi noise akibat ini. Selain itu, posisi pengambilan citra harus stabil dan diatur sesuai dengan spesifikasi peralatan tersebut.

DAFTAR REFERENSI

Abdelraouf, M., Sonbol, A., & Mohamed, S. (2018). Omentin-1 and diabetic retinopathy in type 2 diabetic patients q. *Alexandria Journal of Medicine*, 1–4. https://doi.org/10.1016/j.ajme.2018.04.003

Al-amri, S. S. (2011). Contrast Stretching Enhancement in Remote Sensing Image. *BIOINFO Sensor Network*, *1*(1).

Al-Bander, B., Al-Nuaimy, W., Williams, B. M., & Zheng, Y. (2018). Multiscale sequential convolutional neural networks for simultaneous detection of fovea and optic disc. *Biomedical Signal Processing and Control*, *40*, 91–101. https://doi.org/10.1016/j.bspc.2017.09.008

Al-najjar, Y. A. Y., & Soong, D. C. (2012). Comparison of Image Quality Assessment : *International Journal of Scientific & Engineering Research*, *3*(8), 1–5.

Anitha, J., Vijila, C. K. S., Hemanth, D. J., & Ahsina, A. (2009). Self Organizing Neural Network Based Pathology Classification in Retinal Images. *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC 2009)*, 0–5.

Antwi, P., Li, J., Boadi, P. O., Meng, J., Shi, E., Deng, K., & Bondinuba, F. K. (2017). Estimation of biogas and methane yields in an UASB treating potato starch processing wastewater with backpropagation artificial neural network. *Bioresource Technology*, *228*, 106–115. https://doi.org/10.1016/j.biortech.2016.12.045

Arora, S., Acharya, J., Verma, A., & Panigrahi, P. K. (2008). Multilevel thresholding for image segmentation through a fast statistical recursive algorithm. *Pattern Recognition Letters*, *29*(2), 119–125. https://doi.org/10.1016/j.patrec.2007.09.005

Arunachalam, H. B., Mishra, R., Armaselu, B., & Martinez, M. (2017). Computer aided image segmentation and classification for viable and non-viable tumor identification in osteosarcoma, 195–206.

Banimelhem, ~O, & Yahya, ~Y.~A. (2011). Multi-Thresholding Image Segmentation Using Genetic Algorithm. *World Congress in Computer Science, Computer Engineering, and Applied Computing*, (February).

Barkana, B. D., Saricicek, I., & Yildirim, B. (2017). Performance analysis of descriptive statistical features in retinal vessel segmentation via fuzzy logic, ANN, SVM, and classifier fusion. *Knowledge-Based Systems*, *118*, 165–176. https://doi.org/10.1016/j.knosys.2016.11.022

BY Rishav Kansal, B., Kiran Turaka, M., & Carol L. Shields, M. (2011). Coats Disease: Classification and Treatment. *Retina Oncology Case Reports In Ocular Oncology*.

Cao, L., Ding, S., Fu, X., & Chen, L. (2016). Otsu multilevel thresholding segmentation based on quantum particle swarm optimisation algorithm. *Int. J. Wireless and Mobile Computing*, *10*(3), 272–277.

Cengil, E., Cinar, A., & Guler, Z. (2017). A GPU-based convolutional neural network approach for image classification. *2017 International Artificial Intelligence and Data Processing Symposium (IDAP)*, 1–6.

https://doi.org/10.1109/IDAP.2017.8090194
Cerbulescu, B. J., & Stefan, C. (2003). [Coats disease]. *Oftalmologia*, *58*(3), 21–25. https://doi.org/10.1016/B978-1-4160-3167-3.50063-2
Ch'ng, E. (2015). Big Data and Visual Analytics Lab, (April 2018). https://doi.org/10.1007/978-3-319-63917-8
Cheddad, A., Condell, J., Curran, K., & McKevitt, P. (2010). A hash-based image encryption algorithm. *Optics Communications*, *283*(6), 879–893. https://doi.org/10.1016/j.optcom.2009.10.106
Chen, K., Zhou, Y., Zhang, Z., Dai, M., Chao, Y., & Shi, J. (2016). Multilevel Image Segmentation Based on an Improved Firefly Algorithm. *Mathematical Problem in Engineering*, *2016*, 1–12.
Condurache, A. P., & Mertins, A. (2012). Segmentation of retinal vessels with a hysteresis binary-classification paradigm. *Computerized Medical Imaging and Graphics*, *36*(4), 325–335. https://doi.org/10.1016/j.compmedimag.2012.02.002
Cuevas, E., Ortega-Sánchez, N., Zaldivar, D., & Pérez-Cisneros, M. (2012). Circle detection by Harmony Search Optimization. *Journal of Intelligent and Robotic Systems: Theory and Applications*, *66*(3), 359–376.
Cui, K., Zhang, S., Liu, X., Yan, Z., Huang, L., Yang, X., … Sang, A. (2018). Inhibition of TBK1 reduces choroidal neovascularization in vitro and in vivo. *Biochemical and Biophysical Research Communications*, *1*, 2–8. https://doi.org/10.1016/j.bbrc.2018.06.003
Dash, J., & Bhoi, N. (2017). A Thresholding Based Technique to Extract Retinal Blood Vessels from Fundus Images. *Future Computing and Informatics Journal*. https://doi.org/10.1016/j.fcij.2017.10.001
Dass, R., & Devi, S. (2012). Image Segmentation Techniques. *IJECT*, *3*(1), 66–70.
Degertekin, S. O. (2012). Improved harmony search algorithms for sizing optimization of truss structures. *Computers and Structures*, *92–93*, 229–241. https://doi.org/10.1016/j.compstruc.2011.10.022
Dillak, R. Y., & Bintiri, M. G. (2012). Identifikasi Fase Penyakit Retinopati Diabetes Menggunakan Jaringan Syaraf Tiruanmulti Layer Perceptron. *Seminar Nasional Informatika (SEMNASIF)*, *2012*(semnasIF), 126–133.
Fourie, J., Mills, S., & Green, R. (2010). Harmony filter : A robust visual tracking system using the improved harmony search algorithm. *Image and Vision Computing*, 1–15. https://doi.org/10.1016/j.imavis.2010.05.006
Franklin, S. W., & Rajan, S. E. (2014). Computerized screening of diabetic retinopathy employing blood vessel segmentation in retinal images. *Biocybernetics and Biomedical Engineering*, *34*(2), 117–124. https://doi.org/10.1016/j.bbe.2014.01.004
Fraz, M. M., Jahangir, W., Zahid, S., Hamayun, M. M., & Barman, S. A. (2017). Multiscale segmentation of exudates in retinal images using contextual cues and ensemble classification. *Biomedical Signal Processing and Control*, *35*, 50–62. https://doi.org/10.1016/j.bspc.2017.02.012
Gardner, G. G., Keating, D., Williamson, T. H., & Elliott, A. T. (1996). Automatic detection of diabetic retinopathy using an artificial neural network: A screening tool. *British Journal of Ophthalmology*, *80*(11), 940–944.

https://doi.org/10.1136/bjo.80.11.940

Geem, Z. W., Kim, J. H., & Loganathan, G. (2001). A New Heuristic Optimization Algorithm : Harmony Search. *Simulation*, *76*(2), 60–68.

Geetharamani, R., & Balasubramanian, L. (2016). Retinal blood vessel segmentation employing image processing and data mining techniques for computerized retinal image analysis. *Biocybernetics and Biomedical Engineering*, *36*(1), 102–118. https://doi.org/10.1016/j.bbe.2015.06.004

Gegúndez-Arias, M. E., Aquino, A., Bravo, J. M., & Marín, D. (2012). A function for quality evaluation of retinal vessel segmentations. *IEEE Transactions on Medical Imaging*, *31*(2), 231–9. https://doi.org/10.1109/TMI.2011.2167982

Ghorbanian, S., Jaulim, A., & Chatziralli, I. P. (2012). Diagnosis and treatment of coats' disease: A review of the literature. *Ophthalmologica*, *227*(4), 175–182. https://doi.org/10.1159/000336906

Gonzalez, R. C., & Woods, R. E. (2007). *Digital Image Processing Third Edition*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA.

Guo, Z., Yue, X., Liu, G., Wang, S., & Li, K. (2015). An improved harmony search algorithm for multilevel image segmentation. *ICIC Express Letters*, *9*(9), 2531–2536.

Haralick, R. M., Shanmugam, K., & Dinstein, I. (1973). Textural Features for Image Classification. *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, SMC-3*(6), 610–621.

Hartono, B. (2014). Analisa Teknik Adaptive Histogram Equalization dan Contrast Stretching untuk Perbaikan Kualitas Citra. *Jurnal Teknologi Informasi DINAMIK*, *19*(1), 1–10.

Holambe, S. N., & Kumbhar, P. G. (2016). Comparison between Otsu ' s Image Thresholding Technique and Iterative Triclass. *IJCTT*, *33*(2).

Huang, H., Ma, H., JW van Triest, H., Wei, Y., & Qian, W. (2017). Automatic detection of neovascularization in retinal images using extreme learning machine. *Neurocomputing*. https://doi.org/10.1016/j.neucom.2017.03.093

Jadhav, A. S., & Patil, P. B. (2018). Detection of Blood vessels in Retinal images for diagnosis of Diabetics. *2018 2nd International Conference on Inventive Systems and Control (ICISC)*, (Icisc), 888–891.

Kalavathi, P. (2013). Brain Tissue Segmentation in MR Brain Images using Multiple Otsu's Thresholding Technique, (Iccse), 639–642.

Kanth, S., Jaiswal, A., & Kakkar, M. (2013). Identification of different stages of Diabetic Retinopathy using artificial neural network. *2013 Sixth International Conference on Contemporary Computing (IC3)*, 479–484. https://doi.org/10.1109/IC3.2013.6612243

Karthigeyan, P., Raja, M. S., Hariharan, R., Prakash, S., Delibabu, S., & Gnanaselvam, R. (2015). Comparison of Harmony Search Algorithm, Improved Harmony Search Algorithm with Biogeography Based Optimization Algorithm for Solving Constrained Economic Load Dispatch Problems. *Procedia Technology*, *21*, 611–618. https://doi.org/10.1016/j.protcy.2015.10.071

Kasar, M. M., Bhattacharyya, D., & Kim, T.-H. (2016). Face Recognition Using Neural Network: A Review. *International Journal of Security and Its*

Applications, *10*(3), 81–100. https://doi.org/10.14257/ijsia.2016.10.3.08

Kaur, J., & Sinha, H. P. (2012). Automated Detection of Diabetic Retinopathy Using Fundus Image Analysis, *3*(4), 4794–4799.

Khairuzzaman, A. K., & Chaudhury, S. (2017). Multilevel Thresholding using Grey Wolf Optimizer for image segmentation. *Expert Systems With Applications*. https://doi.org/10.1016/j.eswa.2017.04.029

Khairuzzaman, A. K. M., & Chaudhury, S. (2017). Multilevel thresholding using grey wolf optimizer for image segmentation. *Expert Systems with Applications*, *86*, 64–76. https://doi.org/10.1016/j.eswa.2017.04.029

Kim, J., Sangjun, O., Kim, Y., & Lee, M. (2016). Convolutional Neural Network with Biologically Inspired Retinal Structure. *Procedia Computer Science*, *88*, 145–154. https://doi.org/10.1016/j.procs.2016.07.418

Kotte, S., Rajesh Kumar, P., & Injeti, S. K. (2015). An efficient approach for optimal multilevel thresholding selection for gray scale images based on improved differential search algorithm. *Ain Shams Engineering Journal*. https://doi.org/10.1016/j.asej.2016.06.007

Kumar, P. N. S., Deepak, R. U., Sathar, A., Sahasranamam, V., & Kumar, R. R. (2016). Automated Detection System for Diabetic Retinopathy Using Two Field Fundus Photography. *Procedia Computer Science*, *93*(September), 486–494. https://doi.org/10.1016/j.procs.2016.07.237

Kurugollu, F., Sankur, B., & Harmanci, a. . (2001). Color image segmentation using histogram multithresholding and fusion. *Image and Vision Computing*, *19*, 915–928. https://doi.org/10.1016/S0262-8856(01)00052-X

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278–2323. https://doi.org/10.1109/5.726791

Li, J., Yang, H., Chen, L., & Zhi, C. (2017). Image semantic segmentation optimization by Conditional Random Field integrated with Object Clique Potential. *IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, 1–6.

Liu, Y., Mu, C., Kou, W., & Liu, J. (2014). Modified particle swarm optimization-based multilevel thresholding for image segmentation. *Soft Comput*. https://doi.org/10.1007/s00500-014-1345-2

Madane, M., & Thepade, S. (2016). Score Level Fusion Based Bimodal Biometric Identification Using Thepade's Sorted n-ary Block Truncation Coding with Variod Proportions of Iris and Palmprint Traits. *Procedia Computer Science*, *79*, 466–473. https://doi.org/10.1016/j.procs.2016.03.060

Mahdavi, M., Fesanghary, M., & Damangir, E. (2007). An improved harmony search algorithm for solving optimization problems. *Applied Mathematics and Computation*, *188*(2), 1567–1579. https://doi.org/10.1016/j.amc.2006.11.033

Maleki, F., Nooshyar, M., & Fatin, G. Z. (2014). Breast Cancer Segmentation in Digital Mammograms Based on Harmony Search Optimization, 477–484.

Manikandan, S., Ramar, K., Willjuice Iruthayarajan, M., & Srinivasagan, K. G. (2014). Multilevel thresholding for segmentation of medical brain images using real coded genetic algorithm. *Measurement: Journal of the International Measurement Confederation*, *47*(1), 558–568.

https://doi.org/10.1016/j.measurement.2013.09.031

Maryam, H., Mustapha, A., & Younes, J. (2017). A multilevel thresholding method for image segmentation based on multiobjective particle swarm optimization. *IEEE*, 0–5. Retrieved from http://www.springerlink.com/content/e25184421787hp35/

Miljković, D. (2017). Brief Review of Self-Organizing Maps. *MIPRO*, 1252–1257.

Mishra, D., Bose, I., Chandra De, U., & Pradhan, B. (2014). A multilevel image thresholding using particle swarm optimization. *International Journal of Engineering and Technology*, *6*(2), 1204–1211. Retrieved from http://www.scopus.com/inward/record.url?eid=2-s2.0-84900384133&partnerID=40&md5=aded99016152e5a4b17744bc5e119dc3

Morales, S., Naranjo, V., Angulo, J., Legaz-Aparicio, A. G., & Verdú-Monedero, R. (2017). Retinal network characterization through fundus image processing: Significant point identification on vessel centerline. *Signal Processing: Image Communication*, *59*(January 2016), 50–64. https://doi.org/10.1016/j.image.2017.03.013

Mouravliansky, N. A., & Delibasis, K. K. (2004). Multimodal Registration of Retinal Images Using Self Organizing Maps. *IEEE TRANSACTIONS ON MEDICAL IMAGING*, *23*(12), 1557–1563.

Mozaffari, M. H., & Lee, W.-S. (2017). Convergent heterogeneous particle swarm optimisation algorithm for multilevel image thresholding segmentation. *IET Image Processing*, *11*(8), 605–619. https://doi.org/10.1049/iet-ipr.2016.0489

Nadu, T., Thulasi, N., Simonthomas, S., & Asharaf, P. (2014). Automated Diagnosis of Glaucoma using Haralick Texture Features. *ICICES*, (978).

Naidu, M. S. R., Rajesh Kumar, P., & Chiranjeevi, K. (2017). Shannon and Fuzzy entropy based evolutionary image thresholding for image segmentation. *Alexandria Engineering Journal*. https://doi.org/10.1016/j.aej.2017.05.024

Nasr-Esfahani, E., Karimi, N., Jafari, M. H., Soroushmehr, S. M. R., Samavi, S., Nallamothu, B. K., & Najarian, K. (2018). Segmentation of vessels in angiograms using convolutional neural networks. *Biomedical Signal Processing and Control*, *40*, 240–251. https://doi.org/10.1016/j.bspc.2017.09.012

Nayak, C., & Kaur, L. (2015). Retinal Blood Vessel Segmentation for Diabetic Retinopathy Using Multilayered Thresholding. *International Journal of Science and Research*, *4*(6), 1520–1526.

Nguyen, U. T. V., Bhuiyan, A., Park, L. A. F., & Ramamohanarao, K. (2013). An effective retinal blood vessel segmentation method using multi-scale line detection. *Pattern Recognition*, *46*(3), 703–715. https://doi.org/10.1016/j.patcog.2012.08.009

Nivetha, C., Sumathi, S., & Chandrasekaran, M. (2017). Retinal Blood Vessels Extraction and Detection of Exudates Using Wavelet Transform and PNN Approach for the Assessment of Diabetic Retinopathy. *International Conference on Communication and Signal Processing*, 1962–1966.

Oliva, D., Cuevas, E., Pajares, G., Zaldivar, D., & Perez-cisneros, M. (2013). Multilevel Thresholding Segmentation Based on Harmony Search Optimization. *Journal of Applied Mathematics*, *2013*.

Otsu, N. (1979a). A Threshold Selection Method from Gray-Level Histograms. *IEEE*, *C*(1), 62–66.

Otsu, N. (1979b). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, *9*(1), 62–66. https://doi.org/10.1109/TSMC.1979.4310076

Öztürk, Ş., & Akdemir, B. (2018). Application of Feature Extraction and Classification Methods for Histopathological Image using GLCM , Application of Feature Extraction and Classification Methods for and GLCM , Histopathological Image using a SFTA. *Procedia Computer Science*, *132*(Iccids), 40–46. https://doi.org/10.1016/j.procs.2018.05.057

Panchal, P., Bhojani, R., & Panchal, T. (2016). An Algorithm for Retinal Feature Extraction Using Hybrid Approach. *Procedia Computer Science*, *79*, 61–68. https://doi.org/10.1016/j.procs.2016.03.009

Patil, S. A., & Patel, D. A. (2013). An Overview : Improved Harmony Search Algorithm and Its Applications in Mechanical Engineering, *2*(1), 433–444.

Pratt, H., Coenen, F., Broadbent, D. M., Harding, S. P., & Zheng, Y. (2016). Convolutional Neural Networks for Diabetic Retinopathy. *Procedia Computer Science*, *90*(July), 200–205. https://doi.org/10.1016/j.procs.2016.07.014

Qiang, S., & Guoying, L. (2016). An Edge-Detection Method Based on Adaptive Canny Algorithm and Iterative Segmentation Threshold. *ICCSSE*, 64–67.

