



TUGAS AKHIR - SM141501

**KLASIFIKASI TINGKAT KEPARAHAN *NON-
PROLIFERATIVE DIABETIC RETINOPATHY*
BERDASARKAN *HARD EXUDATE* MENGGUNAKAN
*EXTREME LEARNING MACHINE***

DINDA ULIMA RIZKY YANI
NRP 1213 100 081

Dosen Pembimbing
Dr. Dwi Ratna Sulistyaningrum, S.Si, MT

DEPARTEMEN MATEMATIKA
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember
Surabaya 2017



FINAL PROJECT - SM141501

**CLASSIFICATION OF NON-PROLIFERATIVE DIABETIC
RETINOPATHY'S SEVERITY BASED ON HARD
EXUDATE USING EXTREME LEARNING MACHINE**

**DINDA ULIMA RIZKY YANI
NRP 1213 100 081**

**Supervisor
Dr. Dwi Ratna Sulistyaningrum, S.Si, MT**

**DEPARTMENT OF MATHEMATICS
Faculty of Mathematics and Natural Sciences
Sepuluh Nopember Institute of Technology
Surabaya 2017**

LEMBAR PENGESAHAN

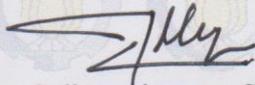
**KLASIFIKASI TINGKAT KEPARAHAN *NON-
PROLIFERATIVE DIABETIC RETINOPATHY*
BERDASARKAN *HARD EXUDATE* MENGGUNAKAN
*EXTREME LEARNING MACHINE***

***CLASSIFICATION OF NON-PROLIFERATIVE
DIABETIC RETINOPATHY'S SEVERITY BASED ON
HARD EXUDATE USING EXTREME LEARNING
MACHINE***

Diajukan untuk memenuhi salah satu syarat
Untuk memperoleh gelar Sarjana Sains pada
Bidang studi Ilmu Komputer
Program Studi S-1 Departemen Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember

Oleh:
Dinda Ulima Rizky Yani
NRP. 1213 100 081

Menyetujui,
Dosen Pembimbing,

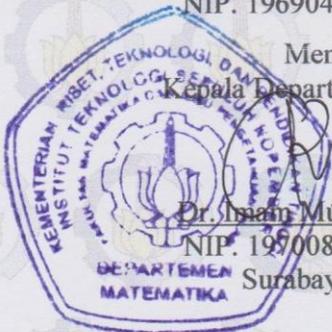


Dr. Dwi Ratna Sulistyaningrum, S.Si, MT
NIP. 19690405 199403 2 003

Mengetahui,
Kepala Departemen Matematika,

Dr. Imam Mukhlash, S.Si, MT
NIP. 19700831 199403 1 003

Surabaya, Juli 2017



**KLASIFIKASI TINGKAT KEPARAHAN *NON-
PROLIFERATIVE DIABETIC RETINOPATHY*
BERDASARKAN *HARD EXUDATE* MENGGUNAKAN
*EXTREME LEARNING MACHINE***

Nama Mahasiswa : Dinda Ulima Rizky Yani
NRP : 1213 100 081
Departemen : Matematika
Dosen Pembimbing : Dr. Dwi Ratna Sulistyaningrum, S.Si, MT

ABSTRAK

Diabetic Retinopathy dapat menyebabkan seseorang kehilangan kemampuan penglihatannya dan pada keadaan yang parah dapat mengakibatkan kebutaan. Tingkat keparahan *Non-Proliferative Diabetic Retinopathy* (NPDR) dapat diketahui dengan mendeteksi kelainan berupa *hard exudate* pada retina, namun diagnosanya tidak bisa dilakukan dengan cepat karena pengamatan retina harus melewati beberapa proses. Teknologi pengolahan citra digital berbasis *machine learning* telah banyak digunakan untuk menyelesaikan permasalahan ini. Pada Tugas Akhir ini telah dilakukan penelitian untuk mengklasifikasikan tingkat keparahan NPDR secara otomatis dengan mengekstraksi karakteristik *hard exudate* menggunakan *Gray Level Co-occurrence Matrix* (GLCM) dan *Neighborhood Gray-tone Difference Matrix* (NGTDM) kemudian menentukan tingkat keparahannya menggunakan *Extreme Learning Machine*. Hasil akurasi tertinggi yang didapat sebesar 91,22% untuk ekstraksi ciri dengan menggunakan GLCM.

***Kata Kunci : Diabetic Retinopathy, NPDR, hard exudate,
PengolahanCitra, Extreme Learning
Machine.***

**CLASSIFICATION OF NON-PROLIFERATIVE
DIABETIC RETINOPATHY'S SEVERITY BASED ON
HARD EXUDATE USING EXTREME LEARNING
MACHINE**

Name of Student :Dinda Ulima Rizky Yani
NRP : 1213 100 081
Department : Mathematics
Supervisor : Dr. Dwi Ratna Sulistyanningrum, S.Si, MT

ABSTRACT

Diabetic Retinopathy may cause vision loss ability and in severe circumstances can lead to blindness. The severity of the Non-Proliferative Diabetic Retinopathy (NPDR) can be known by detecting hard exudates on the retina. But the diagnosis can not be done quickly because the observations of this retinal must pass through several processes. Digital image processing technology and machine learning has been widely used to classify the severity of NPDR automatically by extracting the characteristic of hard exudates using Gray Level Co-occurrence Matrix (GLCM) and Neighborhood Gray-tone Difference Matrix (NGTDM) and using Extreme Learning Machine for classification. The highest accuracy results obtained is 91.22% for extract features using GLCM.