Sadikoglu, F., & Uzelaltinbulat, S. (2016). Biometric Retina Identification Based on Neural Network. *Procedia Computer Science*, *102*(August), 26–33. https://doi.org/10.1016/j.procs.2016.09.365

Sayood, K. (2002). Statistical evaluation of image quality measures. *Journal of Electronic Imaging*, *11*(April), 206–223. https://doi.org/10.1117/1.1455011

Shang, R., Yuan, Y., Jiao, L., Member, S., & Hou, B. (2017). A Fast Algorithm for SAR Image Segmentation Based on Key Pixels, 1–17.

Shuai, B., Zuo, Z., Wang, B., Wang, G., & Member, S. (2017). Scene Segmentation with DAG-Recurrent Neural Networks, *8828*(c). https://doi.org/10.1109/TPAMI.2017.2712691

Singh, R., Agarwal, P., Kashyap, M., & Bhattacharya, M. (2016). Kapur ' s And Otsu ' s Based Optimal Multilevel Image Thresholding Using Social Spider and Firefly Algorithm, (v), 2220–2224.

Sivakamasundari, J., & Natarajan, V. (2015). Content Based Retinal Image Retrieval Scheme using Harmony Search Algorithm, 607–611. https://doi.org/10.1109/ICCSP.2015.7322559

Somasundaram, K., & Kalavathi, P. (2011). Medical Image Binarization using Square Wave Representation. In *Control, Computation and Information Systems*.

Sopharak, A., Uyyanonvara, B., Barman, S., & Williamson, T. H. (2008). Automatic detection of diabetic retinopathy exudates from non-dilated retinal images using mathematical morphology methods. *Computerized Medical Imaging and Graphics*, *32*(8), 720–727. https://doi.org/10.1016/j.compmedimag.2008.08.009

Sreejini, K. S., & Govindan, V. K. (2015). Improved multiscale matched filter for retina vessel segmentation using PSO algorithm. *Egyptian Informatics*

Journal, *16*(3), 253–260. https://doi.org/10.1016/j.eij.2015.06.004

Stutz, D. (2014). Understanding Convolutional Neural Networks. *Nips 2016*, (3), 1–23. https://doi.org/10.1016/j.jvcir.2016.11.003

Suartika, W., Wijaya, A. Y., & Soelaiman, R. (2016). Klasifikasi Citra Menggunakan Convolutional Neural Network (Cnn) Pada Caltech 101. *Jurnal Teknik ITS*, *5*(1), 65–69. Retrieved from http://repository.its.ac.id/48842/

Suresh, A., & Shunmuganathan, K. L. (2012). Image Texture Classification using Gray Level Co-Occurrence Matrix Based Statistical Features. *European Journal of Scientific Research*, *75*(4), 591–597.

Suryawanshi, V. (2017). Guassian Transformed GLCM Features f or Classifying Diabetic Retinopathy. *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, (1), 1108–1111.

Szeliski, R. (2010). Computer Vision : Algorithms and Applications.

Tan, J. H., Acharya, U. R., Bhandary, S. V., Chua, K. C., & Sivaprasad, S. (2017). Segmentation of optic disc, fovea and retinal vasculature using a single convolutional neural network. *Journal of Computational Science*, *20*, 70–79. https://doi.org/10.1016/j.jocs.2017.02.006

Triwijoyo, B. K., Budiharto, W., & Abdurachman, E. (2017). The Classification of Hypertensive Retinopathy using Convolutional Neural Network. *Procedia Computer Science*, *116*, 166–173. https://doi.org/10.1016/j.procs.2017.10.066

Tuceryan, M., & Jain, A. K. (1998). Texture Analysis. *The Handbook of Pattern Recognition and Computer Vision (2nd Editon)*, 1–41.

Vala, M. H. J., & Baxi, A. (2013). A review on Otsu image segmentation algorithm. *International Journal of Advanced Research in Computer Engineering and Technology*, *2*(2), 387–389.

Vijay, P. P., & Patil, N. C. (2016). Gray Scale Image Segmentation using OTSU Thresholding Optimal Approach. *Journal for Research*, *2*(5), 20–24.

Wang, X., Jiang, X., & Ren, J. (2019). Blood vessel segmentation from fundus image by a cascade classification framework. *Pattern Recognition*, *88*, 331–341. https://doi.org/10.1016/j.patcog.2018.11.030

Wang, Z., & C.Bovix, A. (2002). A Universal Image Quality Index. *IEE Signal Processing Letters*, *20*, 2–5.

Welikala, R. A., Foster, P. J., Whincup, P. H., Rudnicka, A. R., Owen, C. G., Strachan, D. P., & Barman, S. A. (2017). Automated arteriole and venule classification using deep learning for retinal images from the UK Biobank cohort. *Computers in Biology and Medicine*, *90*(September), 23–32. https://doi.org/10.1016/j.compbiomed.2017.09.005

Wen, Y., Zhang, K., Li, Z., & Qiao, Y. (2016). A discriminative feature learning approach for deep face recognition. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 9911 LNCS, pp. 499–515). https://doi.org/10.1007/978-3-319-46478-7_31

Wykoff, C. C., Eichenbaum, D. A., Roth, D. B., Hill, L., Fung, A. E., & Haskova, Z. (2018). Ranibizumab Induces Regression of Diabetic Retinopathy in Most Patients at High Risk of Progression to Proliferative Diabetic Retinopathy.

Ophthalmology Retina, 1–13. https://doi.org/https://doi.org/10.1016/j.oret.2018.06.005

Xiong, L., Li, H., & Xu, L. (2017). An enhancement method for color retinal images based on image formation model. *Computer Methods and Programs in Biomedicine*, *143*, 137–150. https://doi.org/10.1016/j.cmpb.2017.02.026

Xu, X., Ding, W., Abràmoff, M. D., & Cao, R. (2017). An improved arteriovenous classification method for the early diagnostics of various diseases in retinal image. *Computer Methods and Programs in Biomedicine*, *141*, 3–9. https://doi.org/10.1016/j.cmpb.2017.01.007

Yang, Z., Shufan, Y., Li, G., & Weifeng, D. (2016). Segmentation of MRI Brain Images with an Improved Harmony Searching Algorithm. *Biomed Research International*, *2016*.

Yin, B., Li, H., Sheng, B., Hou, X., Chen, Y., Wu, W., … Jia, W. (2015). Vessel extraction from non-fluorescein fundus images using orientation-aware detector. *Medical Image Analysis*, *26*(1), 232–242. https://doi.org/10.1016/j.media.2015.09.002

Yourui, H., & Shuang, W. (2008). Multilevel thresholding methods for image segmentation with Otsu based on QPSO. *Proceedings - 1st International Congress on Image and Signal Processing, CISP 2008*, *3*, 701–705. https://doi.org/10.1109/CISP.2008.76

Zhang, B., Zhang, L., Zhang, L., & Karray, F. (2010). Retinal vessel extraction by matched filter with first-order derivative of Gaussian. *Computers in Biology and Medicine*, *40*(4), 438–445. https://doi.org/10.1016/j.compbiomed.2010.02.008

Zhang, M., Diao, M., & Guo, L. (2017). Convolutional Neural Networks for Automatic Cognitive Radio Waveform Recognition. *IEEE Access*, *5*, 11074–11082. https://doi.org/10.1109/ACCESS.2017.2716191

Zhang, W., Li, R., Deng, H., Wang, L., Lin, W., Ji, S., & Shen, D. (2015). NeuroImage Deep convolutional neural networks for multi-modality isointense infant brain image segmentation. *NeuroImage*, *108*, 214–224. https://doi.org/10.1016/j.neuroimage.2014.12.061

Zhang, X., Cui, J., Wang, W., & Lin, C. (2017). A Study for Texture Feature Extraction of High-Resolution Satellite Images Based on a. *Sensor*, *17*(1474), 1–15. https://doi.org/10.3390/s17071474

Zhao, Z.-Q., Xu, S.-T., Liu, D., Tian, W.-D., & Jiang, Z.-D. (2018). A Review of Image Set Classification. *Neurocomputing*. https://doi.org/10.1016/j.neucom.2018.09.090

Zheng, X., Ye, H., & Tang, Y. (2017). Image bi-level thresholding based on gray level-local variance histogram. *Entropy*, *19*(5), 1–8. https://doi.org/10.3390/e19050191

Zhou, D., & Zhou, H. (2016). Minimisation of local within-class variance for image segmentation. *IET Image Processing*, *10*(2), 608–615. https://doi.org/10.1049/iet-ipr.2015.0773

Zong Woo Geem, Joong Hoon Kim, & Loganathan, G. V. (2001). A New Heuristic Optimization Algorithm: Harmony Search. *Simulation*, *76*(2), 60–68. https://doi.org/10.1177/003754970107600201

Lampiran 1. Daftar Publikasi di Jurnal Internasional

No.	Judul	Nama Jurnal	Vol, No, Tahun	Indeks
1	Hybrid Multilevel Thresholding and Improved Harmony Search Algorithm for Segmentation	International Journal of Electrical and Computer Engineering (IJECE)	Vol. 8, No. 6, December 2018, pp. 4593~4602	Scopus, Q2
2	New Optimization Technique to Extract Facial Features	IAENG International Journal of Computer Science	45:4, IJCS_45_4_04, 7 November 2018	Scopus, Q2
3	Feature Extraction for Retina Image Based on Difference Approaches	Computer Engineering and Applications	Vol. 7, No. 3, October 2018	DOAJ
4	Automatic Detection of Background Diabetic Retinopathy Disease Using Hybrid Multilevel Thresholding and Convolutional Neural Network	Journal of Engineering Science & Technology	In printed, Volume 14, Issue 5 (October 2019	Scopus, Q2

Lampiran 2**.** Hasil Multithresholding Improved Harmony Search pada Data Traning

No	Image	k	Thresholds	STD	PSNR	MEAN
1	1	2	54	0.3598	17.0835	785.8195
		3	11, 57	0.6556	21.5325	1.0339e+03
		4	12, 34, 98	5.3726	21,4526	875,3501
		5	12,28,58,73	0.6300	29.2383	914.4561
2	2	2	49	0.5969	19.9203	662.1019
		3	51 174	2.0581e-11	21.1583	758.0191
		4	14, 35, 124	5.3726	23,3140	755,0380
		5	8 27 50 175	3.0164	24.0518	861.1438
3	3	2	58	5.9129e-11	13.3677	2.4645e+03
		3	42, 125	1.8598	16,7948	2.6074e+03
		4	53,101,147	0.2934	20.2634	2.6530e+03
		5	50,101,147,197	0.9981	19.8674	2.6617e+03
4	4	2	49	0.5969	19.9203	662.1019
		3	51 174	2.0581e-11	21.1583	758.0191
		4	16, 52, 110	5.3596	23.7054	985.4062
		5	11 43 53 109	3.0247e-11	20.0918	1.3244e+03
5	5	2	54	6.5269e-11	16.4773	1.2349e+03
		3	41 , 117	1.8598	16.7425	1.5683e+03
		4	28, 71, 124	4.5707	22.1599	1.6853e+03
		5	27, 68, 99 , 129	1.2156	22.4701	1.7093e+03
6	6	2	54	4.2869e-11	19.1762	925.1857
		3	12, 57	0.6556	21.5325	1.0339e+03
		4	11, 50,105	5.3726	20.5982	1.0393e+03

		5	7, 25, 66,111	6.1705	25.5808	1.0562e+03
7	7	2	27	3.0164	21.8403	321.4778
		3	61 145	1.0416e-10	16.4916	2.3749e+03
		4	19 71 149	1.8288	18.7650	2.4502e+03
		5	18, 66 ,104,140	2.3992	20.3063	2.4681e+03
8	8	2	53	4.0935e-12	19.7575	590.1112
		3	13,33	11.8593	28.1062	164.0757
		4	17,34 ,73	5.3596	31.1063	172.1219
		5	10,141 ,142,142	NaN	24.5287	NaN
9	9	2	55	2.8200e-11	14.1017	2.2385e+03
		3	48,141	0.0112	14.9636	2.6239e+03
		4	53,101,147	0.2934	20.2634	2.6530e+03
		5	50,101,147,197	0.9981	19.8674	2.6617e+03
10	10	2	51	0.3598	19.3270	785.8195
		3	15,56	0.9610	22.6765	899.4156
		4	7,27,70	4.2563	27.5055	908.1982
		5	12,28,58,73	0.6300	29.2383	914.4561
11	11	2	64	5.1169e-11	19.2890	1.1571e+03
		3	17, 72	1.8332	23.5619	1.3080e+03
		4	12 54 104	2.3172	20.2236	1.3194e+03
		5	11 43 53 109	3.0247e-11	20.0918	1.3244e+03
12	12	2	54	6.5269e-11	16.4773	1.2349e+03
		3	41 117	1.8598	16.7425	1.5683e+03
		4	28 71 124	4.5707	22.1599	1.6853e+03
		5	27 68 99 129	1.2156	22.4701	1.7093e+03

13	13	2	54	4.2869e-11	19.1762	925.1857
		3	12 57	0.6556	21.5325	1.0339e+03
		4	11 50 105	5.3726	20.5982	1.0393e+03
		5	7 25 66 111	6.1705	25.5808	1.0562e+03
16	16	2	64	5.1169e-11	19.2890	1.1571e+03
		3	17 72	1.8332	23.5619	1.3080e+03
		4	12 54 104	2.3172	20.2236	1.3194e+03
		5	11 43 53 109	3.0247e-11	20.0918	1.3244e+03
17	17	2	58	5.9129e-11	13.3677	2.4645e+03
		3	55 145	0.0825	15.6988	3.0915e+03
		4	51 111 164	0.3411	17.4973	3.1199e+03
		5	48 104 167 210	0.1767	17.1514	3.1456e+03
18	18	2	72	1.0075e-10	16.5517	1.5397e+03
		3	57 137	4.6166e-11	16.9564	1.7536e+03
		4	19 68 124	0.7932	20.0811	1.9518e+03
		5	18 58 98 135	0.8726	22.1383	1.9827e+03
19	19	2	64	3.6387e-11	15.9528	-
		3	61 121	1.3645e-10	17.1968	-
		4	46 101 151	5.5188	19.4137	-
		5	40 89 96 140	2.1278	22.3932	-
20	20	2	50	1.4725e-11	18.8058	-
		3	18 59	0.1527	23.6144	-
		4	16 32 63	1.0051	26.1518	-
		5	12 35 56 146	5.9031	25.2401	-
21	21	2	26	0.2723	22.3557	-

		3	25 54	0.8782	24.9578	-
		4	15 45 48	2.6766	27.0966	-
		5	3 20 42 60	1.2679e-11	27.6611	-
22	22	2	25	1.2726	21.8403	-
		3	24 56	2.0986	23.3193	-
		4	12 41 67	7.7433	23.7475	-
		5	13 34 41 71	4.1166	29.1011	-
23	23	2	49	0.5969	19.9203	662.1019
		3	51 174	2.0581e-11	21.1583	758.0191
		4	14 55 166	3.5844	24.6439	850.8177
		5	8 27 50 175	3.0164	24.0518	861.1438
24	24	2	27	3.0164	21.8403	321.4778
		3	61 145	1.0416e-10	16.4916	2.3749e+03
		4	19 71 149	1.8288	18.7650	2.4502e+03
		5	18 66 104 140	2.3992	20.3063	2.4681e+03
25	25	2	50	7.5731e-11	17.3809	1.5794e+03
		3	36 169	8.5510e-11	15.0856	1.6232e+03
		4	59 91 160	6.9818e-11	23.3004	1.6509e+03
		5	70 82 91 137	2.7091	32.5779	1.6583e+03
26	26	2	53	0.1365	17.0835	1.7460e+03
		3	16 57	0.9094	18.1318	1.8114e+03
		4	13 63 97	3.5315	21.6845	1.8372e+03
		5	21 57 97 148	2.6781	20.4840	1.8489e+03
27	27	2	47	9.5516e-12	20.7306	1.2847e+03
		3	10 68	4.3892e-11	27.2682	1.3117e+03

		4	5 55 101	0.7740	23.3028	1.3169e+03
		5	3 51 94 148	0.6853	22.7095	1.3199e+03
28	28	2	122	8.7556e-12	17.8337	673.8581
		3	17 249	6.4387e-05	23.7522	870.8025
		4	14 36 131	0.0477	23.1254	946.7312
		5	16 37 48 206	4.8329	28.0436	955.4309
29	29	2	49	4.0708e-11	15.5538	1.8038e+03
		3	19 68	0.0040	19.5485	1.8697e+03
		4	20 40 81	3.9292	25.2501	1.1793e+03
		5	19 65 97 160	3.4352	23.0477	1.9066e+03
30	30	2	59	1.0825e-10	14.8478	2.2343e+03
		3	61 150	8.6419e-12	16.6348	2.4620e+03
		4	47 109 149	1.8907	17.4221	2.4831e+03
		5	32 83 105 165	0.3526	25.6988	2.4936e+03
31	31	2	58	8.6419e-11	12.0245	2.8027e+03
		3	41 131	4.5484e-11	14.4012	3.6887e+03
		4	62 88 140	0.2679	22.0411	3.7056e+03
		5	49 83 98 149	1.5577	24.9533	3.7114e+03
32	32	2	69	9.5516e-12	25.5842	1.2847e+03
		3	10 28	0.3399	18.0105	1.3116e+03
		4	3 67 88	1.2932	26.6838	1.3172e+03
		5	8 24 92 141	0.5966	18.2398	1.3197e+03
33	33	2	27	1.4305e-04	21.4448	321.4778
		3	19 65	2.5786	23.5577	531.3946
		4	19 43 63	16.2557	25.5761	554.3316

		5	12 39 40 76	1.8307e-11	26.3870	576.7434
34	34	2	64	6.6988e-04	17.1328	6.6988e-04
		3	24 72	0.7091	22.2601	1.1616e+03
		4	20 40 81	3.9292	25.2501	1.1793e+03
		5	28 40 65 91	0.3623	29.4419	1.1855e+03
35	35	2	53	4.0935e-12	19.7575	590.1112
		3	14 63	2.3909	24.2823	729.7719
		4	4 28 72	5.9100	28.6345	744.3669
		5	20 57 102 119	60.3185	24.7271	683.5374
36	36	2	26	7.9597e-12	23.9653	307.3152
		3	21 221	1.7113e-11	22.9301	357.2430
		4	25 45 240	0.0240	27.0555	376.7537
		5	24 45 59 248	1.7779	26.8599	385.3370
37	37	2	33	1.0973e-11	20.1647	277.8353
		3	35 168	2.0240e-11	22.2011	490.7581
		4	13 33 169	0.6116	24.7788	560.0800
		5	2 25 43 227	2.9231	26.6352	577.8167
38	38	2	60	1.5284	18.5146	806.9313
		3	16 58	13.5273	21.0018	921.8830
		4	16 62 133	0.7515	22.5393	997.4475
		5	16 30 57 118	2.7578	23.1806	1.0151e+03
39	39	2	61	1.2167e-11	18.8946	859.5811
		3	14 68	5.2803	22.3455	1.0152e+03
		4	14 64 131	2.4884	22.4414	1.0459e+03
		5	11 26 69 163	2.5522	25.6792	1.0544e+03

40	40	2	48	4.2073e-12	19.8386	233.8599
		3	15 57	0.4395	24.5318	378.9533
		4	16 50 227	0.4416	25.2962	399.1238
		5	16 37 57 169	0.4475	26.3612	411.5215
41	41	2	63	5.0032e-11	16.3577	1.8874e+03
		3	63 147	3.0019e-11	18.1209	2.1674e+03
		4	18 67 173	1.0706	20.4285	2.3080e+03
		5	19 64 104 137	6.1326	20.8161	2.3435e+03
42	42	2	16	0.0071	23.0774	169.5872
		3	15 54	0.6135	25.8853	275.2516
		4	12 47 66	1.8839	25.3479	292.9202
		5	18 23 48 66	1.0779	30.0206	300.7711
43	43	2	79	1.7435e-04	13.2856	3.2774e+03
		3	63 145	4.5484e-11	16.6514	4.2363e+03
		4	64 133 230	3.6230e-04	16.5913	4.2806e+03
		5	20 69 147 218	0.8892	18.7629	4.3230e+03
44	44	2	44	1.5237e-11	15.5691	1.9365e+03
		3	64 135	4.8895e-11	19.7679	2.0260e+03
		4	64 96 135	1.3340	22.3062	2.0814e+03
		5	9 58 96 151	5.1602	21.3855	2.1105e+03
45	45	2	29	3.0819	23.7094	3.0819
		3	18 51	1.2652	25.9417	336.9484
		4	4 36 62	1.8994	24.5624	340.6207
		5	14 30 49 110	3.3229	28.5470	359.8904
45	45	2	56	5.0032e-12	17.3878	2.2238e+03

		3	57 174	6.4541e-04	18.3581	2.3154e+03
		4	59 108 160	9.1422e-11	20.2056	2.3501e+03
		5	8 60 110 163	3.0250	20.7187	2.3717e+03

Hasil Multithresholding Improved Harmony Search pada Data Uji

No	Image	k	Thresholds	STD	PSNR	MEAN
1	1	2	208	2.5471e-11	19.2858	785.8195
		3	12 87	0.1757	21.5325	1.0339e+03
		4	12 34 98	5.3726	21,4526	875,3501
		5	12 28 58 73	0.6300	29.2383	914.4561
2	2	2	65	3.1725e-11	22.8236	662.1019
		3	12 49	2.0581e-11	21.1583	758.0191
		4	12 75 102	1.1419	29.1222	-
		5	8 27 50 175	3.0164	24.0518	861.1438
3	3	2	58	5.9129e-11	13.3677	2.4645e+03
		3	47 141	7.9597e-11	17.9496	2.6074e+03
		4	53 111 147	0.2934	21.2634	2.6530e+03
		5	51 101 146 197	0.9981	19.9674	2.6617e+03
4	4	2	49	0.5969	19.9203	662.1019
		3	49 97	2.5821	20.3968	-
		4	16 52 110	5.3596	23.7054	985.4062
		5	11 43 53 109	3.0247e-11	20.0918	1.3244e+03
5	5	2	54	6.5269e-11	16.4773	1.2349e+03
		3	17 77	0.0548	24.7345	-
		4	29 72 124	4.5707	23.1599	1.6853e+03

		5	27 68 99 129	1.2157	22.4701	1.7093e+03
6	6	2	54	4.2869e-11	19.2762	925.1857
		3	21 174	0.0555	22.1788	1.0339e+03
		4	11 52 105	5.3726	20.5982	1.0393e+03
		5	7 25 66 111	6.1805	25.5808	1.0562e+03
7	7	2	28	3.0165	21.8403	321.4778
		3	14 45	2.0647	24.4140	2.3749e+03
		4	21 46 70	34.1484	27.6329	2.4502e+03
		5	18 66 105 140	2.3992	20.3063	2.4681e+03

Lampiran 3. Proses Perhitungan Ekstraksi Fitur berdasarkan Tekstur

Nilai ekstraksi fitur tekstur diperoleh dari metode Gray Level Co-occurrence Matric (GLCM). Tahap pertama dalam metode ini yakni nilai pixel pada sebuah citra retina yang sudah di perbaiki kualitasnya, terlebih dahulu di quantization untuk diproses co–occurrence matrix. Pada sebuah citra terdiri dari 256 variasi warna yang di bagi dalam 8 variasi, namun disini penulis hanya menggunakan 4 variasi untuk contoh perhitungan manual dari metode GLCM seperti ditunjukkan pada Tabel 1 sebagai berikut :

Tabel 1. Kuantisasi yang digunakan dari 8 variasi

Level	Nilai
0	0 – 31
1	32 – 63
2	64 – 95
3	96 – 127

Perhitungan nilai matrik kookurence menggunakan citra 4x4 sebagai citra awal yang ditunjukkan pada Tabel 2 dan matrik yang telah di *Quantization* berdasarkan Tabel kuantisasi diatas dapat dilihat pada Tabel 3 sebagai berikut :

Tabel 2. Matrik input citra retina awal

108	97	91	81
101	93	84	79
97	90	79	71
91	84	74	63

Tabel 3. Matrik citra retina setelah di Quantization

3	2	2	2
3	2	2	2
3	2	2	2
3	2	2	1

Area kerja dari proses perhitungan matrik yang dapat membantu dalam menentukan titik *i* sebagai *pixel* referensi dan *j* sebagai *pixel* ketetanggaan, dapat dilihat pada tabel 4. berikut ini :

Tabel 4. Framework matrik GLCM

j / i	0	1	2	3
0	#(0,0)	#(0,1)	#(0,2)	#(0,3)
1	#(1,0)	#(1,1)	#(1,2)	#(1,3)
2	#(2,0)	#(2,1)	#(2,2)	#(2,3)
3	#(3,0)	#(3,1)	#(3,2)	#(3,3)

Tahap kedua yakni menghitung dari keempat matriks GLCM untuk 4 orientasi berbeda masing-masing untuk $\theta = 0°$, $\theta = 45°$, $\theta = 90°$, dan $\theta = 135°$ berdasarkan berdasarkan Gambar 2.7 hingga Gambar 2.9, dengan radius $\delta = 1$ (d adalah jarak antara dua piksel). Dengan demikian, dapat diperoleh $GLCM_{0,}$ $GLCM_{45,}$ $GLCM_{90,}$ $GLCM_{135.}$ Hasil perhitungan untuk orientasi keempat sudut ditunjukkan pada Tabel 5 hingga Tabel 8 dibawah ini :

Tabel 5. Hasil matrik orientasi 0°

0	0	0	0
0	0	0	0
0	0	0.5	0.14
0	0	0.28	0.07

Tabel 6. Hasil matrik orientasi 45°

0	0	0	0
0	0	0	0
0	0	0.66	0.16
0	0	0.16	0

Tabel 7. Hasil matrik orientasi 90°

0	0	0	0
0	0	0	0
0	0	0.66	0.04
0	0	0.04	0.25

Tabel 8. Hasil matrik orientasi 135°

0	0	0	0
0	0	0.05	0
0	0.05	0.55	0.16
0	0	0.16	0

Dari setiap GLCM matriks yang dikonstruksi pada tahap kedua, kemudian dilakukan ekstraksi ciri-ciri seperti kontras (C), korelasi (CO), energi (E), dan homogeneitas (H) dengan menggunakan persamaan (2.23) hingga (2.27). Berikut adalah perhitungan dari 4 ciri-ciri :

1. Kontras

   Rumus untuk menghitung kontras menggunakan persamaan (2.23). Syaratnya ketika nilai *i* dan *j* sama, sel berada pada diagonal dan $(i - j) = 0$. Nilai-nilai ini merepresentasikan pixel yang keseluruhannya mirip dengan tetangga mereka, sehingga mereka diberi bobot 0.

   Perhitungan *pixel* (2,3):

   $$= \sum_{i,j} (i - j)^2 p(i,j) = (2 - 3)^2 p(2,3) = 1 * 0{,}14 = 0{,}14$$

   Hasil perhitungan kontras berdasarkan orientasi sudut 0° ditunjukkan pada Tabel 9 berikut ini :

Tabel 9. Hasil perhitungan kontras sudut 0°

0	0	0	0
0	0	0	0
0	0	0	0.14
0	0	0.28	0

2. Korelasi

   Rumus untuk menghitung korelasi menggunakan persamaan (2.24.

   Perhitungan *pixel* (2,2):

   $$\mu i = \sum i(P_{i,j}) = 2 * 0.5 = 1$$

   $$\mu j = \sum i(p_{i,j}) = 2 * 0.5 = 1$$

$$\sigma i = \sqrt{\sum P_{i,j}(i-\mu i)^2} = \sqrt{0{,}5(2-1)^2} = 0{,}70710678$$

$$\sigma j = \sqrt{\sum P_{i,j}(j-\mu j)^2} = \sqrt{0{,}5(2-1)^2} = 0{,}70710678$$

$$= \sum_{i,j} \frac{(i-\mu i)(j-\mu j)p(i,j)}{\sigma i \sigma j}$$

$$= \frac{(2-\mu i)(2-\mu j)(2{,}2)}{\sigma i \sigma j}$$

$$= \frac{(2-1)(2-1)(0{,}5)}{0{,}70710678 * 0{,}70710678}$$

$$= 0.99991371$$

Hasil perhitungan korelasi berdasarkan orientasi sudut 0° ditunjukkan pada tabel 10 berikut ini :

Tabel 10. Hasil perhitungan korelasi sudut 0°

0	0	0	0
0	0	0	0
0	0	0.99991371	0.22534703
0	0	0.32150206	0.128467

3. Energi

Rumus untuk menghitung energi menggunakan persamaan (2.25).

Perhitungan *pixel* (2,2):

$$= \sum_{i,j} p(i,j)^2 = p(2{,}2)^2 = 0{,}5^2 = 0{,}25$$

Hasil perhitungan energi berdasarkan orientasi sudut 0° ditunjukkan pada tabel 11 berikut ini :

Tabel 11. Hasil perhitungan energi sudut 0°

0	0	0	0
0	0	0	0
0	0	0.25	0.0196
0	0	0.00784	0.0049

4. Homogenitas

Rumus untuk menghitung homogenitas menggunakan persamaan (2.27).

Perhitungan *pixel* (2,2):

$$= \sum_{i,j} \frac{p(i,j)}{1+(i-j)} = \frac{p(2,2)}{1+(2-2)} = \frac{0,5}{1+0} = 0,5$$

Hasil perhitungan homogenitas berdasarkan orientasi sudut 0° ditunjukkan pada tabel 12 berikut ini :

Tabel 12. Hasil perhitungan homogenitas sudut 0°

0	0	0	0
0	0	0	0
0	0	0.5	0.07
0	0	0.14	0.07

Tahap selanjutnya membuat fitur vektor dari properti tersebut dengan mengambil rata-rata setiap ciri dengan keempat orientasi θ yang ditunjukkan pada tabel 13 berikut ini.

Tabel 13 Hasil perhitungan rata-rata setiap fitur

Ciri-ciri	Orientasi θ				Rata-rata
	0°	45°	90°	135°	
Contrast	0.42	0.32	0.08	0.42	0.31
Correlation	1.8935	2.6350	2.7218	2.8150	2.5163
Energy	0.3529	0.4868	0.5013	0.3587	0.4249
Homogeneity	0.78	0.82	0.95	0.76	0.8275

Untuk hasil proses ekstraksi fitur tekstur citra retina dengan GLCM secara keseluruhan ditunjukkan pada Tabel 14 .

Tabel 14. Hasil proses ekstraksi fitur tekstur citra retina

No.	Nama Penyakit Retina	Ciri GLCM	Orientasi θ				
			0°	45°	90°	135°	Rata-rata
1	*Background Diabetic Retinopathy*	Contrast	0.10383	0.14665	0.1008	0.14822	0.12487
		Correlation	0.98539	0.97922	0.98581	0.979	0.98235
		Energy	0.16897	0.16043	0.1676	0.15958	0.16415
		Homogeneity	0.9553	0.94005	0.95299	0.9373	0.94641
2	*Central Retinal Vein Occlusion*	Contrast	0.25521	0.37647	0.22187	0.39582	0.31234
		Correlation	0.982	0.97328	0.98435	0.9719	0.97788
		Energy	0.20282	0.19068	0.20611	0.19001	0.1974
		Homogeneity	0.93558	0.91632	0.93805	0.91513	0.92627
3	*Proliverative Diabetic Retinopathy*	Contrast	0.17606	0.25256	0.16302	0.25524	0.21172
		Correlation	0.98336	0.97599	0.98459	0.97574	0.97992
		Energy	0.15174	0.1417	0.15388	0.14438	0.14792
		Homogeneity	0.93618	0.91634	0.9389	0.92019	0.9279
4	*Coats*	Contrast	0.16169	0.24205	0.15855	0.22774	0.19751

		Correlation	0.97583	0.96364	0.97629	0.96579	0.97039
		Energy	0.16824	0.15737	0.16723	0.15739	0.16256
		Homogeneity	0.93447	0.91293	0.9327	0.91401	0.92353
		Contrast	0.095435	0.13404	0.10101	0.14191	0.1181
5	*Retinitis*	Correlation	0.98081	0.97289	0.97965	0.97129	0.97616
		Energy	0.17232	0.16424	0.17309	0.16412	0.16844
		Homogeneity	0.95269	0.93907	0.95429	0.93573	0.94544

Lampiran 3. Listing Program Ekstraksi Tekstur

```
function varargout = Texture_Analysis(varargin)
% TEXTURE_ANALYSIS MATLAB code for Texture_Analysis.fig
%      TEXTURE_ANALYSIS, by itself, creates a new TEXTURE_ANALYSIS
or raises the existing
%      singleton*.
%
%      H = TEXTURE_ANALYSIS returns the handle to a new
TEXTURE_ANALYSIS or the handle to
%      the existing singleton*.
%
%      TEXTURE_ANALYSIS('CALLBACK',hObject,eventData,handles,...)
calls the local
%      function named CALLBACK in TEXTURE_ANALYSIS.M with the
given input arguments.
%
%      TEXTURE_ANALYSIS('Property','Value',...) creates a new
TEXTURE_ANALYSIS or raises the
%      existing singleton*.  Starting from the left, property
value pairs are
%      applied to the GUI before Texture_Analysis_OpeningFcn gets
called.  An
%      unrecognized property name or invalid value makes property
application
%      stop.  All inputs are passed to Texture_Analysis_OpeningFcn
via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows
only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help
Texture_Analysis

% Begin initialization code
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
    'gui_Singleton',  gui_Singleton, ...
    'gui_OpeningFcn', @Texture_Analysis_OpeningFcn, ...
    'gui_OutputFcn',  @Texture_Analysis_OutputFcn, ...
    'gui_LayoutFcn',  [] , ...
    'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
```

```
% End initialization code


% --- Executes just before Texture_Analysis is made visible.
function Texture_Analysis_OpeningFcn(hObject, ~, handles,
varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to Texture_Analysis (see
VARARGIN)

% Choose default command line output for Texture_Analysis
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
movegui(hObject,'center');

% UIWAIT makes Texture_Analysis wait for user response (see
UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = Texture_Analysis_OutputFcn(~, ~, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, ~, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
[filename,pathname] = uigetfile({'*.*'});

if ~isequal(filename,0)
    Info = imfinfo(fullfile(pathname,filename));
    if Info.BitDepth == 1
        msgbox('Citra masukan harus citra RGB atau Grayscale');
        return
    elseif Info.BitDepth == 8
        Img = imread(fullfile(pathname,filename));
        axes(handles.axes1)
```

```
        cla('reset')
        imshow(Img)
        title('Contrast Stretching Image')
    elseif Info.BitDepth == 8
        Img = rgb2gray(imread(fullfile(pathname,filename)));
        axes(handles.axes1)
        cla('reset')
        imshow(Img)
        title('Contrast Stretching Image')
    else
        Img =
imadjust(rgb2gray(imread(fullfile(pathname,filename))));
        axes(handles.axes1)
        cla('reset')
        imshow(Img)
        title('Contrast Stretching Image')
    end
    set(handles.pushbutton2,'Enable','on');
    set(handles.uitable1,'Data',[])
else
    return
end

handles.Img = Img;
guidata(hObject,handles);

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(~, ~, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
Img = handles.Img;
pixel_dist = str2double(get(handles.edit1,'String'));
GLCM = graycomatrix(Img,'Offset',[0 pixel_dist; -pixel_dist
pixel_dist; -pixel_dist 0; -pixel_dist -pixel_dist]);
stats =
graycoprops(GLCM,{'contrast','correlation','energy','homogeneity'}
);

Contrast = stats.Contrast;
Correlation = stats.Correlation;
Energy = stats.Energy;
Homogeneity = stats.Homogeneity;

data = get(handles.uitable1,'Data');
data{1,1} = num2str(Contrast(1));
data{1,2} = num2str(Contrast(2));
data{1,3} = num2str(Contrast(3));
data{1,4} = num2str(Contrast(4));
data{1,5} = num2str(mean(Contrast));

data{2,1} = num2str(Correlation(1));
data{2,2} = num2str(Correlation(2));
data{2,3} = num2str(Correlation(3));
```

```
data{2,4} = num2str(Correlation(4));
data{2,5} = num2str(mean(Correlation));

data{3,1} = num2str(Energy(1));
data{3,2} = num2str(Energy(2));
data{3,3} = num2str(Energy(3));
data{3,4} = num2str(Energy(4));
data{3,5} = num2str(mean(Energy));

data{4,1} = num2str(Homogeneity(1));
data{4,2} = num2str(Homogeneity(2));
data{4,3} = num2str(Homogeneity(3));
data{4,4} = num2str(Homogeneity(4));
data{4,5} = num2str(mean(Homogeneity));

set(handles.uitable1,'Data',data,'ForegroundColor',[0 0 0])

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(~, ~, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
axes(handles.axes1)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])

set(handles.pushbutton2,'Enable','off')
set(handles.edit1,'String','1')
set(handles.uitable1,'Data',[])


function edit1_Callback(~, ~, ~)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit1 as text
%        str2double(get(hObject,'String')) returns contents of
edit1 as a double


% --- Executes during object creation, after setting all
properties.
function edit1_CreateFcn(hObject, ~, ~)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
```

```
%        See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

Lampiran 4. Hasil Ekstraksi Bentuk dengan data training

No	Matrik	Eccentricity	Area	Perimeter
Choroidal Neovasscularization				
1	0,965758	0,470508	0,965758	441,378
2	0,974951	0,497139	0,974951	435,964
3	0,903559	0,741644	0,903559	474,134
4	0,753413	0,875291	0,753413	817,652
5	0,983934	0,336988	0,983934	576,524
6	0,846519	0,812903	0,846519	580,918
7	0,920848	0,762929	0,920848	698,716
8	1,250189	0,764748	1,250189	16,474
9	0,884924	0,548498	0,884924	949,94
10	0,824165	0,621617	0,824165	342,065
11	1,786741	0,765333	1,786741	7,501
12	0,874792	0,656912	0,874792	708,387
13	0,858258	0,671035	0,858258	533,992
14	1,057289	0,698655	1,057289	36,158
15	0,90365	0,668369	0,90365	928,72
16	0,938132	0,535067	0,938132	424,126
17	0,857899	0,701112	0,857899	449,046
18	0,938934	0,299021	0,938934	313,234
19	0,889958	0,77349	0,889958	698,525
20	2,551916	0,765092	2,551916	4,962
21	0,898024	0,446911	0,898024	1238,031
22	1,087696	0,817894	1,087696	20,394
23	2,385857	0,731925	2,385857	4,59
24	0,891327	0,739663	0,891327	1225,974
25	0,951217	0,246425	0,951217	1257,901
26	0,77925	0,869295	0,77925	1518,962
Branch Retinal Vein Occlussion				
27	0,911225	0,68956	0,911225	880,916
28	Inf	0	Inf	0
29	6,542259	0,866025	6,542259	1,96
30	0,61183	0,928568	0,61183	164,718
31	6,542259	0,866025	6,542259	1,96
32	Inf	0	Inf	0
33	0,970803	0,341402	0,970803	139,064
34	1,140385	0,885337	1,140385	15,92
35	0,845583	0,868583	0,845583	82,14
Histoplasmosis				
36	0,771762	0,90691	0,771762	83,578
37	0,876475	0,752331	0,876475	80,857
38	1,113532	0,58742	1,113532	39,606

39	1,120058	0,514434	1,120058	30,699
40	3,975084	0	3,975084	3,556
41	0,806724	0,856865	0,806724	441,704
42	0,881209	0,71837	0,881209	746,608
43	0,889958	0,77349	0,889958	698,525
44	0,931203	0,581242	0,931203	618,397
45	0,58291	0,860885	0,58291	829,394

Lampiran 5. Proses Perhitungan Klasifikasi dengan SOM

Tahap terakhir dalam klasifikasi penyakit pada citra retina menggunakan metode Self-Organizing Map. SOM mengambil data input dari GLCM berupa fitur vektor dan kemudian SOM mencari prototipe yang mirip dengan permintaan citra input dari retina. Jumlah sampel tekstur yang paling mirip dengan citra retina diperoleh sebagai output. Data training dan data testing yang dijadikan nilai input terdiri dari 149 citra dengan jumlah citra tiap kelas yang tidak sama, hal ini berdasarkan jumlah citra retina terbanyak dibandingkan dengan kelas penyakit yang lainnya karena hasil output dari ekstraksi fitur GLCM bukan berupa citra melainkan nilai vektor. Data input berupa file CSV (Comma Separated Value) yang berisi nilai vektor dari ekstraksi fitur tekstur meliputi empat properti yaitu kontras, korelasi, energi dan homogenitas. Selama proses berlangsung sistem menampilkan *quantization error* untuk setiap iterasinya, dan berhentu ketika telah mencapai *quantization error* terkecil. Nilai akhir bobot neuron output dijadikan acuan dalam proses pemetaan data input ke dalam *cluster*.

1. **Listing Program Python Klasifikasi**

Pada proses klasifikasi menggunakan pemrograman Python 3.6 dengan menentukan variabel-variabel yang diperlukan matrik input seperti *sigma*, *input_len*, *learning rate*, *decay function*, *neighborhood function*, *random seed* dan *max iterasi* (*epoch*) berikut listing programnya :