Keywords : Diabetic Retinopathy, NPDR, hard exudate, image processing, Extreme LearningMachine

KATA PENGANTAR

Segala puji syukur penulis panjatkan ke hadirat Allah SWT, karena dengan rahmat dan karunia-Nya penulis dapat menyelesaikan Tugas Akhir yang berjudul

“KLASIFIKASI TINGKAT KEPARAHAN *NON-PROLIFERATIVE DIABETIC RETINOPATHY* BERDASARKAN *HARD EXUDATE* MENGGUNAKAN *EXTREME LEARNING MACHINE* ”

yang merupakan salah satu persyaratan akademis dalam menyelesaikan Program Sarjana Departemen Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Teknologi Sepuluh Nopember Surabaya.

Tugas Akhir ini dapat diselesaikan dengan baik berkat kerja sama, bantuan, dan dukungan dari banyak pihak. Sehubungan dengan hal tersebut, penulis ingin mengucapkan terima kasih kepada :

1. Dr. Dwi Ratna Sulistyaningrum, S.Si, MT selaku Dosen Wali sekaligus Dosen Pembimbing yang telah memberikan arahan akademik selama penulis menempuh pendidikan di Departemen Matematika ITS dan memberikan bimbingan serta motivasi kepada penulis dalam mengerjakan Tugas Akhir ini sehingga dapat terselesaikan dengan baik.
2. Dr. Imam Mukhlash, S.Si, MT selaku Kepala Departemen Matematika ITS.
3. Dr. Didik Khusnul Arif, S.Si, M.Si selaku Ketua Program Studi S1 Departemen Matematika ITS.
4. Drs. Iis Herisman, M.Si selaku Sekretaris Program Studi S1 Departemen Matematika ITS.
5. Seluruh jajaran dosen dan staf Departemen Matematika ITS.
6. Keluarga tercinta yang senantiasa memberikan dukungan dan do'a dengan ikhlas.
7. Teman-teman angkatan 2013 yang saling mendukung dan memotivasi.

8. Semua pihak yang tak bisa penulis sebutkan satu-persatu, terima kasih telah membantu sampai terselesaikannya Tugas Akhir ini.

Penulis menyadari bahwa Tugas Akhir ini masih jauh dari kesempurnaan. Oleh karena itu, penulis mengharapkan kritik dan saran dari pembaca. Akhir kata, semoga Tugas Akhir ini dapat bermanfaat bagi semua pihak yang berkepentingan.

Wassalamu' alaikum Wr. Wb.

Surabaya, Juni 2017

Penulis

DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PENGESAHAN	v
ABSTRAK.....	vii
<i>ABSTRACT</i>	ix
KATA PENGANTAR.....	xi
DAFTAR ISI.....	xiii
DAFTAR GAMBAR.....	xv
DAFTAR TABEL.....	xvii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah	3
1.4 Tujuan.....	3
1.5 Manfaat.....	4
1.6 Sistematika Penulisan Tugas Akhir.....	4
BAB II TINJAUAN PUSTAKA	7
2.1 Penelitian Terdahulu.....	7
2.2 <i>Diabetic Retinopathy</i>	9
2.3 Pengolahan Citra Digital	12
2.4 Operasi Morfologi.....	15
2.5 <i>Gray Level Co-occurrence Matrix</i>	16
2.6 <i>Neighborhood Gray-tone Difference Matrix</i>	19
2.7 <i>Extreme Learning Machine</i>	21
BAB III METODE PENELITIAN.....	25
3.1 Tahapan Penelitian	25
3.2 Studi Literatur	26
3.3 Pengumpulan Data	26
3.4 Perancangan dan Implementasi.....	26
3.5 Pengujian Sistem.....	28
3.6 Penarikan Kesimpulan.....	30
3.7 Penyusunan Laporan Tugas Akhir	30
BAB IV PERANCANGAN DAN IMPLEMENTASI	31

4.1	Perancangan Sistem.....	31
4.1.1	Praproses Citra	31
4.1.2	Ekstraksi Ciri	35
4.1.3	Desain Arsitektur Extreme Learning Machine	41
4.1.4	Pemodelan Extreme Learning Machine ...	42
4.1.5	Klasifikasi dengan Extreme Learning Machine	45
4.2	Implementasi.....	47
4.2.1	Implementasi User Interface	47
4.2.2	Implementasi praproses citra	48
4.2.3	Implementasi Proses Ekstraksi Ciri dengan GLCM dan NGTDM	54
4.2.4	Implementasi Proses Klasifikasi Menggunakan Extreme Learning Machine	59
BAB V	PENGUJIAN DAN PEMBAHASAN	63
5.1	Lingkungan Pengujian Sistem	63
5.2	Pengujian Tahap Praproses Data.....	63
5.3	Pengujian Tahap Klasifikasi Menggunakan <i>Extreme Learning Machine</i>	65
5.3.1	Pengujian dengan Ekstraksi Ciri Statistik Menggunakan GLCM	66
5.3.2	Pengujian dengan Ekstraksi Ciri Statistik Menggunakan NGTDM	68
5.3.3	Pengujian dengan Ekstraksi Ciri Statistik Menggunakan GLCM dan NGTDM	69
5.4	Pembahasan Hasil Pengujian	71
BAB VI	KESIMPULAN DAN SARAN	73
6.1	Kesimpulan	73
6.2	Saran.....	74
	DAFTAR PUSTAKA.....	75
	LAMPIRAN A.....	79
	BIODATA PENULIS.....	85