```
import sys
sys.path.insert(0, '../')
from minisom import MiniSom
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.gridspec import GridSpec

data = np.genfromtxt('retina_training.csv', delimiter=',', usecols=(0, 1, 2, 3))

# data normalization
data = np.apply_along_axis(lambda x: x/np.linalg.norm(x), 1, data)

# Initialization and training
som = MiniSom(7, 7, 4, sigma=3, learning_rate=0.5,
            neighborhood_function='triangle', random_seed=10)
#som.random_weights_init(data)
som.random_weights_init(data)
print("Training...")
som.train_random(data, 4000)  # random training
print("\n...ready!")
```

```
plt.figure(figsize=(7, 7))
# Plotting the response for each pattern in the retina dataset
plt.pcolor(som.distance_map().T, cmap='bone_r')  # plotting the distance map as background
#plt.colorbar()

target = np.genfromtxt('retina_training.csv', delimiter=',', usecols=(4), dtype=str)
t = np.zeros(len(target), dtype=int)
t[target == 'BackgroundDiabeticRetinopathy'] = 0
t[target == 'Coats'] = 1
t[target == 'CentralRetinalVeinOcclusion'] = 2
t[target == 'ProliverativeDiabeticRetinopathy'] = 3
t[target == 'Retinitis'] = 4

# use different colors and markers for each label
markers = ['o', 's', 'D', 'x', '+']
colors = ['C0', 'C1', 'C2', 'C3', 'C4']
for cnt, xx in enumerate(data):
    w = som.winner(xx)  # getting the winner
    # palce a marker on the winning position for the sample xx
    plt.plot(w[0]+.5, w[1]+.5, markers[t[cnt]], markerfacecolor='None',
             markeredgecolor=colors[t[cnt]], markersize=12, markeredgewidth=2)
plt.axis([0, 7, 0, 7])
plt.savefig('resulting_images/som_retina.png')
plt.show()

label = np.genfromtxt('retina_training.csv', delimiter=',', usecols=(4), dtype=str)
labels_map = som.labels_map(data, label)
label_names = np.unique(label)

plt.figure(figsize=(7, 7))
the_grid = GridSpec(7, 7)
for position in labels_map.keys():
    label_fracs = [labels_map[position][l] for l in label_names]
    plt.subplot(the_grid[6-position[1], position[0]], aspect=1)
    patches, texts = plt.pie(label_fracs)
plt.legend(patches, label_names, bbox_to_anchor=(0, 1), ncol=3)
plt.savefig('resulting_images/som_retina_pies.png')
plt.show()

som = MiniSom(7, 7, 4, sigma=1., learning_rate=0.5,
              neighborhood_function='gaussian', random_seed=10)
som.random_weights_init(data)
max_iter = 10000
q_error_pca_init = []
iter_x = []
for i in range(max_iter):
    percent = 100*(i+1)/max_iter
    rand_i = np.random.randint(len(data))
    som.update(data[rand_i], som.winner(data[rand_i]), i, max_iter)
    if (i+1) % 100 == 0:
        error = som.quantization_error(data)
        q_error_pca_init.append(error)
        iter_x.append(i)
```

```
    sys.stdout.write(f'\riteration={i:2d} status={percent:0.2f}% error={error}')