DAFTAR GAMBAR

Gambar 2. 1: Retina mata (a) normal, (b) Diabetic Retinopathy	10
Gambar 2. 2: Kelainan pada retina berupa hard exudate	11
Gambar 2. 3: Representasi citra digital dalam koordinat (x,y)[10]	13
Gambar 2. 4: Orientasi sudut dari GLCM	16
Gambar 2. 5: Arsitektur Extreme Learning Machine	22
Gambar 3. 1: Diagram alir tahapan penelitian	25
Gambar 3. 2: Diagram alir proses secara umum	29
Gambar 4. 1: Diagram alir proses eliminasi optic disk.....	32
Gambar 4. 2: Diagram alir proses segmentasi exudates	34
Gambar 4. 3: Area kerja GLCM.....	36
Gambar 4. 4: (a) Matriks asal, (b) Matriks kookurensi	36
Gambar 4. 5: (a) Sebaran nilai graytone citra, (b) Entry nilai graytone ke-i, (c) Jumlah nilai graytone ke-i.....	39
Gambar 4. 6: Desain Arsitektur Extreme Learning Machine yang digunakan pada Tugas Akhir ini.....	42
Gambar 4. 7: Desain user interface klasifikasi tingkat keparahan NPDR	48
Gambar 4. 8: (a) Citra asli, (b) Red Channel, (c) Green Channel	50
Gambar 4. 9: (a) Citra asli, (b) Red Channel, (c) Green Channel	51
Gambar 5. 1: Citra input (a) moderate NPDR, (b) severe NPDR.....	64
Gambar 5. 2: Pengujian proses eliminasi optic disk pada citra: (a) moderate NPDR, (b) severe NPDR	64
Gambar 5. 3: Pengujian proses segmentasi exudate pada citra: (a) moderate NPDR, (b) severe NPDR	65

DAFTAR TABEL

Tabel 4. 1. Representasi tingkat keparahan NPDR dalam Matriks t.....	46
Tabel 5. 1 Spesifikasi perangkat keras dan perangkat lunak.	63
Tabel 5. 2 Hasil pengujian dengan ekstraksi ciri GLCM.....	67
Tabel 5. 3 Hasil perhitungan akurasi dengan ekstraksi ciri GLCM.....	67
Tabel 5. 4 Hasil pengujian dengan ekstraksi ciri NGTDM...	69
Tabel 5. 5 Hasil perhitungan akurasi dengan ekstraksi ciri NGTDM.....	69
Tabel 5. 6 Hasil pengujian dengan ekstraksi ciri GLCM dan NGTDM.....	70
Tabel 5. 7 Hasil perhitungan akurasi dengan ekstraksi ciri GLCM dan NGTDM.....	71

BAB I

PENDAHULUAN

Pada bab ini dibahas mengenai latar belakang yang mendasari penulisan Tugas Akhir ini. Di dalamnya mencakup identifikasi permasalahan pada topik Tugas Akhir kemudian dirumuskan menjadi permasalahan yang diberikan batasan-batasan dalam pembahasan pada Tugas Akhir ini.

1.1 Latar Belakang

Diabetes telah menyebabkan lebih dari tiga juta kematian di dunia setiap tahunnya. World Health Organization (WHO) melaporkan pada tahun 2014 sebanyak 422 juta orang menderita penyakit ini[1]. Jumlah ini diprediksi akan meningkat sebanyak dua kali lipat untuk 20 tahun mendatang[2]. Diabetes yang tidak terkontrol dalam jangka waktu lama dapat menyebabkan komplikasi penyakit seperti serangan jantung, stroke, gagal ginjal bahkan kebutaan. Diabetes yang paling banyak diderita adalah diabetes melitus. Menurut WHO, Indonesia menduduki peringkat keempat terbanyak penyandang diabetes melitus dan diprediksi akan mencapai 21,3 juta pada tahun 2030. Berdasarkan The DiabCare Asia 2008 Study, 42% penyandang diabetes melitus di Indonesia mengalami komplikasi retinopati[3]. Semakin lama seseorang menyandang diabetes melitus maka semakin besar pula kemungkinan ia terkena *Diabetic Retinopathy*. *Diabetic Retinopathy* merupakan kondisi dimana terdapat kelainan (luka) pada retina. Keterlambatan diagnosa *Diabetic Retinopathy* dapat menyebabkan penderita kehilangan kemampuan penglihatannya sampai akhirnya mengalami kebutaan. Tentu saja hal ini dapat menurunkan kualitas hidup penderita.

Penentuan tingkat keparahan *Diabetic Retinopathy* dilakukan dengan mendeteksi kelainan pada retina. Namun

untuk mendeteksi kelainan pada retina tidak bisa dilakukan dengan mata telanjang melainkan harus menggunakan citra fundus retina atau pengamatan retina menggunakan kamera fundus. Pada *Non-Proliferative Diabetic Retinopathy* (NPDR) dapat terjadi gejala *Retinopathy* yang berbeda seperti *microaneuriysms*, *haemorrhages*, dan *hard exudate* yang merupakan luka pada retina. *Hard exudate* memiliki ciri yang lebih menonjol dibanding gejala yang lain sehingga lebih mudah untuk diamati. *Hard exudate* merupakan bercak darah yang mengandung lemak, berwarna kekuningan dengan bentuk tak menentu. Jika timbunan lemak ini menutupi daerah makula retina maka penderita NPDR beresiko mengalami kebutaan. Gejala ini sering muncul pada penderita *Diabetic Retinopathy* tingkat sedang (moderate) dan parah (parah). Jika gejala ini muncul maka penderita NPDR harus mendapatkan penanganan yang cepat. Namun pengamatan citra fundus retina ini harus melewati beberapa proses seperti menilai warna jingga pada retina, mengamati tepian dan warna *optic disk* kemudian menilai retina secara keseluruhan sehingga diagnosa tingkat keparahan NPDR tidak bisa dilakukan dengan cepat. Untuk menyelesaikan permasalahan tersebut dibutuhkan teknologi untuk memproses citra fundus retina secara cepat dan akurat untuk membantu dokter menetapkan tindakan medis yang tepat. Teknologi yang mampu mengklasifikasikan tingkat keparahan *Diabetic Retinopathy* adalah pengolahan citra digital berbasis *machine learning*.

Pengolahan citra digital dipilih sebagai teknologi yang tepat karena mampu menganalisis suatu citra dan mengekstraksi fitur-fiturnya dengan tepat dan jelas sehingga citra mudah untuk diteliti. Teknologi pengolahan citra digital berbasis *machine learning* sudah banyak digunakan dalam bidang medis. Pada Tugas Akhir ini telah diajukan algoritma *Extreme Learning Machine* sebagai metode untuk mengklasifikasi tingkat keparahan *Non-Proliferative Diabetic*

Retinopathy. Alasan digunakannya metode *Extreme Learning Machine* adalah karena metode ini telah mampu mengklasifikasi retina mata yang normal dan abnormal secara cepat dan akurat[4].

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, dapat dirumuskan permasalahan dalam Tugas Akhir ini adalah sebagai berikut :

1. Bagaimana cara mengklasifikasi tingkat keparahan *Non–Proliferative Diabetic Retinopathy* berdasarkan *hard exudate* menggunakan *Extreme Learning Machine*?
2. Bagaimana tingkat akurasi dari metode *Extreme Learning Machine* untuk klasifikasi tingkat keparahan *Non–Proliferative Diabetic Retinopathy* berdasarkan *hard exudate*?

1.3 Batasan Masalah

Pada penelitian ini, penulis membuat batasan masalah sebagai berikut :

1. Citra fundus retina diambil dari dataset MESSIDOR.
2. Tingkat keparahan *Non–Proliferative Diabetic Retinopathy* dibagi menjadi 2 yaitu *moderate* (sedang) dan *severe* (parah).
3. Citra fundus retina yang digunakan sebagai *input* adalah citra fundus retina yang memiliki *exudate* dan teridentifikasi *moderate* NPDR atau *severe* NPDR.
4. Verifikasi dilakukan dengan membandingkan hasil klasifikasi pada tahap pengujian dengan data tingkat keparahan yang ada pada dataset MESSIDOR

1.4 Tujuan

Berdasarkan permasalahan yang telah dirumuskan sebelumnya, tujuan penelitian Tugas Akhir ini adalah :

1. Mengaplikasikan metode *Extreme Learning Machine* untuk mengklasifikasi tingkat keparahan *Non–Proliferative Diabetic Retinopathy* berdasarkan *hard exudate* secara tepat.
2. Mengetahui secara empiris tingkat keakuratan metode yang digunakan.

1.5 Manfaat

Manfaat dari Tugas Akhir ini adalah sistem yang telah dibangun dapat menunjukkan bagaimana teknologi pengolahan citra digital berbasis *machine learning* dapat mengklasifikasikan tingkat keparahan suatu penyakit.

Hasil dari penelitian ini diharapkan dapat membantu *ophthalmologists* (dokter spesialis mata) untuk mendeteksi tingkat keparahan *Non–Proliferative Diabetic Retinopathy* secara otomatis berdasarkan banyaknya *hard exudate* yang ditemukan pada retina.

1.6 Sistematika Penulisan Tugas Akhir

Sistematika dari penulisan Tugas Akhir ini adalah sebagai berikut :

1. BAB I PENDAHULUAN

Bab ini menjelaskan tentang gambaran umum dari penulisan Tugas Akhir ini yang meliputi latar belakang masalah, perumusan masalah, batasan masalah, tujuan, manfaat penelitian, dan sistematika penulisan.

2. BAB II TINJAUAN PUSTAKA

Bab ini berisi tentang materi-materi yang mendukung Tugas Akhir ini, antara lain penelitian terdahulu, *Diabetic Retinopathy*, Pengolahan Citra Digital, Operasi Morfologi, *Gray Level Co–occurrence Matrix*, *Neighborhood Gray–tone Difference Matrix*, dan *Extreme Learning Machine*.

3. **BAB III METODE PENELITIAN**
Pada bab ini dibahas tentang langkah – langkah dan metode yang digunakan untuk menyelesaikan Tugas Akhir ini.
4. **BAB IV PERANCANGAN DAN IMPLEMENTASI**
Pada bab ini dibahas tentang bagaimana tahapan–tahapan dalam perancangan dan implementasi sistem. Pembahasan perancangan sistem dimulai dari perancangan praproses data hingga proses klasifikasi. Implementasi dilakukan dengan menggunakan MATLAB R2015a berdasarkan rancangan sistem yang telah dibuat.
5. **BAB V PENGUJIAN DAN PEMBAHASAN**
Bab ini membahas tahap pengujian berdasarkan implementasi yang telah dibuat beserta pengujian kinerja sistem.
6. **BAB VI KESIMPULAN DAN SARAN**
Bab ini berisi kesimpulan Tugas Akhir yang diperoleh dari bab pengujian sistem serta saran untuk pengembangan penelitian selanjutnya.

BAB II

TINJAUAN PUSTAKA

Pada bab ini dibahas mengenai dasar teori yang digunakan dalam penyusunan Tugas Akhir ini. Dasar teori yang dijelaskan dibagi menjadi beberapa subbab yaitu *Diabetic Retinopathy*, Pengolahan Citra Digital, Operasi Morfologi, *Gray Level Co-occurrence Matrix*, *Neighborhood Gray-tone Difference Matrix*, *Extreme Learning Machine*.

2.1 Penelitian Terdahulu

Pada tahun 2013 pernah dilakukan penelitian yang berjudul “*Classification of Non-Proliferative Diabetic Retinopathy Based on Hard Exudate Using Soft Margin SVM*” oleh Handayani Tjandrasa, dkk[5]. Hal yang paling disoroti dalam penelitian tersebut adalah segmentasi *exudate* pada citra fundus retina. Hasil segmentasi *exudate* ini berupa citra biner yang kemudian di ekstraksi menggunakan parameter area, perimeter, jumlah *centroid* dan standar deviasi. Model klasifikasi ini menggunakan 147 citra yang diambil dari *database* MESSIDOR. Hasil klasifikasi pada penelitian ini menunjukkan tingkat akurasi yang baik yaitu sebesar 90,54%.