plt.plot(iter_x, q_error_pca_init)
plt.ylabel('quantization error')
plt.xlabel('iteration index')
```

2. Hasil Uji Coba Program untuk Data Training

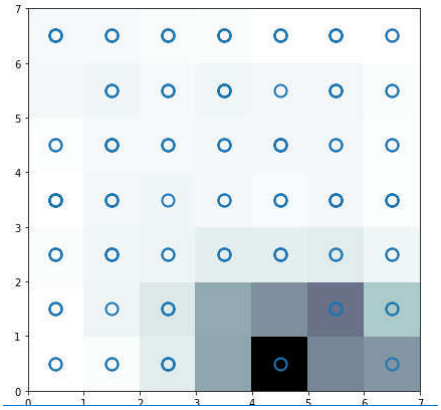
Data training yang dijadikan nilai input terdiri dari 66 citra BDR, 25 citra CRVO, 20 citra PDR, 12 citra Coats, dan 16 citra Retinitis. List parameter yang digunakan pada metode SOM untuk data training dan data testing dapat dilihat pada Tabel 1 dibawah ini.

Tabel 1. List Parameter SOM

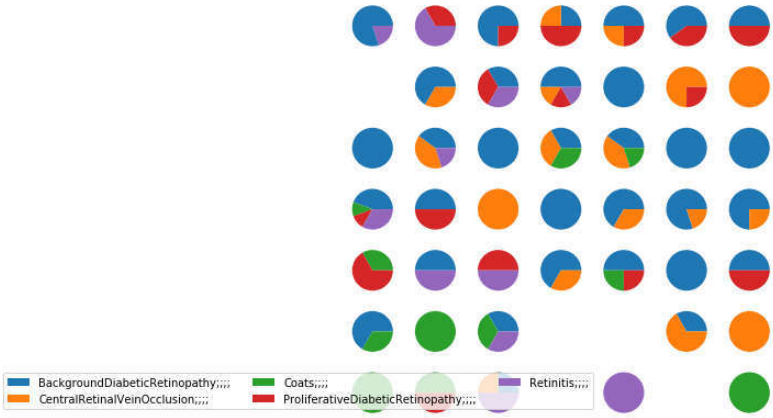
Variabel	Deskripsi	Nilai
X	Int. Dimensi x dari SOM	7
Y	Int. Dimensi y dari SOM	7
Input_len	Int. Jumlah elemen dari vektor dalam input. (Jumlah fitur dalam dataset)	4
Sigma	Float, jari-jari tetangga yang berbeda di SOM. Opsional (default = 1.0). Penyebaran fungsi lingkungan, harus memadai ke dimensi peta. (pada iterasi t memiliki sigma(t) = sigma/(1+t/T) di mana T adalah #num_iteration/2) learning_rate, laju belajar awal (pada iterasi t yang miliki learning_rate(t) = learning_rate/(1+t/T di mana T adalah # num_iteration /2)	3.0 & 1.0
Learning_rate	(α). Yang menentukan beberapa banyak bobot yang disesuaikan dalam setiap iterasi. Bilangan real positif (default = 0.3)	0.6
Decay_function	Function (default = None). Fungsi yang mengurangi learning_rate dan sigma setiap iterasi. Fungsi asli : learning_rate/(1+t/(max_iterations/2))	
Neighborhood_function	Function, opsional (default = 'gaussian'). Fungsi yang memberatkan lingkungan dari posisi di peta. Nilai yang mungkin: 'gaussian','mexican_hat','bubble','triangle'	'triangle' & 'gaussian'
Random_seed	Int, optional (default = None)	10
max_iter	Jumlah iterasi maksimum	10000

Hasil uji coba program dari data training berupa peta neuron pemenang ditunjukkan pada Gambar 1, juga Keterkaitan antar neuron satu dengan neuron yang lainnya ditunjukkan

pada Gambar 2. Hasil penurunan *quantization error* dengan meningkatkan jumlah iterasi ditunjukan pada Tabel 2 dan grafik penurunan *quantization error* secara keseluruhan ditunjukkan pada Gambar 3 berikut ini.



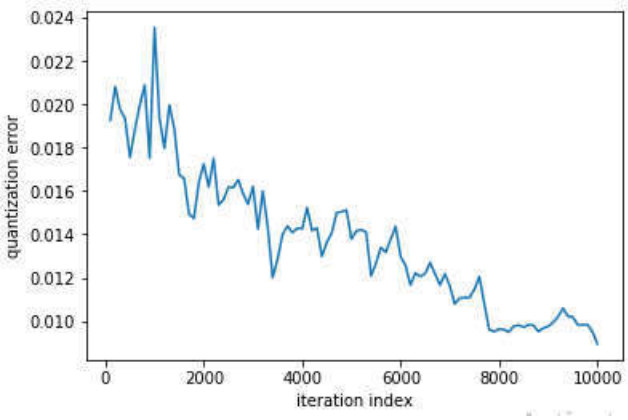
Gambar 1. Peta neuron pemenang dari data training



Gambar 2. Peta keterkaitan antar neuron

Tabel 2. Hasil quantization error nilai iterasi dari data training

Iterasi	Error
1000	0.0101
2000	0.0094
3000	0.0097
4000	0.0097
5000	0.0089
6000	0.0093
7000	0.0103
8000	0.0105
9000	0.0089
10000	0.0089

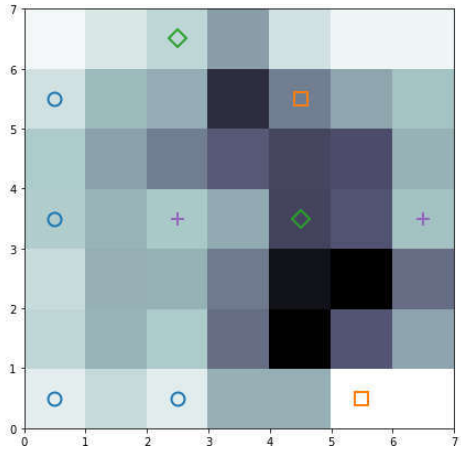


Gambar 3 Grafik hasil quantization error nilai iterasi dari data training

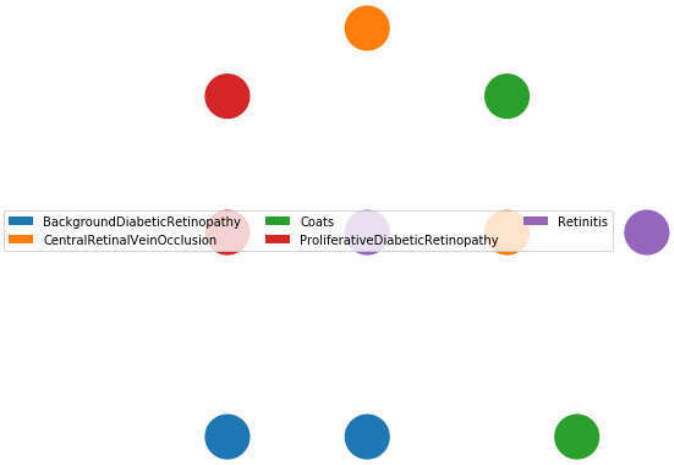
Jika dianalisis dari hasil uji coba data training dan data testing diatas, pada Gambar 4.6 dari uji coba data training tersebut dapat dilihat yang menjadi neuron pemenang pada peta lebih dominan ke penyakit *Background Diabetic Retinopathy*, hal ini karena jumlah citranya lebih dominan terdiri dari 66 citra input dibandingkan dengan keempat kelas penyakit lainnya. Dapat dilihat juga pada Gambar 4.6, hasil dari data training dengan *quantization error* terendah mencapai 0.008944726537641107. Pada tahap awal pembelajaran, peta cenderung mengatur dirinya sendiri pada vektor input. Oleh karena itu, *quantization error* rata-rata berkurang dengan cepat. Tahap ini berlangsung hingga iterasi 1000. Pada akhir fase fine-tunning (disini dari iterasi 1000 hingga 10000) *quantization error* rata-rata memiliki tren yang stabil dan semua unit peta dipesan sebaik mungkin melaui vektor input. Pada tahap ini, jaringan harus tertata dengan baik dan unit peta tersebar di seluruh vektor input.

3. Hasil Uji Coba Program untuk Data Testing

Data testing yang dijadikan nilai input terdiri 2 citra dari tiap kelas. List parameter yang digunakan pada metode SOM untuk data testing dapat dilihat pada Tabel 3 karena menggunakan parameter dan program yang sama namun dataset dan directory–nya yang harus diubah. Hasil uji coba program dari data testing berupa peta neuron pemenang yang ditunjukkan pada Gambar 4, juga Keterkaitan antar neuron satu dengan neuron lainnya ditunjukkan pada Gambar 5. Hasil penurunan *quantization error* dengan meningkatkan jumlah iterasi ditunjukkan pada Tabel 3 dan grafik penurunan *quantization error* secara keseluruhan pada Gambar 6 berikut ini.



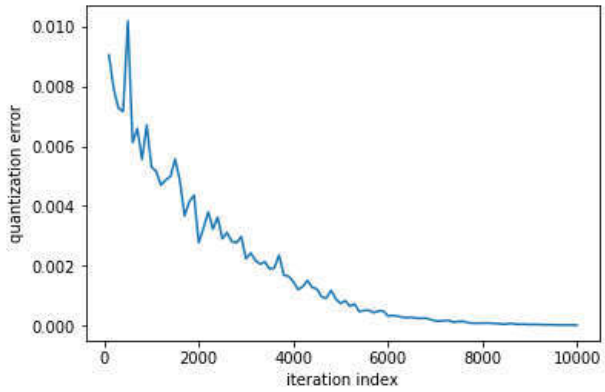
Gambar 4. Peta neuron pemenang dari data testing



Gambar 5. Peta keterkaitan antar neuron

Tabel 3. Hasil quantization error nilai iterasi dari data testing

Iterasi	Error
1000	2.8488087548470154e-05
2000	2.546896907419231e-05
3000	1.3612282043416231e-05
4000	3.9548788309462034e-05
5000	2.4084082309643136e-05
6000	2.83927605982289e-058
7000	2.5782407748221148e-05
8000	1.095579631510343e-05
9000	1.591839679034452e-05
10000	1.4732869080665522e-05



Gambar 6. Grafik quantization error nilai iterasi dari data testing

Dari Gambar 5 uji coba data testing diatas, yang menjadi neuron pemenenang pada peta mengalami konstanta terhadap lima kelas penyakit retina tersebut, hal ini karena jumlah citranya yang sama masing-masing 2 citra input. Dapat dilihat juga pada Gambar 5, hasil dari data testing dengan *quantization error* terendah mencapai 1.4732869080665522e-05 paling cocok dengan data input citra retina.

4. Analisa dan Pembahasan

Berikut adalah sampel 10 data penyakit citra retina yang dilakukan proses training data berupa perhitungan manual dengan empat variabel input yaitu kontras, korelasi, energi, dan homogenitas. Untuk data sampel, diambil dari dataset testing dilampiran dan kemudian dinormalisasikan, ditunjukkan pada Tabel 4. berikut ini.

Tabel 4. Data testing setelah di normalisasi

No.	Nama Penyakit	Kontras	Korelasi	Energi	Homogenitas
X_1	BDR	0.19	0.98	0.15	0.94
X_2	BDR	0.16	0.98	0.18	0.94
X_3	Coats	0.11	0.98	0.14	0.96
X_4	Coats	0.09	0.98	0.17	0.96
X_5	CRVO	0.14	0.98	0.18	0.94
X_6	CRVO	0.19	0.98	0.22	0.96
X_7	PDR	0.19	0.98	0.17	0.94
X_8	PDR	0.14	0.98	0.16	0.94
X_9	Retinitis	0.16	0.98	0.20	0.95
X_{10}	Retinitis	0.09	0.98	0.22	0.97

Berdasarkan input data normalisasi pada tabel diatas kemudian dilakukan training data dengan maksimum 5 iterasi. Berikut adalah hasil *clustering* 10 data dengan maksimum 5 iterasi

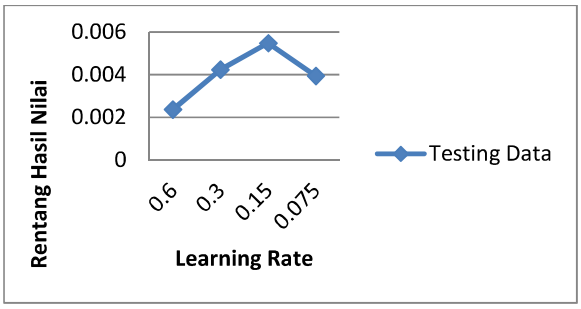
dengan *learning rate* awal 0.6 dan bobot awal (0.1 – 1). Hasil klasifikasi dari masing-masing data input dengan maksimum 5 iterasi dinyatakan index 0 untuk *kelas* 1, index 1 untuk *kelas* 2, index 2 untuk *kelas* 3 dan index 3 untuk *kelas* 4.

Tabel 5. Hasil Klasifikasi 5 iterasi

No.	Variabel Input				Klasifikasi			
	Kontras	Korelasi	Energi	Homogenitas	1	2	3	4
X_1	0.19	0.98	0.15	0.94	*			
X_2	0.16	0.98	0.18	0.94	*			
X_3	0.11	0.98	0.14	0.96	*			
X_4	0.09	0.98	0.17	0.96	*			
X_5	0.14	0.98	0.18	0.94	*			
X_6	0.19	0.98	0.22	0.96	*			
X_7	0.19	0.98	0.17	0.94	*			
X_8	0.14	0.98	0.16	0.94	*			
X_9	0.16	0.98	0.20	0.95	*			
X_{10}	0.09	0.98	0.22	0.97	*			

Dari Tabel 5 setelah melalui 5 iterasi didapatkan hasil berupa pengelompokkan dari 10 data dengan jumlah *kelas* 0 sebanyak 10 buah. Dapat dilihat juga hasil training data dengan *learning rate* 0.6 dan *learning rate* 0.0375 menghasilkan *cluster* dengan anggota semakin kecil angka indeksnya. Keakuratan hasil klasifikasi dengan semakin kecil nilai *learning rate* $\frac{10}{10} \times 100\%$ yaitu mencapai 100%.

Berikut ini pengujian data testing menggunakan pemrograman python untuk mengetahui pengaruh dari nilai *learning rate* nilai quantization error yang dapat dilihat pada Gambar 7 dibawah ini.



Gambar 7. Grafik Quantization Error nilai Learning Rate

Pada grafik Gambar 7 diatas merupakan hasil program data dengan pemrograman python menggunakan metode *Self-Organizing Map*. Dari grafik tersebut dapat diketahui bahwa

semakin rendah nilai dari *learning rate* maka nilai keakuratan semakin besar. Dengan rentang nilai 0 sampai 0.006 dan nilai max_iterasi sebesar 100, pada garis warna biru yang merupakan nilai *quantization error* dari data testing, dapat dilihat bahwa pada *learning rate* 0.6 nilainya sangatlah rendah yaitu 0.002372254, *learning rate* 0.3 nilainya meningkat yaitu 0.00424427, *learning rate* 0,15 nilainya juga meningkat yaitu 0.005489519 dan *learning rate* 0,075 nilainya menurun kembali yaitu 0.003933054. Semakin kecil nilai *learning rate* maka semakin kecil nilai quantization error. Yang mana artinya bahwa nilai keakuratan quantization error semakin kuat.

Lampiran 6. Proses Perhitungan Kalsifikasi dengan MLP

A. Hasil Pengujian Data Traning

Pelatihan dilakukan dengan perbedaan jumlah epoch pada 45 data citra retina. Adapun hasil pengujian dengan data latih ini dilakukan dengan tiga skenario yaitu sebagai berikut:

1. Pengujian dengan skenario 300 Epoch

Pengujian pada skenario ini dilakukan menggunakan pengetahuan yang dihasilkan dengan 300 epoch dalam proses *training* pada 45 citra retina. Inisialisasi angka yang digunakan pada target tersebut yaitu 1 untuk penyakit CNV, 2 untuk BRVO dan 3 untuk *Histoplasmosis*. Adapun hasil dari pengujian tersebut ditunjukan pada tabel 1.

Tabel 1 Hasil Pengujian dengan 300 Epoch

No	Data Sampel	Target Sesungguhnya	Target Hasil	Keterangan
1	Choroidal1	1	1	Target
2	Choroidal2	1	1	Target
3	Choroidal3	1	2	Bukan Target
4	Choroidal4	1	1	Target
5	Choroidal5	1	1	Target
6	Choroidal6	1	2	Bukan Target
7	Choroidal7	1	1	Target
8	Choroidal8	1	1	Target
9	Choroidal9	1	1	Target
10	Choroidal10	1	1	Target
11	Choroidal11	1	1	Target
12	Choroidal12	1	1	Target
13	Choroidal13	1	1	Target
14	Choroidal14	1	1	Target
15	Choroidal15	1	2	Bukan Target
16	Choroidal16	1	1	Target
17	Choroidal17	1	1	Target
18	Choroidal18	1	1	Target
19	Choroidal19	1	2	Bukan Target
20	Choroidal20	1	1	Target
21	Choroidal21	1	1	Target
22	Choroidal22	1	1	Target
23	Choroidal23	1	1	Target
24	Choroidal24	1	2	Bukan Target
25	Choroidal25	1	1	Target
26	Choroidal26	1	2	Bukan Target
27	BRVO1	2	2	Target
28	BRVO2	2	2	Target
29	BRVO3	2	2	Target

30	BRVO4	2	2	Target
31	BRVO5	2	2	Target
32	BRVO6	2	2	Target
33	BRVO7	2	2	Target
34	BRVO8	2	2	Target
35	BRVO9	2	2	Target
36	Histoplasmosis1	3	3	Target
37	Histoplasmosis2	3	2	Bukan Target
38	Histoplasmosis3	3	3	Target
39	Histoplasmosis4	3	3	Target
40	Histoplasmosis5	3	3	Target
41	Histoplasmosis6	3	2	Bukan Target
42	Histoplasmosis7	3	2	Bukan Target
43	Histoplasmosis8	3	2	Bukan Target
44	Histoplasmosis9	3	2	Bukan Target
45	Histoplasmosis10	3	3	Target

Tabel 2 merupakan hasil pelatihan menggunakan pengetahuan 50 epoch. Hasil akurasi dari siste perangkat lunak sebesar 76,941%. Dapat disimpulkan bahwa tingkat keberhasilan yang dihasilkan sistem dalam mengenali kelainan retina yang sesuai dengan target sesungguhnya dapat dirumuskan :

Tabel 2. Tabel Kondisi Akurasi Epoch 300

	Kondisi	
Hasil	**Positif kelainan**	**Negatif**
Positif	TP_{34}	FP_{13}
Negatif	FN_{11}	TN_{32}

Persentasi *Sensitivity* $= \left[\frac{TP}{TP+FN}\right] \text{ x } 100$

$= \left[\frac{34}{34+11}\right] \text{x} 100$

$= 75{,}555\%$

Persentasi *Specificity* $= \left[\frac{TN}{FP+TN}\right] \text{x} 100$

$= \left[\frac{32}{13+32}\right] \text{x} 100$

$= 71{,}111\%$

Persentasi Keakurasian $= \left[\frac{TP+TN}{TP+FP+TN+FN}\right] \text{x} 100$

$= \left[\frac{34+32}{34+13+32+11}\right] \text{x} 100$

= 76,941%

2. Pengujian dengan skenario 400 Epoch

Pengujian ini dilakukan dengan pengetahuan yang dihasilkan dengan 250 epoch dalam proses training pada 45 data citra kelainan retina. Adapun hasil dari pengujian tersebut ditunjukkan pada Tabel 3 di bawah ini.

Tabel 3. Hasil pengujian 400Epoch

No	Data Sampel	Target Sesungguhnya	Target Hasil	Keterangan
1	Choroidal1	1	1	Target
2	Choroidal2	1	1	Target
3	Choroidal3	1	1	Target
4	Choroidal4	1	1	Target
5	Choroidal5	1	2	Bukan Target
6	Choroidal6	1	1	Target
7	Choroidal7	1	1	Target
8	Choroidal8	1	1	Target
9	Choroidal9	1	1	Target
10	Choroidal10	1	1	Target
11	Choroidal11	1	1	Target
12	Choroidal12	1	1	Target
13	Choroidal13	1	1	Target
14	Choroidal14	1	1	Target
15	Choroidal15	1	1	Target
16	Choroidal16	1	1	Target
17	Choroidal17	1	1	Target
18	Choroidal18	1	1	Target
19	Choroidal19	1	2	Bukan Target
20	Choroidal20	1	1	Target
21	Choroidal21	1	1	Target
22	Choroidal22	1	1	Target
23	Choroidal23	1	1	Target
24	Choroidal24	1	1	Target
25	Choroidal25	1	1	Target
26	Choroidal26	1	1	Target
27	BRVO1	2	2	Target
28	BRVO2	2	2	Target
29	BRVO3	2	2	Target
30	BRVO4	2	2	Target
31	BRVO5	2	2	Target

32	BRVO6	2	2	Target
33	BRVO7	2	1	Bukan Target
34	BRVO8	2	2	Target
35	BRVO9	2	2	Target
36	Histoplasmosis1	3	3	Target
37	Histoplasmosis2	3	3	Target
38	Histoplasmosis3	3	3	Target
39	Histoplasmosis4	3	3	Target
40	Histoplasmosis5	3	2	Bukan Target
41	Histoplasmosis6	3	3	Target
42	Histoplasmosis7	3	3	Target
43	Histoplasmosis8	3	2	Bukan Target
44	Histoplasmosis9	3	3	Target
45	Histoplasmosis10	3	3	Target

Tabel 4. merupakan hasil pelatihan menggunakan pengetahuan 400 epoch. Hasil akurasi dari sistem perangkat lunak sebesar 87,777%Dapat disimpulkan bahwa tingkat keberhasilan yang dihasilkan sistem dalam mengenali kelainan retina yang sesuai dengan target sesungguhnya dapat dirumuskan :

Tabel 4. Tabel Kondisi Akurasi Epoch 400

	Kondisi	
Hasil	**Positif kelainan**	**Negatif**
Positif	TP_{40}	FP_6
Negatif	FN_5	TN_{39}

Persentasi *Sensitivity* $= \left[\frac{TP}{TP+FN}\right] x100$

$= \left[\frac{40}{40+5}\right] x100$

$= 80\%$

Persentasi *Specificity* $= \left[\frac{TN}{FP+TN}\right] x100$

$= \left[\frac{39}{6+39}\right] x100$

$= 86{,}888\%$

Persentasi Keakurasian $= \left[\frac{TP+TN}{TP+FP+TN+FN}\right] x100$

$$= \left[\frac{40+39}{40+6+39+5}\right] x100$$

$$= 87,777\%$$

3. Pengujian dengan skenario 500 Epoch

Pengujian ini dilakukan dengan pengetahuan yang dihasilkan dengan 500 epoch dalam proses training pada 45 data citra kelainan retina. Adapun hasil dari pengujian tersebut ditunjukkan pada Tabel 5 di bawah ini.

Tabel 5. Hasil Pelatihan 500 Epoch

No	Data Sampel	Target Sesungguhnya	Target Hasil	Keterangan
1	Choroidal1	1	1	Target
2	Choroidal2	1	1	Target
3	Choroidal3	1	1	Target
4	Choroidal4	1	1	Target
5	Choroidal5	1	1	Target
6	Choroidal6	1	1	Target
7	Choroidal7	1	1	Target
8	Choroidal8	1	1	Target
9	Choroidal9	1	1	Target
10	Choroidal10	1	1	Target
11	Choroidal11	1	1	Target
12	Choroidal12	1	1	Target
13	Choroidal13	1	1	Target
14	Choroidal14	1	1	Target
15	Choroidal15	1	1	Target
16	Choroidal16	1	1	Target
17	Choroidal17	1	1	Target
18	Choroidal18	1	1	Target
19	Choroidal19	1	2	Bukan Target
20	Choroidal20	1	1	Target
21	Choroidal21	1	1	Target
22	Choroidal22	1	1	Target
23	Choroidal23	1	1	Target
24	Choroidal24	1	1	Target
25	Choroidal25	1	1	Target
26	Choroidal26	1	1	Target
27	BRVO1	2	2	Target
28	BRVO2	2	2	Target
29	BRVO3	2	2	Target
30	BRVO4	2	2	Target
31	BRVO5	2	2	Target

32	BRVO6	2	2	Target
33	BRVO7	2	2	Target
34	BRVO8	2	2	Target
35	BRVO9	2	2	Target
36	Histoplasmosis1	3	3	Target
37	Histoplasmosis2	3	3	Target
38	Histoplasmosis3	3	3	Target
39	Histoplasmosis4	3	3	Target
40	Histoplasmosis5	3	3	Target
41	Histoplasmosis6	3	3	Target
42	Histoplasmosis7	3	3	Target
43	Histoplasmosis8	3	2	Bukan Target
44	Histoplasmosis9	3	3	Target
45	Histoplasmosis10	3	3	Target

Tabel 6 merupakan hasil pelatihan menggunakan pengetahuan 500 epoch. Dapat disimpulkan bahwa tingkat keberhasilan yang dihasilkan sistem dalam mengenali kelainan retina yang sesuai dengan target,sesungguhnya dapat dirumuskan seperti:

Tabel 6. Tabel Kondisi Akurasi Epoch 500

	Kondisi	
Hasil	**Positif kelainan**	**Negatif**
Positif	TP_{43}	FP_3
Negatif	FN_2	TN_{42}

Persentasi *Sensitivity* $= \left[\frac{TP}{TP+FN}\right] \times 100$

$= \left[\frac{43}{43+2}\right] \times 100$

$= 95{,}555\%$

Persentasi *Specificity* $= \left[\frac{TN}{FP+TN}\right] \times 100$

$= \left[\frac{42}{3+42}\right] \times 100$

$= 93{,}333\%\%$

Persentasi Keakurasian $= \left[\frac{TP+TN}{TP+FP+TN+FN}\right] \times 100$

$= \left[\frac{43+42}{43+3+42+2}\right] \times 100$

$= 94{,}44\%$

TN *(True Negative)* yang berarti data negatif yang terklasifikai salah oleh sistem, di sini terdapat 42 jumlah TN. TP*(True Postive)* yang berarti data positif yang terklasifikasi dengan benar oleh sistem dengan jumlah TP 43 citra. FN*(False Negative)* yaitu jumlah negatif namun terklasifikasi salah oleh sistem yang berjumlah 2 citra. Sedangkan FP*(False Positive)* yaitu jumlah data positif namun terklasifikasi salah oleh sistem yang berjumlah 3 citra. Didapat presentasi sensitivity pada epoch tertinggi ini yaitu 85,714%. Untuk *Specificity* didapat sebesar 85,714% sedangkan keakurasian didapat sebesar 85,714%

B. Hasil Pengujian Data Testing

Pengujian dilakukan dengan perbedaan jumlah epoch pada 7 data uji citra retina. Adapun hasil pengujian dengan data latih ini dilakukan dengan tiga skenario yang sama seperti proses pelatihan data, yaitu sebagai berikut:

1. Pengujian dengan skenario 300 Epoch

Pengujzian pada skenario ini dilakukan menggunakan pengetahuan yang dihasilkan dengan 300 epoch dalam proses *training* pada 7 citra retina. Inisialisasi angka yang digunakan pada target tersebut yaitu 1 untuk penyakit , 2 untuk BRVO dan 3 untuk *Histoplasmosis*. Adapun hasil dari pengujian tersebut ditunjukan pada tabel 7.

Tabel 7. Hasil Pengujian 300 Epoch

No	Data Sampel	Target Sesungguhnya	Target Hasil	Keterangan
1	Choroidal1	1	1	Target
2	Choroidal2	1	1	Target
3	Choroidal3	1	1	Target
4	BRVO1	2	2	Target
5	BRVO2	2	1	Bukan Target
6	Histoplasmosis1	3	2	Bukan Target
7	Histoplasmosis2	3	3	Target

Tabel 8 merupakan hasil pelatihan menggunakan pengetahuan 300 epoch, hasil akurasi sebesar 78,571%. Dapat disimpulkan bahwa tingkat keberhasilan yang dihasilkan sistem dalam mengenali kelainan retina yang sesuai dengan target,sesungguhnya dapat dirumuskan seperti:

Tabel 8. Kondisi Akurasi epoch 300

	Kondisi	
Hasil	**Positif kelainan**	**Negatif**

Positif	TP_5	FP_1
Negatif	FN_2	TN_6

Persentasi *Sensitivity* $= \left[\frac{TP}{TP+FN}\right] x100$

$= \left[\frac{5}{5+2}\right] x\ 100$

$= 71{,}428\%$

Persentasi *Specificity* $= \left[\frac{TN}{FP+TN}\right] x100$

$= \left[\frac{6}{1+6}\right] x\ 100$

$= 85{,}714\%$

Persentasi Keakurasian $= \left[\frac{TP+TN}{TP+FP+TN+FN}\right] x\ 100$

$= \left[\frac{5+6}{5+1+6+2}\right] x\ 100$

$= 78{,}571\%$

2. Pengujian dengan skenario 400 Epoch

Pengujian pada skenario ini dilakukan menggunakan pengetahuan yang dihasilkan dengan 400 epoch dalam proses *training* pada 7 citra retina. Adapun hasil dari pengujian tersebut ditunjukan pada tabel 9.

Tabel 9. Hasil Pengujian 400 Epoch

No	Data Sampel	Target Sesungguhnya	Target Hasil	Keterangan
1	Choroidal1	1	1	Target
2	Choroidal2	1	1	Target
3	Choroidal3	1	1	Target
4	BRVO1	2	2	Target
5	BRVO2	2	1	Bukan Target
6	Histoplasmosis1	3	3	Target
7	Histoplasmosis2	3	3	Target

Pada tabel 10 presentasi hasil keakurasian pada epoch ke 400 sebesar 85.7143%.

Tabel 10. Kondisi Akurasi epoch 400

	Kondisi	
Hasil	**Positif kelainan**	**Negatif**
Positif	TP_6	FP_1
Negatif	FN_6	TN_6

Persentasi *Sensitivity* $= \left[\frac{TP}{TP+FN}\right] \times 100$

$= \left[\frac{6}{1+6}\right] \times 100$

$= 85{,}714\%$

Persentasi *Specificity* $= \left[\frac{TN}{FP+TN}\right] \times 100$

$= \left[\frac{6}{1+6}\right] \times 100$

$= 85{,}714\%$

Persentasi Keakurasian $= \left[\frac{TP+TN}{TP+FP+TN+FN}\right] \times 100$

$= \left[\frac{6+6}{6+1+6+1}\right] \times 100$

$= 85{,}714\%$

3. Pengujian dengan skenario 500 Epoch

Pengujian pada skenario ini dilakukan menggunakan pengetahuan yang dihasilkan dengan 500 epoch dalam proses *training* pada 7 citra retina. Adapun hasil dari pengujian tersebut ditunjukan pada tabel 11.

Tabel 11. Hasil Pengujian 500 Epoch

No	Data Sampel	Target Sesungguhnya	Target Hasil	Keterangan
1	Choroidal1	1	1	Target
2	Choroidal2	1	1	Target
3	Choroidal3	1	1	Target
4	BRVO1	2	2	Target
5	BRVO2	2	1	Bukan Target
6	Histoplasmosis1	3	3	Target
7	Histoplasmosis2	3	3	Target

Pada tabel 12 presentasi hasil keakurasian pada epoch ke 500 sama seperti pengujian data dengan 400 epoch yaitu sebesar 85,714%.

Tabel 12. Kondisi Akurasi epoch 500

	Kondisi	
Hasil	**Positif kelainan**	**Negatif**
Positif	TP_6	FP_1
Negatif	FN_1	TN_6

Persentasi *Sensitivity* $= \left[\frac{TP}{TP+FN}\right] \times 100$

$= \left[\frac{6}{1+6}\right] \times 100$

$= 85{,}714\%$

Persentasi *Specificity* $= \left[\frac{TN}{FP+TN}\right] \times 100$

$= \left[\frac{6}{1+6}\right] \times 100$

$= 85{,}714\%$

Persentasi Keakurasian $= \left[\frac{TP+TN}{TP+FP+TN+FN}\right] \times 100$

$= \left[\frac{6+6}{6+1+6+1}\right] \times 100$

$= 85{,}714\%$

TN *(True Negative)* yang berarti data negatif yang terklasifikai salah oleh sistem, di sini terdapat 6 jumlah TN. TP*(True Postive)* yang berarti data positif yang terklasifikasi dengan benar oleh sistem dengan jumlah TP 6 citra. FN*(False Negative)* yaitu jumlah negatif namun terklasifikasi salah oleh sistem yang berjumlah 1 citra. Sedangkan FP*(False Positive)* yaitu jumlah data positif namun terklasifikasi salah oleh sistem yang berjumlah 1 citra. Didapat presentasi sensitivity pada epoch tertinggi ini yaitu 85,714%. Untuk *Specificity* didapat sebesar 85,714% sedangkan keakurasian didapat sebesar 85,714%

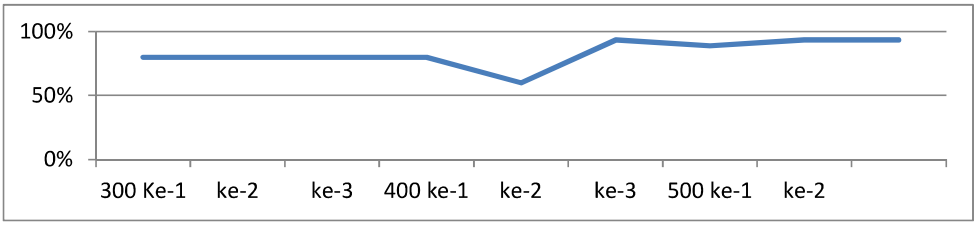
A. Analisa Hasil Data Traning

Hasil pelatihan yang telah dilakukan dengan dengan perbedaan jumlah epoch telah dijelaskan pada subbab sebelumnya, dan telah dirangkum pada Tabel 13 di bawah ini:

Tabel 13. Perbandingan Hasil Pelatihan Berdasarkan Epoch

No.	Epoch	Pelatihan ke	Jumlah Target Hasil	Presentase(%)
1.	300	1	36	80%
		2	36	80%
		3	36	80%
2.	400	1	36	80%
		2	25	60%
		3	42	93.3333%
3.	500	1	40	88.8889%
		2	42	93.3333%
		3	42	93.3333%
Average				83.2098%

Dari ke 45 data citra retina dapat dilihat hasil perbedaan epoch nya seperti Tabel 13 di atas, dari epoch terendah hingga epoch tertinggi berbeda nilai presentase akurasinya. Hasil masing-masing presentase tersebut dihitung rata-ratanya yaitu sebesar 83.2098 % yang bisa diartikan bahwa hasil pelatihan algoritma tersebut sudah baik. Untuk grafik presentase tersebut bisa dilihat pada Gambar 1 di bawah ini.

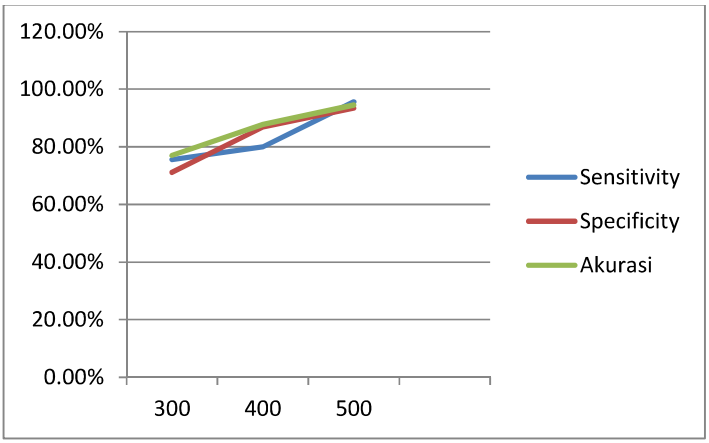


Gambar 1. Grafik Perbandingan Tingkat Akurasi Hasil Pelatihan

Tabel 14**.** Perbandingan Hasil Pengukuran Pelatihan Berdasarkan Epoch

No.	**Epoch**	**Sensitivity**	**Specificity**	**Akurasi**
1	300	75,56%	71,11%	76,94%
2	400	80%	86,89%	87,78%
3	500	95,56%	93,333%%	94,44%

Pada tabel 14. dilihat hasil pengukuran nilai *sensitivity, specificity* dan akurasinya berkaitan dengan nilai epochnya. Semakin tinggi nilai epochnya,maka nilai pengukuran *sensitivity, specificity* dan akurasinya juga semakin tinggi, karena hasil dari pengukuran tersebut diperoleh dari nilai keluaran epoch tersebut. Hasil pengukuran pelatihan tersebut bisa dibuktikan dengan grafik pada Gambar 2 di bawah ini.



Gambar 2. Perbandingan Tingkat Pengukuran Hasil Pelatihan

B. Analisa Hasil Data Testing

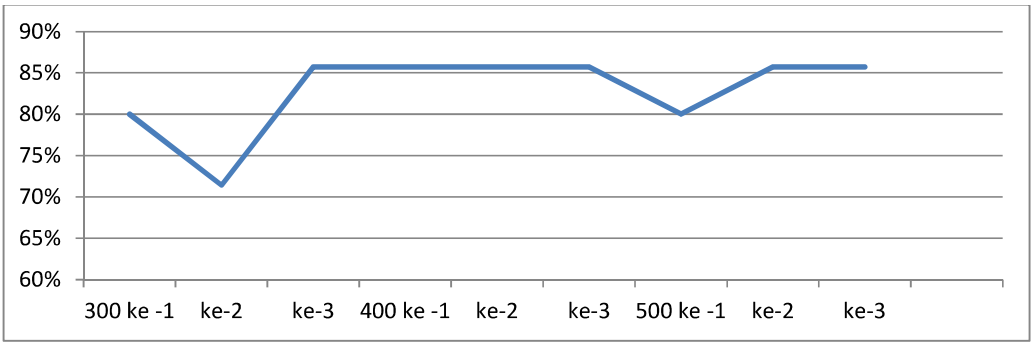
Hasil pengujian yang telah dilakukan dengan dengan perbedaan jumlah epoch telah dijelaskan pada subbab sebelumnya, dan telah dirangkum pada Tabel 15 di bawah ini:

Tabel 15. Perbandingan Hasil Pengujian

No.	Epoch	Pengujian ke	Jumlah Target Hasil	Presentase(%)
1.	300	1	6	80%
		2	5	71.4286%
		3	6	85.7143%
2.	400	1	6	85.7143%
		2	6	85.7143%
		3	6	85.7143%
3.	500	1	6	80%
		2	6	85.7143%
		3	6	85.7143%
Average				82.8572%

Dari ke 7 data citra retina dapat dilihat hasil perbedaan epoch nya seperti Tabel 15 di atas, dari epoch terendah hingga epoch tertinggi berbeda nilai presentase akurasinya. Hasil masing-masing presentase tersebut dihitung rata-ratanya yaitu sebesar 82.8572% yang bisa diartikan bahwa hasil pelatihan algoritma MLP tersebut sudah baik. Dengan pengujian yang telah dilakukan tersebut dapat diimpulkan bahwa data yang digunakan pada proses pelatihan sangat menentukan keakuratan dalam dalam mengenali data yang diuji.

Adapun grafik terhadap Tabel 16 Perbandingan Hasil Pelatihan di atas terdapat pada gambar 3. di bawah ini.

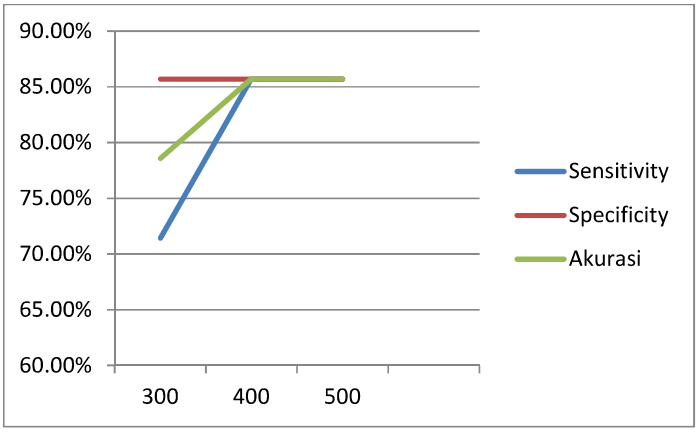


Gambar 3. Perbandingan Tingkat Akurasi Hasil Pengujian

Tabel 16. Perbandingan Hasil Pengukuran Pengujian Berdasarkan Epoch

	Epoch	Sensitivity	Specificity	Akurasi
1	300	71,43%	85,71%	78,57%
2	400	85,71%	85,71%	85,71%
3	500	85,71%	85,71%	85,71%

Pada tabel 16. dilihat hasil pengukuran nilai *sensitivity, specificity* dan akurasinya berkaitan dengan nilai epochnya. Semakin tinggi nilai epochnya,maka nilai pengukuran *sensitivity, specificity* dan akurasinya juga semakin tinggi, karena hasil dari pengukuran tersebut diperoleh dari nilai keluaran epoch tersebut. Hasil pengukuran pengujian tersebut bisa dibuktikan dengan grafik pada Gambar 4 di bawah ini.

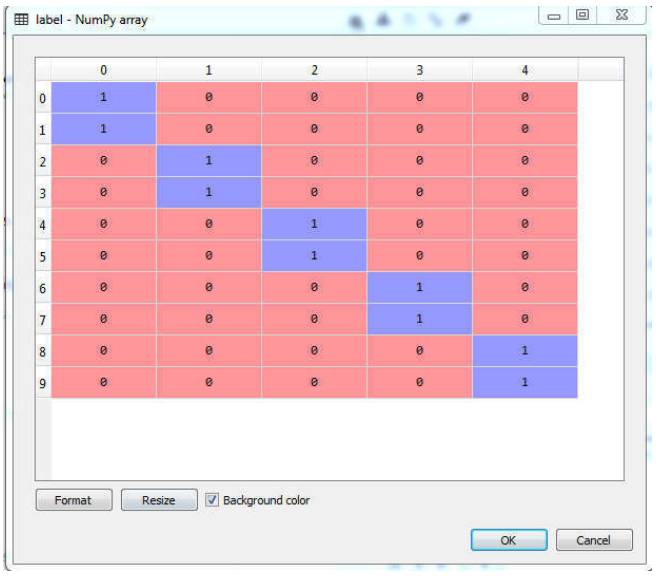


Gambar 4. Perbandingan Tingkat Pengukuran Hasil Pengujian

Lampiran 7. Proses Perhitungan Klasifikasi dengan BP

Setelah melakukan ekstraksi fitur maka tahapan selanjutnya yaitu klasifikasi dengan menggunkan metode *Back Propagation*. Metode *Back Propagation* merupakan metode dengan menggunakan proses data latih dan data testing. Pada klasifikasi penyakit retina menggunakan *Back Propagation* data latih atau yang disebut dengan data training dan sedangkan data uji digunakan untuk mengukur tingkat keakurasian suatu metode dalam mengenali kelas penyakitnya masing- masing. Pada penelitian ini dilakukan dua kali percobaan yaitu dengan menggunakan epoch yaitu 100 epoch dan 1000 epoch. Dan pada penelitian ini menggunakan dua data yaitu data pertama adalah data training 1 yang berjumlah 10 data citra training dan data testing 1 yang berjumlah 5 data citra testing. Dan yang kedua yaitu data training 2 yang berjumlah 10 data citra training dan data testing 1 yang berjumlah 5 data citra testing.

Adapun Tahap awal klasifikasi pada penelitian ini yaitu memasukan hasil dari ekstraksi fitur ke dalam inputan BP. Setelah memasukan hasil ekstraksi fitur kedalam inputan BP maka langkah selanjutnya yaitu pemberian label atau predicted label pada setiap kelas penyakit, pada penelitian pemberian label target pada setiap kelas penyakit dengan penggunaan index label yaitu pada label [1. 0. 0. 0. 0.] yaitu kelas penyakit BDR, lalu [0. 1. 0. 0. 0.] masuk pada kelas CNV, [0. 0. 1. 0. 0.] masuk pada kelas Coats, [0. 0. 0. 1. 0.] masuk pada kelas normal dan [0. 0. 0. 0. 1.] masuk pada kelas PDR. Seperti pada gambar 1 dibawah ini



	0	1	2	3	4
0	1	0	0	0	0
1	1	0	0	0	0
2	0	1	0	0	0
3	0	1	0	0	0
4	0	0	1	0	0
5	0	0	1	0	0
6	0	0	0	1	0
7	0	0	0	1	0
8	0	0	0	0	1
9	0	0	0	0	1

Gambar 1. Pemberian Lebel Target Pada Setiap Penyakit

Setelah melakukan pemberian lebel maka langkah selanjutnya yaitu proses pelatihan *Back Propagation*. Data latih pada penelitian ini di dapatkan dari hasil ekstraksi fitur warna. Adapun pseudocode data latih pada penelitan ini seperti pada Gambar 2 dibawah ini.

```
#============================== tahapa training =====================

feature = []
label = []
for idx, img in enumerate(image_data):
   label.append(img[1][0])
   x = img[0][0]
   feature.append(np.reshape(x, (x.shape[0] * x.shape[1] * 3)))
   if idx == 9:
     break
feature = np.array(feature)
label = np.array(label)
print (label)
```

Gambar 2. Pseudocode tahap training

Pada proses training terdapat 5 kelas penyakit retina yaitu kelas penyakit BDR, lalu kelas penyakit CNV lalu kelas penyakit coats, kelas retina normal, dan yang terakhir yaitu kelas PDR. Pada tiap kelas hanya terdapat 2 citra retina hal ini dikarenakan jumlah ram pada leptop yang digunakan tidak cukup untuk digunakan pada proses kasifikasi penyakit retina sehingga citra retina yang hanya bias di proses dalam satu kali percobaan yaitu maksimal 2 citra di tiap masing-masing kelas data training dan 1 citra di tiap kelas testing. Sehingga total dari citra yang digunakan dalam satu kali proses klasifikasi yaitu 15 gambar citra retina. Sehingga pada penelitian ini dilakukan pengulangan pengujian terhadap data testing dan training yang berbeda-beda. Adapun pseudocode yang digunakan dalam proses testing seperti pada Gambar 3 dibawah ini.

```
feature_testing = []
label_testing = []
for i, img1 in enumerate(image_testing):
   label_testing.append(img1[1][0])
   x = img1[0][0]
   feature_testing.append(np.reshape(x, (x.shape[0] * x.shape[1] * 3)))
if i == 9:
                                                    break
feature_testing = np.array(feature_testing)
label_testing = np.array(label_testing)

pca = PCA()
feature_transform = pca.fit_transform(feature_training)
feature_transform_testing = pca.fit_transform(feature_testing)
```

Gambar 3 Pseudocode Tahap Testing

Dapat dilihat pada pseudecode memiliki tahapan yang sama pada tahapan data testing, hal ini dikarenakan mendapatkan penyataraan antara proses training dan testing sehingga data yang dihasilkan akan lebih akurat. Adapun hasil yang dapat dilihat dari hasil uji klasifikasi penyakit menggunakan BP.

B. Hasil Pengujian Klasifikasi Penyakit Citra Retina Menggunakan BP

Pada hasil pengujian ini dilakukan tahap klasifikasi penyakit citra retina menggunakan BP dengan 2 pengujian yaitu pengujian pertama yang menggunakan data uji 1 yang terdiri dari 10 data training dengan 5 kelas penyakit dan dan 5 data testing yang menggunakan 100 epoch. Lalu pengujian kedua yaitu menggunakan data yang sama dengan 10 data training dengan 5 kelas penyakit dan 5 data testing dengan 5 jenis penyakit retina yang menggunakan 500 epoch. Sedangkan yang pengujian ketiga Lalu pengujian kedua yaitu menggunakan data yang sama dengan 10 data training dengan 5 kelas penyakit dan 5 data testing dengan 5 jenis penyakit retina yang menggunakan 1000 epoch. Lima kelas penyakit tersebut yaitu kelas penyakit BDR, CNV, Coats, normal, dan PDR.

1. Hasil Uji 1 Dengan Menggunakan 100 Epoch

Adapun hasil uji klasifikasi dengan menggunakan 10 data training dan 5 data testing dengan menggunakan 100 epoch, dan dengan menggunakan fungsi aktivasi sigmoid menghasilkan 2

citra retina yang benar dalam mengklasifikasi penyakit dan 3 yang salah dalam mengklasifikasikan penyakit dapat dilihat pada gambar 4 dibawah ini.