Selanjutnya pada tahun 2014 Gowthaman[6] melakukan penelitian yang berjudul “*Automatic Identification and Classification of Microaneurysms for Detection of Diabetic Retinopathy*”. Pada penelitian tersebut Gowthaman membandingkan kinerja metode SVM dan ELM untuk mengidentifikasi dan mengklasifikasi *microaneuriysms* untuk mendeteksi DR. Pada tahap *preprocessing*, *noise* pada citra fundus retina dihilangkan kemudian dilakukan penajaman kontras. Setelah itu dilakukan segmentasi *microaneuriysms* untuk memisahkan pembuluh darah dari retina lalu diekstraksi. Setelah didapatkan ekstraksi fiturnya kemudian citra diklasifikasikan kedalam 2 kelas yaitu normal (tidak terdapat *microaneuriysms*) dan *abnormal* (terdapat

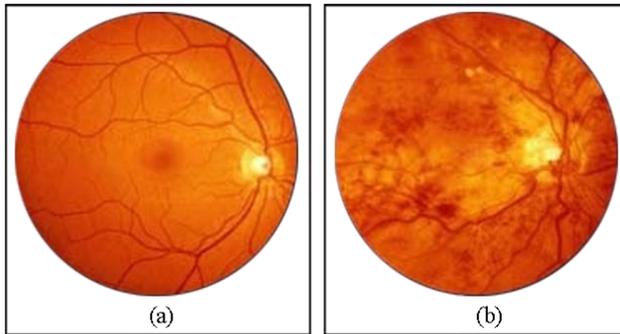
microaneuriysms). Terdapat 2 *dataset* yang digunakan dalam penelitian ini. Yang pertama adalah *dataset* DRIVE sebanyak 40 citra dengan ketentuan 20 citra latih dan 20 citra uji. Yang kedua adalah *dataset* DIABERDB1 sebanyak 89 citra dengan ketentuan 45 citra latih dan 44 citra uji. Dari hasil penelitian tersebut untuk metode SVM didapat *specificity* sebesar 95,89%, akurasi sebesar 95,83% dan *sensitivity* sebesar 95,74% dan untuk metode ELM didapat *specificity* sebesar 97,94%, akurasi sebesar 97,91% dan *sensitivity* sebesar 97,87% dengan menggunakan dataset DRIVE. Sedangkan dengan menggunakan *dataset* DIABERDB1 untuk metode SVM didapat *specificity* sebesar 95,43%, akurasi 93,65%, dan *sensitivity* sebesar 91,12% dan untuk metode ELM didapat *specificity* sebesar 98,34%, akurasi sebesar 96,58% dan *sensitivity* sebesar 94,08%. Selain itu metode ELM juga membutuhkan waktu eksekusi lebih sedikit dibandingkan dengan SVM. Artinya ELM memiliki kinerja yang sangat baik dalam hal efisiensi dan waktu komputasi.

Kemudian pada tahun 2016 Harry Pratt, dkk[7] melakukan penelitian mengenai klasifikasi *Diabetic Retinopathy* (DR) dengan judul “*Convolutonal Neural Network for Diabetic Retinopathy*”. Dalam penelitian tersebut *Diabetic Retinopathy* diklasifikasikan kedalam 5 kelas yaitu normal, *mild* DR, *moderate* DR, *severe* DR, dan *proliferative* DR menggunakan metode *Convolutonal Neural Network*. Data yang digunakan sebanyak 80000 citra yang berasal dari *dataset kaggle*. Citra fundus retina yang digunakan memiliki resolusi yang tinggi dan memori yang besar sehingga pada tahap *preprocessing* citra diubah menjadi berukuran 512x512 *pixel* agar dapat diproses oleh *software* yang digunakan. Setelah itu citra diaugmentasi dengan diputar secara acak dengan sudut 0° sampai 90°, dibalik secara horizontal dan vertikal, serta digeser acak secara horizontal dan vertikal. Pengujian dilakukan menggunakan 5000 citra yang berasal dari *dataset* yang sama. Terdapat 3 parameter keberhasilan

yaitu *specificity* (jumlah pasien yang berhasil diidentifikasi tidak memiliki DR dari total pasien yang tidak memiliki DR), *sensitivity* (jumlah pasien yang berhasil diidentifikasi memiliki DR dari total pasien yang memiliki DR), dan akurasi (jumlah pasien yang berada di kelas yang tepat). Hasil penelitian menunjukkan *specificity* sebesar 95%, akurasi sebesar 75% dan *sensitivity* sebesar 30%.

2.2 Diabetic Retinopathy

Diabetic Retinopathy merupakan komplikasi serius dari penyakit diabetes melitus yang menyebabkan menurunnya kemampuan penglihatan secara perlahan dan pada keadaan yang parah dapat menyebabkan kebutaan. *Diabetic Retinopathy* dibagi menjadi dua berdasarkan ada tidaknya kelainan pembuluh darah baru yaitu *Non-Proliferative Diabetic Retinopathy* (NPDR) dan *Proliferative Diabetic Retinopathy* (PDR)[8]. NPDR terjadi jika terdapat kerusakan pada pembuluh darah dalam retina dan terdapat cairan di atas permukaan retina. Keadaan ini merupakan stadium awal *Diabetic Retinopathy* dimana terjadi penurunan penglihatan secara perlahan sehingga tidak disadari oleh penderita. Sedangkan PDR merupakan *Diabetic Retinopathy* stadium lanjut yang berkembang dari NPDR. Pada fase ini mulai tumbuh kelainan pembuluh darah baru yang sifatnya sangat rapuh di berbagai area di permukaan retina. Jika pembuluh darah baru ini pecah maka akan terjadi pendarahan. Retina yang tertutupi oleh darah ini akan menyebabkan penglihatan tiba-tiba menjadi gelap dan bila terus dibiarkan dapat menyebabkan kebutaan permanen. Perbedaan retina mata normal dan retina mata dengan *Diabetic Retinopathy* dapat dilihat pada Gambar 2.1.