```
Epoch 100/100
10/10 [==============================] - 0s 2ms/step - loss: 0.0333 - acc: 1.0000
==================Hasil Prediksi=======================
Nilai perdiksi :[0.43004197 0.03445492 0.00343839 0.50357294 0.02849178] Kelas Sebenarnya :[1. 0. 0. 0. 0.]
Hasil Prediksi Clasifikasi adalah:normal, Validasi :SALAH
Nilai perdiksi :[0.05297687 0.8119989  0.07018293 0.05964485 0.00519638] Kelas Sebenarnya :[0. 1. 0. 0. 0.]
Hasil Prediksi Clasifikasi adalah:CNV, Validasi :BENAR
Nilai perdiksi :[0.07234222 0.00660214 0.75002164 0.02145392 0.14958009] Kelas Sebenarnya :[0. 0. 1. 0. 0.]
Hasil Prediksi Clasifikasi adalah:Coats, Validasi :BENAR
Nilai perdiksi :[0.00335647 0.02068433 0.01422087 0.00182159 0.9599168 ] Kelas Sebenarnya :[0. 0. 0. 1. 0.]
Hasil Prediksi Clasifikasi adalah:PDR, Validasi :SALAH
Nilai perdiksi :[0.05928168 0.25208846 0.17837375 0.4936391  0.01661695] Kelas Sebenarnya :[0. 0. 0. 0. 1.]
Hasil Prediksi Clasifikasi adalah:normal, Validasi :SALAH
Nilai perdiksi :[0.43005085 0.03445372 0.00343846 0.50356543 0.02849167] Kelas Sebenarnya :[1. 0. 0. 0. 0.]
Hasil Prediksi Clasifikasi adalah:normal, Validasi :SALAH
Nilai perdiksi :[0.05297531 0.81200725 0.07017998 0.05964123 0.00519627] Kelas Sebenarnya :[0. 1. 0. 0. 0.]
Hasil Prediksi Clasifikasi adalah:CNV, Validasi :BENAR
Nilai perdiksi :[0.07234126 0.00660238 0.7500341  0.02145318 0.1495691 ] Kelas Sebenarnya :[0. 0. 1. 0. 0.]
Hasil Prediksi Clasifikasi adalah:Coats, Validasi :BENAR
Nilai perdiksi :[0.00335648 0.02068478 0.01422079 0.00182164 0.95991635] Kelas Sebenarnya :[0. 0. 0. 1. 0.]
Hasil Prediksi Clasifikasi adalah:PDR, Validasi :SALAH
Nilai perdiksi :[0.05928168 0.25208846 0.17837375 0.4936391  0.01661694] Kelas Sebenarnya :[0. 0. 0. 0. 1.]
Hasil Prediksi Clasifikasi adalah:normal, Validasi :SALAH
 Total Waktu Proses 14.815847396850586 Detik
```

Gambar 4 Hasil Pengujian 1 Klasifikasi Menggunakan 100 Epoch

Pada Gambar 4 terdapat 5 hasil klasifikasi citra menggunakan BP dengan 100 epoch yang memiliki 5 target real yaitu [1.0.0.0.0] merupakan target real dari BDR, lalu [0.1.0.0.0] merupakan target real dari CNV, [0.0.1.0.0] merupakan target real dari Coats, [0.0.0.1.0] merupakan target real dari Normal dan yang terakhir [0.0.0.0.1] merupakan target real dari PDR. Pada pengujian 1 dengan menggunakan 100 Epoch mendapatkan hasil 2 hasil klasifikasi yang benar atau sama dengan target real atau kelas sebenarnya yaitu 1 citra CNV dan 1 citra Coats. Setelah melakukan pengujian 1 dan didapatkan hasil klasifikasi maka untuk mempermudah menganalisa suatu pengujian tersebut maka dibuatlah tabel 4.46 untuk mempermudah dalam membaca hasil klasifikasi sebenarnya dan hasil prediksi kelas dengan pengujian 100 epoch.

Tabel 1. Hasil Pengujian Kelas Pengujian 100 Epoch

KELAS SEBENARNYA	LABEL PENYAKIT	HASIL PREDIKSI KELAS
BDR	[1.0.0.0.0]	Salah (Normal)
CNV	[0.1.0.0.0]	Benar (CNV)
COATS	[0.0.1.0.0]	Benar (COATS)
NORMAL	[0.0.0.1.0]	Salah (PDR)
PDR	[0.0.0.0.1]	Salah (Normal)

Pada tabel 1 dapat dilihat terdapat dua hasil klasifikasi yang benar yaitu pada penyakit CNV yang hasil kelas sebenarnya sama dengan hasil prediksi yaitu sama-sama menunjukan penyakit CNV dengan lebel [0.1.0.0.0]. dan pada penyakit Coats juga mendapatkan hasil pengklasifikasian yang benar yaitu hasil kelas sebanarnya sama dengan hasil prediksi yang sama-sama menunjukan penyakit Coats dengan label [0.0.1.0.0]. Setelah mendapatkan tabel 1 Tahap selanjutnya yaitu membuat tabel 2 dan tabel 3 untuk mempermudah mengetahui nilai nilai True Positive (TP), False Negative (FN), False Positive (FV), dan True Negative (TN). Serta mempermudah dalam mendapatkan nilai spesifisitas atau yang disebut tingkat negatif sejati atau yang disebut True Positive Rate (TPR), alse Positive Rate (FPR), False negative rate (FNR), True Negative Rate (TNR), Positive Predictive Value (PPV), False Discovery Rate (FDR), Negative Predictive Value (NPV), dan Accuracy (ACC). Tabel 2 Tabel Hasil Prediksi Kelas Berdasarkan Kelas Sebenarnya 100 epoch

Tabel 2. Hasil Prediksi Kelas Berdasarkan Kelas Sebenarnya 100 epoch

		KELAS SEBENARNYA				
		BDR	CNV	COATS	NORMAL	PDR
HASIL PREDIKSI KELAS	BDR				1	
	CNV		1			
	COATS			1		
	NORMAL					1
	PDR				1	

Dari tabel 2 dapat dilihat terdapat hasil dari prediksi kelas dan hasil dari kelas sebenarnya ditandai dengan angka 1 (satu) untuk mempermudah dalam mengenali penyakit yang telah dilakukan prediksi kelasi sesuai dengan kelas sebenarnya sehingga mempermudah dalam menentukan nilai. Setelah membuat tabel 2 maka langkah selanjutnya dengan membuat tabel 3 untuk mendapatkan nilai True Positive (TP), False Negative (FN), False Positive (FV), dan True Negative (TN).

Tabel 3. Hasil Pengujian 100 Epoch

		KELAS SEBENARNYA	
Hasil Prediksi Kelas		PENYAKIT	Bukan PENYAKIT
	PENYAKIT	2 (TP)	2 (FP)
	Bukan PENYAKIT	1 (FN)	0 (TN)

Pada tabel 3 didapatkan nilai TP, FN, FP, dan TN yang digunakan untuk menghitung nilai True Positive Rate (TPR), alse Positive Rate (FPR), False negative rate (FNR), True Negative Rate (TNR), Positive Predictive Value (PPV), FFalse Discovery Rate (FDR), Negative Predictive Value (NPV), dan Accuracy (ACC).

Sensitifitas atau True Positive Rate (TPR) = $\frac{TP}{TP+FN} = \frac{2}{2+1} = 0.67$

False Positive Rate (FPR) = $\frac{FP}{FP+TN} = \frac{2}{2+0} = 1$

False Negative Rate (FNR) = $\frac{FN}{FN+TP} = \frac{1}{1+2} = 0.3$

Spesifikasi atau True Negative Rate (TNR) = $\frac{TN}{TN+FP} = \frac{0}{0+2} = 0$

Presisi atau Positive Predictive Value (PPV) = $\frac{TP}{TP+FP} = \frac{2}{2+2} = 1$

False Discovery Rate (FDR) = $\frac{FP}{FP+TP} = \frac{2}{2+2} = 1$

Nilai Prediksi Negatif atau Negative Predictive Value (NPV) = $\frac{TN}{TN+FN} = \frac{0}{0+1} = 0$

Akurasi atau Accuracy (ACC) = $\frac{TP+TN}{TP+TN+FP+FN} = \frac{2+0}{2+0+2+1} = \frac{2}{5} = 0.4$

2. Hasil Uji 2 Dengan Menggunakan 500 Epoch

Adapun hasil uji klasifikasi dengan menggunakan 10 data training dan 5 data testing dengan menggunakan 500 epoch, dan dengan menggunakan fungsi aktivasi sigmoid menghasilkan 3 citra retina yang benar dalam mengklasifikasi penyakit dan 2 yang salah dalam mengklasifikasikan penyakit dapat dilihat pada gambar 5 dibawah ini.

```
Epoch 500/500
10/10 [==============================] - 0s 2ms/step - loss: 0.0019 - acc: 1.0000
==================Hasil Prediksi========================
Nilai perdiksi :[7.3872507e-01 2.2497312e-03 5.3386873e-04 2.3656139e-01 2.1929925e-02] Kelas Sebenarnya :
[1. 0. 0. 0. 0.] Hasil Prediksi Clasifikasi adalah:BDR, Validasi :BENAR
Nilai perdiksi :[7.4530714e-03 9.8437381e-01 6.3761906e-03 1.3265493e-03 4.7047707e-04] Kelas Sebenarnya :
[0. 1. 0. 0. 0.] Hasil Prediksi Clasifikasi adalah:CNV, Validasi :BENAR
Nilai perdiksi :[7.4764322e-03 3.9182504e-04 9.4271386e-01 4.6828729e-03 4.4734970e-02] Kelas Sebenarnya :
[0. 0. 1. 0. 0.] Hasil Prediksi Clasifikasi adalah:Coats, Validasi :BENAR
Nilai perdiksi :[3.8865241e-04 1.0081084e-03 1.8000496e-03 2.8275821e-04 9.9652040e-01] Kelas Sebenarnya :
[0. 0. 0. 1. 0.] Hasil Prediksi Clasifikasi adalah:PDR, Validasi :SALAH
Nilai perdiksi :[0.06150616 0.08346253 0.54719245 0.27668008 0.03115873] Kelas Sebenarnya :[0. 0. 0. 0. 1.]
Hasil Prediksi Clasifikasi adalah:Coats, Validasi :SALAH
Nilai perdiksi :[7.3874158e-01 2.2495745e-03 5.3390232e-04 2.3654471e-01 2.1930143e-02] Kelas Sebenarnya :
[1. 0. 0. 0. 0.] Hasil Prediksi Clasifikasi adalah:BDR, Validasi :BENAR
Nilai perdiksi :[7.4528009e-03 9.8437423e-01 6.3759866e-03 1.3264967e-03 4.7046941e-04] Kelas Sebenarnya :
[0. 1. 0. 0. 0.] Hasil Prediksi Clasifikasi adalah:CNV, Validasi :BENAR
Nilai perdiksi :[7.4761538e-03 3.9183674e-04 9.4272590e-01 4.6829483e-03 4.4723235e-02] Kelas Sebenarnya :
[0. 0. 1. 0. 0.] Hasil Prediksi Clasifikasi adalah:Coats, Validasi :BENAR
Nilai perdiksi :[3.8865019e-04 1.0081434e-03 1.8000505e-03 2.8276737e-04 9.9652040e-01] Kelas Sebenarnya :
[0. 0. 0. 1. 0.] Hasil Prediksi Clasifikasi adalah:PDR, Validasi :SALAH
Nilai perdiksi :[0.06150617 0.08346254 0.5471925  0.27668008 0.03115874] Kelas Sebenarnya :[0. 0. 0. 0. 1.]
Hasil Prediksi Clasifikasi adalah:Coats, Validasi :SALAH
 Total Waktu Proses 30.351735591888428 Detik
```

Gambar 5 Hasil Pengujian 2 Klasifikasi Menggunakan 500 Epoch

Pada Gambar 5 terdapat 5 hasil klasifikasi citra menggunakan BP dengan 500 epoch yang memiliki 5 target real kelas sebenarnya yaitu [1.0.0.0.0] merupakan target real dari BDR, lalu [0.1.0.0.0] merupakan target real dari CNV, [0.0.1.0.0] merupakan target real dari Coats, [0.0.0.1.0] merupakan target real dari Normal dan yang terakhir [0.0.0.0.1] merupakan target real dari PDR. Pada pengujian 2 dengan menggunakan 500 Epoch mendapatkan hasil 3 hasil klasifikasi yang benar atau sama dengan target real yaitu BDR, CNV, dan Coats. Setelah melakukan pengujian 2 dan didapatkan hasil klasifikasi maka untuk mempermudah menganalisa suatu pengujian tersebut maka dibuatlah tabel 4 untuk mempermudah dalam membaca hasil klasifikasi sebenarnya dan hasil prediksi kelas dengan pengujian 500 epoch.

Tabel 4. Hasil Pengujian Kelas Pengujian 500 Epoch

KELAS SEBENARNYA	LABEL PENYAKIT	HASIL PREDIKSI KELAS
BDR	[1.0.0.0.0]	Benar (BDR)
CNV	[0.1.0.0.0]	Benar (CNV)
COATS	[0.0.1.0.0]	Benar (COATS)
NORMAL	[0.0.0.1.0]	Salah (PDR)
PDR	[0.0.0.0.1]	Salah (COATS)

Pada tabel 4 dapat dilihat terdapat tiga hasil klasifikasi yang benar yaitu pada penyakit BDR yang hasil kelas sebenarnya sama dengan hasil prediksi yaitu sama-sama menunjukan penyakit

BDR dengan label [1.0.0.0.0], lalu penyakit CNV yang hasil kelas sebenarnya sama dengan hasil prediksi yaitu sama-sama menunjukan penyakit CNV dengan lebel [0.1.0.0.0]. dan pada penyakit Coats juga mendapatkan hasil pengklasifikasian yang benar yaitu hasil kelas sebanarnya sama dengan hasil prediksi yang sama-sama menunjukan penyakit Coats dengan label [0.0.1.0.0]. Setelah mendapatkan tabel 4, tahap selanjutnya yaitu membuat tabel 5 dan tabel 6 untuk mempermudah mengetahui nilai nilai True Positive (TP), False Negative (FN), False Positive (FV), dan True Negative (TN). Serta mempermudah dalam mendapatkan nilai spesifisitas atau yang disebut tingkat negatif sejati atau yang disebut True Positive Rate (TPR), alse Positive Rate (FPR), False negative rate (FNR), True Negative Rate (TNR), Positive Predictive Value (PPV), FFalse Discovery Rate (FDR), Negative Predictive Value (NPV), dan Accuracy (ACC). Tabel 5 Tabel Hasil Prediksi Kelas Berdasarkan Kelas Sebenarnya 500 epoch.

Tabel 5. Hasil Prediksi Kelas Berdasarkan Kelas Sebenarnya 500 epoch

		KELAS SEBENARNYA				
		BDR	CNV	COATS	NORMAL	PDR
HASIL PREDIKSI KELAS	BDR	1				
	CNV		1			
	COATS			1		
	NORMAL					1
	PDR			1		

Dari tabel 5 dapat dilihat terdapat hasil dari prediksi kelas dan hasil dari kelas sebenarnya ditandai dengan angka 1 (satu) untuk mempermudah dalam mengenali penyakit yang telah dilakukan prediksi kelasi sesuai dengan kelas sebenarnya sehingga mempermudah dalam menentukan nilai. Setelah membuat tabel 5 maka langkah selanjutnya dengan membuat tabel 6 untuk mendapatkan nilai True Positive (TP), False Negative (FN), False Positive (FV), dan True Negative (TN).

Tabel 6. Hasil Pengujian 500 Epoch

		KELAS SEBENARNYA	
Hasil Prediksi Kelas		PENYAKIT	Bukan PENYAKIT
	PENYAKIT	3 (TP)	0 (FP)
	Bukan PENYAKIT	1 (FN)	1 (TN)

Pada tabel 6 didapatkan nilai TP, FN, FP, dan TN yang digunakan untuk menghitung nilai True Positive Rate (TPR), alse Positive Rate (FPR), False negative rate (FNR), True Negative Rate (TNR), Positive Predictive Value (PPV), FFalse Discovery Rate (FDR), Negative Predictive Value (NPV), dan Accuracy (ACC).

Sensitifitas atau True Positive Rate (TPR) = $\frac{TP}{TP+FN} = \frac{3}{3+1} = 0.75$

False Positive Rate (FPR) = $\frac{FP}{FP+TN} = \frac{0}{0+1} = 0$

False Negative Rate (FNR) = $\frac{FN}{FN+TP} = \frac{1}{1+3} = 0.25$

Spesifikasi atau True Negative Rate (TNR) = $\frac{TN}{TN+FP} = \frac{1}{1+0} = 1$

Presisi atau Positive Predictive Value (PPV) = $\frac{TP}{TP+FP} = \frac{3}{3+0} = 1$

False Discovery Rate (FDR) = $\frac{FP}{FP+TP} = \frac{0}{0+3} = 0$

Nilai Prediksi Negatif atau Negative Predictive Value (NPV) = $\frac{TN}{TN+FN} = \frac{1}{1+1} = 0.5$

Akurasi atau Accuracy (ACC) = $\frac{TP+TN}{TP+TN+FP+FN} = \frac{3+1}{3+1+0+1} = \frac{4}{5} = 0.8$

3. Hasil Uji 3 Dengan Menggunakan 1000 Epoch

Adapun hasil uji klasifikasi dengan menggunakan 10 data training dan 5 data testing dengan menggunakan 1000 epoch, dan dengan menggunakan fungsi aktivasi sigmoid menghasilkan 3 citra retina yang benar dalam mengklasifikasi penyakit dan 2 yang salah dalam mengklasifikasikan penyakit dapat dilihat pada gambar 6 dibawah ini.

```
Epoch 1000/1000
10/10 [==============================] - 0s 3ms/step - loss: 5.8395e-04 - acc: 1.0000
====================Hasil Prediksi========================
Nilai perdiksi :[5.0077337e-01 1.0361027e-02 9.5186813e-05 4.5616332e-01 3.2607086e-02] Kelas Sebenarnya :
[1. 0. 0. 0. 0.] Hasil Prediksi Clasifikasi adalah:BDR, Validasi :BENAR
Nilai perdiksi :[1.4614924e-03 9.8612183e-01 8.7423883e-03 3.5317407e-03 1.4254996e-04] Kelas Sebenarnya :
[0. 1. 0. 0. 0.] Hasil Prediksi Clasifikasi adalah:CNV, Validasi :BENAR
Nilai perdiksi :[2.1630120e-03 1.3253139e-04 9.9182230e-01 4.1247255e-04 5.4696421e-03] Kelas Sebenarnya :
[0. 0. 1. 0. 0.] Hasil Prediksi Clasifikasi adalah:Coats, Validasi :BENAR
Nilai perdiksi :[2.1295067e-05 3.8837511e-04 3.3925229e-04 9.8425517e-05 9.9915266e-01] Kelas Sebenarnya :
[0. 0. 0. 1. 0.] Hasil Prediksi Clasifikasi adalah:PDR, Validasi :SALAH
Nilai perdiksi :[0.16153453 0.06981307 0.7054231  0.05660725 0.0066221 ] Kelas Sebenarnya :[0. 0. 0. 0. 1.]
Hasil Prediksi Clasifikasi adalah:Coats, Validasi :SALAH
Nilai perdiksi :[5.0081366e-01 1.0360359e-02 9.5191746e-05 4.5612425e-01 3.2606531e-02] Kelas Sebenarnya :
[1. 0. 0. 0. 0.] Hasil Prediksi Clasifikasi adalah:BDR, Validasi :BENAR
Nilai perdiksi :[1.4613634e-03 9.8612273e-01 8.7421387e-03 3.5312774e-03 1.4254195e-04] Kelas Sebenarnya :
[0. 1. 0. 0. 0.] Hasil Prediksi Clasifikasi adalah:CNV, Validasi :BENAR
Nilai perdiksi :[2.1632311e-03 1.3255360e-04 9.9182206e-01 4.1252046e-04 5.4696752e-03] Kelas Sebenarnya :
[0. 0. 1. 0. 0.] Hasil Prediksi Clasifikasi adalah:Coats, Validasi :BENAR
Nilai perdiksi :[2.1295005e-05 3.8836733e-04 3.3924499e-04 9.8428427e-05 9.9915266e-01] Kelas Sebenarnya :
[0. 0. 0. 1. 0.] Hasil Prediksi Clasifikasi adalah:PDR, Validasi :SALAH
Nilai perdiksi :[0.16153453 0.06981305 0.7054231  0.05660727 0.0066221 ] Kelas Sebenarnya :[0. 0. 0. 0. 1.]
Hasil Prediksi Clasifikasi adalah:Coats, Validasi :SALAH
 Total Waktu Proses 39.912282943725586 Detik
```

Gambar 6 Hasil Pengujian 3 Klasifikasi Menggunakan 1000 Epoch

Pada Gambar 6 terdapat 5 hasil klasifikasi citra menggunakan BP dengan 1000 epoch yang memiliki 5 target real kelas sebenarnya yaitu [1.0.0.0.0] merupakan target real dari BDR, lalu [0.1.0.0.0] merupakan target real dari CNV, [0.0.1.0.0] merupakan target real dari Coats, [0.0.0.1.0] merupakan target real dari Normal dan yang terakhir [0.0.0.0.1] merupakan target real dari PDR. Pada pengujian 3 dengan menggunakan 1000 Epoch mendapatkan hasil 3 hasil klasifikasi yang benar atau sama dengan target real yaitu BDR, CNV, dan Coats. Setelah melakukan pengujian 2 dan didapatkan hasil klasifikasi maka untuk mempermudah menganalisa suatu pengujian tersebut maka dibuatlah tabel 7 untuk mempermudah dalam membaca hasil klasifikasi sebenarnya dan hasil prediksi kelas dengan pengujian 1000 epoch.

Tabel 7. Hasil Pengujian Kelas Pengujian 1000 Epoch

KELAS SEBENARNYA	LABEL PENYAKIT	HASIL PREDIKSI KELAS
BDR	[1.0.0.0.0]	Benar (BDR)
CNV	[0.1.0.0.0]	Benar (CNV)
COATS	[0.0.1.0.0]	Benar (COATS)
NORMAL	[0.0.0.1.0]	Salah (PDR)
PDR	[0.0.0.0.1]	Salah (COATS)

Pada tabel 7 dapat dilihat terdapat tiga hasil klasifikasi yang benar yaitu pada penyakit BDR yang hasil kelas sebenarnya sama dengan hasil prediksi yaitu sama-sama menunjukan penyakit BDR dengan label [1.0.0.0.0], lalu penyakit CNV yang hasil kelas sebenarnya sama dengan

hasil prediksi yaitu sama-sama menunjukan penyakit CNV dengan lebel [0.1.0.0.0]. dan pada penyakit Coats juga mendapatkan hasil pengklasifikasian yang benar yaitu hasil kelas sebanarnya sama dengan hasil prediksi yang sama-sama menunjukan penyakit Coats dengan label [0.0.1.0.0]. Setelah mendapatkan tabel 7, tahap selanjutnya yaitu membuat tabel 8 dan tabel 9 untuk mempermudah mengetahui nilai nilai True Positive (TP), False Negative (FN), False Positive (FV), dan True Negative (TN). Serta mempermudah dalam mendapatkan nilai spesifisitas atau yang disebut tingkat negatif sejati atau yang disebut True Positive Rate (TPR), alse Positive Rate (FPR), False negative rate (FNR), True Negative Rate (TNR), Positive Predictive Value (PPV), FFalse Discovery Rate (FDR), Negative Predictive Value (NPV), dan Accuracy (ACC). Tabel 8 Tabel Hasil Prediksi Kelas Berdasarkan Kelas Sebenarnya 1000 epoch

Tabel 8. Hasil Prediksi Kelas Berdasarkan Kelas Sebenarnya 1000 epoch

		KELAS SEBENARNYA				
		BDR	CNV	COATS	NORMAL	PDR
HASIL PREDIKSI KELAS	BDR	1				
	CNV		1			
	COATS			1		
	NORMAL					1
	PDR			1		

Dari tabel 8 dapat dilihat terdapat hasil dari prediksi kelas dan hasil dari kelas sebenarnya ditandai dengan angka 1 (satu) untuk mempermudah dalam mengenali penyakit yang telah dilakukan prediksi kelasi sesuai dengan kelas sebenarnya sehingga mempermudah dalam menentukan nilai. Setelah membuat tabel 8 maka langkah selanjutnya dengan membuat tabel 9 untuk mendapatkan nilai True Positive (TP), False Negative (FN), False Positive (FV), dan True Negative (TN).

Tabel 9. Hasil Pengujian 1000 Epoch

		KELAS SEBENARNYA	
Hasil Prediksi Kelas		PENYAKIT	Bukan PENYAKIT
	PENYAKIT	3 (TP)	0 (FP)
	Bukan PENYAKIT	1 (FN)	1 (TN)

Pada tabel 9 didapatkan nilai TP, FN, FP, dan TN yang digunakan untuk menghitung nilai True Positive Rate (TPR), alse Positive Rate (FPR), False negative rate (FNR), True Negative Rate (TNR), Positive Predictive Value (PPV), FFalse Discovery Rate (FDR), Negative Predictive Value (NPV), dan Accuracy (ACC).

Sensitifitas atau True Positive Rate (TPR) = $\frac{TP}{TP+FN} = \frac{3}{3+1} = 0.75$

False Positive Rate (FPR) = $\frac{FP}{FP+TN} = \frac{0}{0+1} = 0$

False Negative Rate (FNR) = $\frac{FN}{FN+TP} = \frac{1}{1+3} = 0.25$

Spesifikasi atau True Negative Rate (TNR) = $\frac{TN}{TN+FP} = \frac{1}{1+0} = 1$

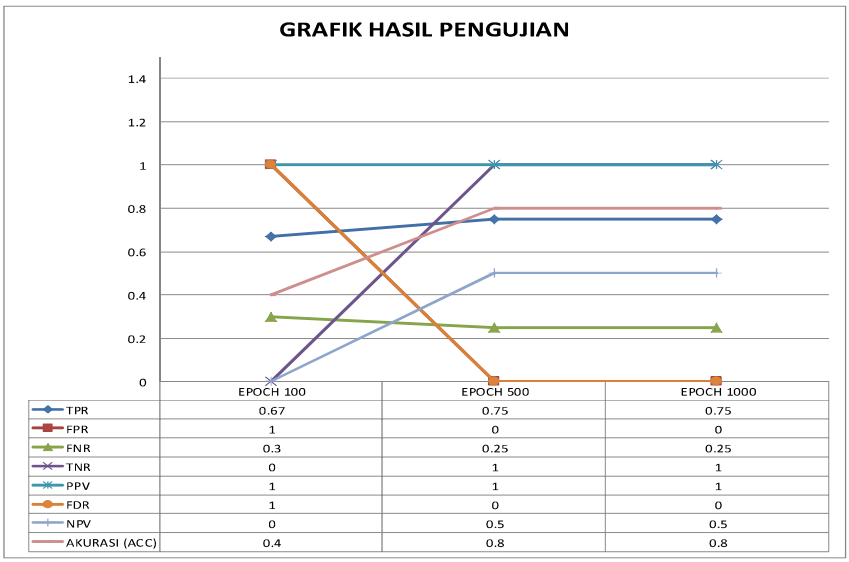
Presisi atau Positive Predictive Value (PPV) = $\frac{TP}{TP+FP} = \frac{3}{3+0} = 1$

False Discovery Rate (FDR) = $\frac{FP}{FP+TP} = \frac{0}{0+3} = 0$

Nilai Prediksi Negatif atau Negative Predictive Value (NPV) = $\frac{TN}{TN+FN} = \frac{1}{1+1} = 0.5$

Akurasi atau Accuracy (ACC) = $\frac{TP+TN}{TP+TN+FP+FN} = \frac{3+1}{3+1+0+1} = \frac{4}{5} = 0.8$

Pada pengujian yang telah dilakukan pada klasifikasi pengujian 1 dan pengujian 2 dan pengujian 3 dengan menggunakan epoch yang berbeda yaitu 100, 500, 1000 terdapat peningkatan akurasi pada pengujian 2 sedangkan pada pengujian 3 mendapatkan hasil yang sama dengan pengujian 3 hal ini dapat dilihat pada gambar 7 dibawah ini.



Gambar 7 Grafik Perbandingan nilai pengukuran kinerja

Dapat dilihat pada nilai nilai True Positive Rate (TPR) yang mengalami kenaikan dari epoch 100 ke epoch 500 yaitu dari nilai 0.67 menjadi 0.75 sedangkan pada epoch 500 ke 1000 tidak mengalami perubahan yaitu dari 0.75 menjadi 0.75. Pada False Positive Rate (FPR) terjadi penurunan dari epoch 100 hingga ke epoch 1000 yaitu dari nilai 1 menjadi 0. Pada nilai False negative rate (FNR) juga terjadi penurunan dari 0.3 menjadi 0.25. Sedangkan nilai True Negative Rate (TNR) terdapat peningkatan yaitu dari nlai 0 menjadi 1. Pada nilai Positive Predictive Value (PPV) tidak terjadi perubahan yaitu tetap pada nilai 1 pada epoch 100 hingga epoch 1000. False Discovery Rate (FDR) mengalami penurunan nilai yaitu dari 1 menjadi 0 pada epoch 100 hingga epoch 1000. Negative Predictive Value (NPV) mengalami kenaikan yaitu dari 0 menjadi 0.5 dan terakhir yaitu Accuracy (ACC) yang mengalami peningkatan akurasi pada epoch 100 ke epoch 500 yaitu dari nilai 0.4 menjadi 0.8 sedangkan pada epoch 500 ke epoch 1000 tidak terjadi perubahan yaitu tetap pada nilai 0.8..

C. Waktu Proses Klasifikasi

Pada penelitian ini pada pengujian 1 dengan penggunaan epoch 100 memiliki waktu proses 14,8 detik dalam melakukan proses klasifikasi menggunakan BP dengan 100 epoch dapat dilihat pada gambar 8 dibawah ini.

```
Total Waktu Proses 14.815847396850586 Detik
```

Gambar 8 Total Waktu Proses 100 epoch

Pada pengujian 2 dengan penggunaan epoch 500 memiliki wkatu proses 30,35 detik dalam melakukan proses klasifikasi menggunakan BP dengan 500 epoch dapat dilihat pada gambar 9 dibawah ini.

```
Total Waktu Proses 30.351735591888428 Detik
```

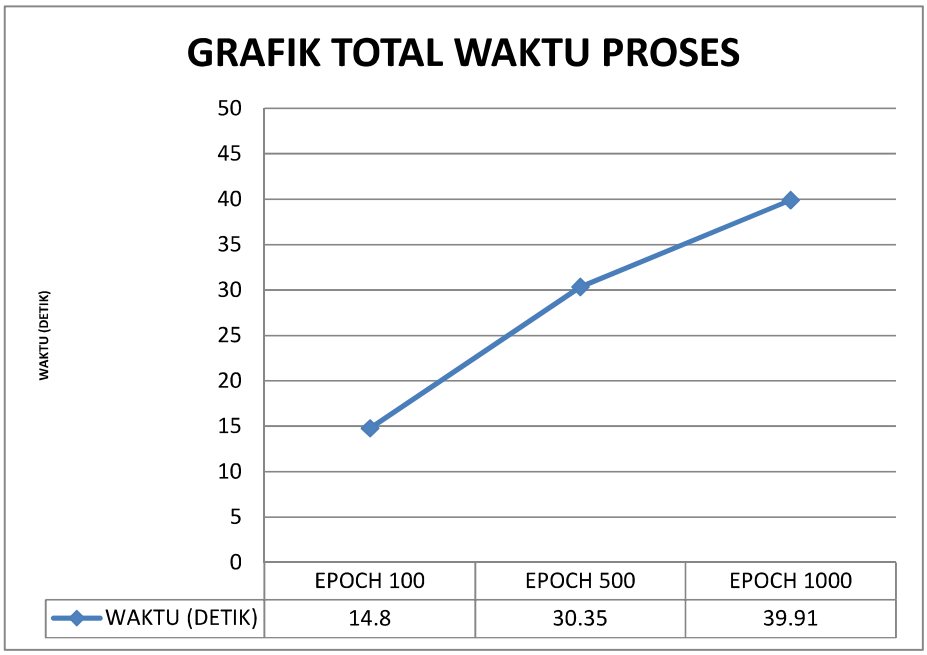
Gambar 9 Total Waktu Proses 500 epoch

Dan pada pengujian 3 dengan penggunaan epoch 1000 memiliki wkatu proses 39,91 detik dalam melakukan proses klasifikasi menggunakan BP dengan 1000 epoch dapat dilihat pada gambar 10 dibawah ini.

```
Total Waktu Proses 39.912282943725586 Detik
```

Gambar 10 Total Waktu Proses 1000 epoch

Pada pengujian yang telah dilakukan pada klasifikasi pengujian 1 dan pengujian 2 dengan menggunakan data yang berbeda terdapat peningkatan akurasi pada pengujian 2 hal ini dapat dilihat pada gambar 11 dibawah ini.



Gambar 11 Grafik Akurasi dengan 100 Epoch

Pada gambar 11 yaitu gambar grafik waktu proses klasifikasi yang dilakukan pada pengujian 1 yaitu dengan epoch 100 dan pengujian 2 dengan epoch 500 dan pengujian 3 dengan 1000 epoch . Dan dapat dilihat terjadi kenaikan pada waktu proses yang dilakukan yaitu pada pengujian epoch 100 waktu akhir proses klasifikasi yaitu selama 14.8 detik sedangkan pada epoch 500 mendapatkan waktu proses selama 30.35 detik dan pada epoch 1000 meningkat

menjadi 39.91 detik. Sehingga semakin banyak epoch maka proses klasifikasi menghasilkan waktu yang lebih lama.

D. Hasil Learning Rate 0.01 dan 0.001 Pada Epoch 100

Untuk mendapatkan learning rate yang sesuai dalam mengklasifikasi penyakit maka dilakukan dua percobaan yaitu dengan menggunakan learning rate 0.01 dan 0.001 untuk mendapatkan loss yang kecil karena dengan nilai loss yang kecil maka tingkat kesalahan pembelajaran BP dalam mengklasifikasikan suatu kelas penyakit dapat dilihat pada gambar 12 dibawah ini

```
Epoch 96/100
10/10 [==============================] - 0s 5ms/step - loss: 5.4958e-04 - acc: 1.0000
Epoch 97/100
10/10 [==============================] - 0s 5ms/step - loss: 5.4038e-04 - acc: 1.0000
Epoch 98/100
10/10 [==============================] - 0s 4ms/step - loss: 5.3131e-04 - acc: 1.0000
Epoch 99/100
10/10 [==============================] - 0s 5ms/step - loss: 5.2210e-04 - acc: 1.0000
Epoch 100/100
10/10 [==============================] - 0s 5ms/step - loss: 5.1322e-04 - acc: 1.0000
```

Gambar 12 Hasil Loss Dengan Learning Rate 0.01

Pada gambar 12 diatas dapat dilihat penggunaan learning rate dengan nilai 0.01 pada 100 epoch atau 100 perulangan dan saat perulang training ke 100 atau akhir mendapatkan nilai yang kecil yaitu 5.1322e-04 sehingga tingkat kesalahan atau error pembelajaran kecil dalam mengklasifikasikan suatu penyakit. Adapun pada percobaan kedua yaitu dengan learning rate 0.001 pada gambar 13 dibawah ini.

```
Epoch 96/100
10/10 [==============================] - 0s 5ms/step - loss: 0.0089 - acc: 1.0000
Epoch 97/100
10/10 [==============================] - 0s 7ms/step - loss: 0.0088 - acc: 1.0000
Epoch 98/100
10/10 [==============================] - 0s 5ms/step - loss: 0.0086 - acc: 1.0000
Epoch 99/100
10/10 [==============================] - 0s 6ms/step - loss: 0.0085 - acc: 1.0000
Epoch 100/100
10/10 [==============================] - 0s 6ms/step - loss: 0.0083 - acc: 1.0000
```

Gambar 13 Hasil Loss Dengan Learning Rate 0.001

Pada gambar 13 diatas dapat dilihat penggunaan learning rate dengan nilai 0.001 pada 100 epoch atau 100 perulangan dan saat perulang training ke 100 atau akhir mendapatkan nilai

yang kecil yaitu 0.0083 sehingga tingkat kesalahan atau error pembelajaran kecil dalam mengklasifikasikan suatu penyakit.