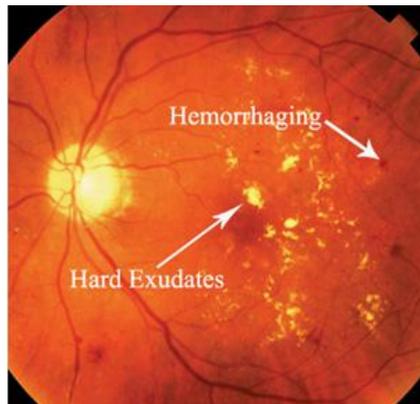


Gambar 2. 1: Retina mata (a) normal, (b) *Diabetic Retinopathy*

Penanganan untuk *Diabetic Retinopathy* dilakukan berdasarkan tingkat keparahannya. Penentuan tingkat keparahan *Diabetic Retinopathy* dilakukan dengan mendeteksi kelainan pada retina. Namun, untuk mendeteksi kelainan pada retina tidak bisa dilakukan dengan mata telanjang melainkan harus menggunakan citra fundus retina atau pengamatan retina menggunakan kamera fundus. Pemeriksaan pada penderita *Diabetic Retinopathy* mula-mula dilakukan menilai refleks retina kemudian menilai tampilan tepi dan warna *optic disk*. Setelah itu, penderita *Diabetic Retinopathy* diminta melihat ke delapan arah mata angin untuk menilai retina secara keseluruhan. Terakhir, dilakukan pemeriksaan untuk menilai makula atau daerah penglihatan sentral pada retina.

Tingkat keparahan NPDR sendiri dibagi menjadi 3 yaitu, *early NPDR*, *moderate NPDR* dan *severe NPDR* [9]. Gejala dari *Diabetic Retinopathy* yaitu munculnya luka-luka pada retina seperti *microaneuriysms*, *haemorrhages*, dan *exudate*. *Microaneuriysms* adalah bercak-bercak merah akibat bocornya pembuluh darah kecil pada retina yang merupakan tanda awal dari *Diabetic Retinopathy* (*early NPDR* atau disebut juga *mild NPDR*). *Haemorrhages* atau pendarahan terjadi akibat pecahnya dinding *microaneuriysms* dan menyebabkan bercak darah menjadi lebih besar. Gejala ini

sering muncul pada penderita *moderate* NPDR. Sedangkan *exudate* merupakan luka yang menandakan bahwa NPDR sudah berada pada fase yang kritis dan membutuhkan penanganan yang lebih serius. Terdapat 2 jenis *exudate* yaitu *soft exudate* dan *hard exudate*. *Soft exudate* merupakan kondisi dimana terjadi ketidakcukupan *supply* darah ke retina. Sedangkan *hard exudate* merupakan bercak darah berwarna kuning yang mengandung lemak dan protein seperti pada Gambar 2.2. Jika timbunan lemak ini mendekati atau berada pada bagian makula maka akan menimbulkan kebutaan. Hal ini sangat mungkin terjadi pada penderita *Diabetic Retinopathy* tingkat sedang (*moderate* NPDR) dan parah (*severe* NPDR).



Gambar 2. 2: Kelainan pada retina berupa *hard exudate*

Gejala-gejala kelainan pada retina sudah muncul sejak fase awal *Diabetic Retinopathy* namun biasanya penderita baru menyadarinya pada stadium lanjut. Oleh sebab itu, pemeriksaan *Diabetic Retinopathy* merupakan hal yang sangat penting mengingat penderita *Diabetic Retinopathy* tidak menyadari gejala awal yang timbul pada retina mata mereka. Penanganan NPDR bisa dilakukan dengan meningkatkan

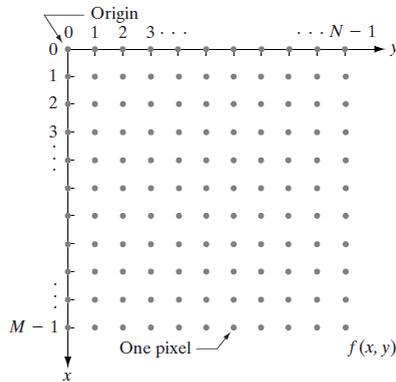
kesehatan tubuh penderita untuk mencegah perkembangan dari penyakit diabetes yang dideritanya. Namun, penanganan yang dilakukan pada *Diabetic Retinopathy* stadium akhir (PDR) akan sangat berbeda karena penanganannya biasa mungkin tidak efektif. Penderita PDR harus menjalani terapi untuk mencegah munculnya pembuluh darah baru, salah satunya adalah *panretinal photocoagulation* yaitu terapi menggunakan laser untuk membakar dan menghancurkan beberapa bagian pada retina. Oleh karena itu, keterlambatan diagnosa *Diabetic Retinopathy* akan menyebabkan berbagai hal yang tidak diinginkan seperti hilangnya kemampuan penglihatan secara tiba-tiba, komplikasi penyakit yang lebih serius, bahkan kebutaan total.

2.3 Pengolahan Citra Digital

Pengolahan citra merupakan proses pengolahan dan analisis citra yang melibatkan persepsi visual. Pengolahan citra memiliki ciri yaitu data masukan dan informasi keluaran dalam bentuk citra atau masukan berupa citra dan keluaran berupa informasi yang diekstrak dari citra. Pengolahan citra dilakukan untuk menganalisis dan memperbaiki citra agar lebih mudah diinterpretasi oleh mata manusia. Sesungguhnya citra merupakan kumpulan intensitas cahaya dalam bidang dua dimensi. Oleh karena itu sebuah citra didefinisikan sebagai fungsi dua dimensi, $f(x,y)$, dengan x dan y merupakan koordinat spasial. Sehingga bisa dikatakan bahwa citra atau *image* merupakan representasi dari suatu obyek dalam bidang dua dimensi yang ditulis dalam koordinat kartesian $x - y$. Jika nilai x, y dan f bernilai diskrit dan terbatas maka citra tersebut merupakan citra digital[10].

Pengolahan citra digital merupakan proses pengolahan dan analisis citra digital dengan menggunakan komputer[11]. Setiap koordinat dalam sebuah citra digital mewakili satu elemen terkecil dari obyek yang biasanya disebut piksel. Setiap satu piksel pada citra digital mewakili nilai intensitas

cahaya. Oleh karena itu citra dianggap sebagai sebuah matrik $x-y$ yang berisi nilai piksel. Sebuah citra digital yang memiliki M baris dan N kolom direpresentasikan oleh Gambar 2.3.



Gambar 2. 3: Representasi citra digital dalam koordinat (x,y) [10]

Beberapa langkah fundamental dalam proses pengolahan citra digital antara lain:

1. *Image enhancement*

Sebuah citra terkadang memiliki kualitas yang buruk. *Image enhancement* atau perbaikan citra dilakukan untuk meningkatkan kualitas citra. Peningkatan kualitas citra biasanya dilakukan dengan memanipulasi parameter-parameter yang ada pada citra. Fitur perbaikan kualitas citra ini akan sangat berguna untuk proses ekstraksi ciri, analisis citra dan tampilan informasi visual citra[12]. Salah satu contoh dari operasi perbaikan citra adalah penapisan derau (*noise filtering*). *Noise* pada citra bisa menyebabkan citra menjadi terlalu kontras, dan kurang tajam. Kontras menyatakan sebaran terang (*lightness*) dan gelap (*darkness*) di dalam sebuah citra. Citra dengan kontras yang bagus tidak memiliki nilai keabuan yang

mendominasi. Artinya histogram citranya memperlihatkan sebaran nilai keabuan yang relatif seragam.

2. Segmentasi citra

Proses segmentasi merupakan elemen paling penting dalam analisis citra secara otomatis karena pada langkah ini obyek yang menjadi perhatian akan diekstrak dari citra untuk diproses selanjutnya seperti dideskripsikan atau dikenali[13]. Proses segmentasi secara otomatis merupakan bagian paling sulit dari pengolahan citra digital. Karena proses segmentasi yang dilakukan secara kasar akan membuat penelitian jauh dari kata sukses. Begitu juga sebaliknya, segmentasi yang terlalu lemah dan tidak menentu akan menyebabkan kegagalan pada akhirnya. Pada umumnya, semakin akurat sebuah segmentasi maka semakin tinggi pula kemungkinan pengenalan citra untuk sukses[10]. Algoritma segmentasi citra bekerja berdasarkan 2 hal, yaitu diskontinuitas dan similaritas. Pengelompokan citra pada pendekatan pertama (diskontinuitas) dilakukan berdasarkan perubahan intensitas yang mendadak, seperti perbedaan warna dasar citra dan warna tepi obyek. Sedangkan pengelompokan pada pendekatan kedua (similaritas) dilakukan berdasarkan prinsip kemiripan area citra menurut kriteria yang telah ditentukan.

3. Representasi dan deskripsi

Citra yang telah disegmentasi menjadi area-area tertentu kemudian direpresentasikan dan dideskripsikan menjadi bentuk yang lain agar bisa diproses lebih lanjut. Representasi dari segmentasi citra dapat berupa data piksel-piksel di perbatasan obyek maupun di area-area obyek. Deskripsi dari citra berisi data mengenai perbatasan citra, tekstur obyek dan lain-lain.

2.4 Operasi Morfologi

Operasi Morfologi adalah teknik pengolahan citra yang didasarkan pada bentuk segmen dalam citra dengan mengimplementasikan teori himpunan. Segmentasi dilakukan dengan membedakan antara obyek dan *background*. Karena difokuskan pada suatu obyek, maka operasi ini biasanya diterapkan pada citra biner. Kemudian algoritma morfologi berkembang sehingga operasi morfologi sederhana seperti erosi, dilasi, *opening*, dan *closing* dapat diterapkan pada citra *grayscale*. Hasil operasi morfologi dapat dimanfaatkan untuk pengambilan keputusan dengan analisis lebih lanjut.

Misalkan terdapat sebuah citra digital $f(x,y)$ dan struktur elemen $b(x,y)$ dengan $(x,y) \in Z \times Z$ dan f dan b adalah fungsi yang menentukan nilai keabuan citra[10]. Maka:

- a. Erosi dapat dihitung dengan menggunakan persamaan berikut:

$$(f \ominus b)(s,t) = \min \{ f(s+x, t+y) - b(x,y) \mid (s+x), (t+y) \in D_f; (x,y) \in D_b \} \quad (2.1)$$

dengan D_f dan D_b masing-masing adalah domain dari f dan b .

- b. Dilasi dapat dihitung dengan menggunakan persamaan berikut:

$$(f \oplus b)(s,t) = \max \{ f(s-x, t-y) + b(x,y) \mid (s-x), (t-y) \in D_f; (x,y) \in D_b \} \quad (2.2)$$

- c. *Opening*, kombinasi antara operasi erosi yang diikuti dengan operasi dilasi dengan struktur elemen yang sama dapat dihitung dengan menggunakan persamaan berikut:

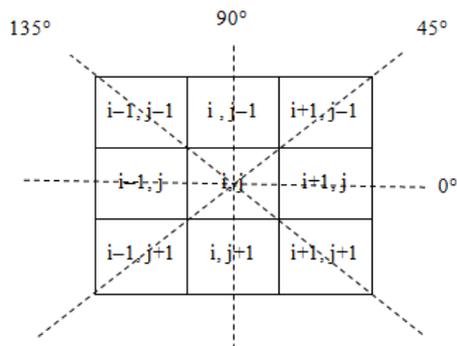
$$f \circ b = (f \ominus b) \oplus b \quad (2.3)$$

- d. *Closing*, operasi kebalikan dari *opening* yaitu operasi dilasi diikuti dengan operasi erosi dengan struktur elemen yang sama dapat dihitung dengan menggunakan persamaan berikut:

$$f \bullet b = (f \oplus b) \ominus b \quad (2.4)$$

2.5 Gray Level Co-occurrence Matrix

Gray Level Co-occurrence Matrix (GLCM) pertama kali diperkenalkan oleh Haralick[14] untuk mengukur karakteristik tekstur orde kedua yang memegang peranan penting dalam penglihatan manusia. GLCM merupakan metode yang sering digunakan untuk mengekstraksi ciri citra menggunakan *gray level matrix* atau matrik derajat keabuan. *Co-occurrence matrix* atau matriks kookurensi merupakan matriks yang merepresentasikan hubungan ketetanggaan antar piksel pada berbagai orientasi sudut (θ) dan jarak spasial (d). Jarak dinyatakan dalam piksel dan orientasi dinyatakan dalam derajat. Orientasi dibentuk dalam empat arah sudut dengan interval sudut 45° , yaitu 0° , 45° , 90° , dan 135° .



Gambar 2. 4: Orientasi sudut dari GLCM

Misal sebuah citra berukuran $N_y \times N_x$ memiliki K tingkat keabuan. $L_x = \{1, 2, \dots, N_x\}$ adalah domain spasial horizontal dan $L_y = \{1, 2, \dots, N_y\}$ adalah domain spasial vertikal, $G = \{1, 2, \dots, K\}$ adalah himpunan tingkat keabuan citra, maka citra I dapat didefinisikan oleh:

$$I : L_y \times L_x \rightarrow G$$

yaitu fungsi yang menentukan nilai keabuan di G untuk setiap pasangan koordinat di $L_y \times L_x$. Kemudian diasumsikan bahwa informasi ciri tekstur dari citra cukup ditentukan oleh

matriks frekuensi relatif Q dari dua sel ketetanggaan dengan jarak d . Sel pertama adalah sel yang memiliki nilai keabuan ke- i dan sel kedua adalah sel yang memiliki nilai keabuan ke- j . Berdasarkan orientasi sudutnya, frekuensi relatif $Q(i, j)$ didefinisikan oleh:

$$Q(i, j, d, 0^\circ) = \#\{(k, l), (m, n) \in (L_y \times L_x) \times (L_y \times L_x) \mid k - m = 0, |l - n| = d, I(k, l) = i, I(m, n) = j\} \quad ,$$

untuk jarak d dan orientasi sudut 0°

$$Q(i, j, d, 45^\circ) = \#\{(k, l), (m, n) \in (L_y \times L_x) \times (L_y \times L_x) \mid (k - m = d, l - n = -d) \text{ atau } (k - m = -d, l - n = d), I(k, l) = i, I(m, n) = j\} \quad ,$$

untuk jarak d dan orientasi sudut 45°

$$Q(i, j, d, 90^\circ) = \#\{(k, l), (m, n) \in (L_y \times L_x) \times (L_y \times L_x) \mid |k - m| = d, l - n = 0, I(k, l) = i, I(m, n) = j\} \quad ,$$

untuk jarak d dan orientasi sudut 90°

$$Q(i, j, d, 135^\circ) = \#\{(k, l), (m, n) \in (L_y \times L_x) \times (L_y \times L_x) \mid (k - m = d, l - n = d) \text{ atau } (k - m = -d, l - n = -d), I(k, l) = i, I(m, n) = j\} \quad ,$$

untuk jarak d dan orientasi sudut 135° , dengan $\#$ menyatakan banyaknya elemen dalam himpunan. Perlu diketahui bahwa matrik Q ini simetris sedemikian hingga $Q(i, j; d, a) = Q(j, i; d, a)$.

Elemen–elemen yang dapat diekstrak menggunakan GLCM antara lain:

- a) *Angular Second Moment* (ASM), yaitu ukuran keseragaman tekstur pada citra. ASM dapat dihitung menggunakan persamaan berikut:

$$ASM = \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} [q(i, j)]^2 \quad (2.5)$$

dengan,

$q(i, j)$ = elemen ke (i, j) dari matrik frekuensi relatif Q (Matriks kookurensi)

K = jumlah *gray level* yang berbeda dari citra

- b) *Contrast*, digunakan untuk mengukur kontras keabuan atau variasi intensitas lokal pada citra. *Contrast* dapat dihitung menggunakan persamaan berikut:

$$Contrast = \sum_{n=0}^{K-1} n^2 \left\{ \sum_{i=1}^K \sum_{j=1}^K q(i, j) \right\}, \quad |i - j| = n \quad (2.6)$$

- c) *Correlation*, yaitu ukuran ketergantungan linear dari tingkat keabuan diantara piksel yang bertetangga. *Correlation* dapat dihitung menggunakan persamaan berikut:

$$Correlation = \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} \frac{(ij)q(i, j) - \mu_x \mu_y}{\sigma_x \sigma_y} \quad (2.7)$$

dengan,

μ_x = mean dari q_x

μ_y = mean dari q_y

σ_x = standar deviasi dari q_x

σ_y = standar deviasi dari q_y

- d) *Dissimilarity* menggunakan faktor pembobotan yang sama dengan nilai mutlak dari perbedaan tingkat keabuan. Hasilnya hampir sama dengan *contrast*. *Dissimilarity* dapat dihitung menggunakan persamaan berikut:

$$Dissimilarity = \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} q(i, j) |i - j| \quad (2.8)$$

- e) *Inverse Difference Moment* (IDM), yaitu ukuran homogenitas citra keabuan yang sejenis. IDM dapat dihitung menggunakan persamaan berikut:

$$IDM = \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} \frac{1}{1 + (i - j)^2} q(i, j) \quad (2.9)$$

- f) *Entropy*, yaitu ukuran ketidak aturan bentuk pada citra. Entropy dapat dihitung menggunakan persamaan berikut:

$$Entropy = - \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} q(i, j) \log(q(i, j)) \quad (2.10)$$

2.6 Neighborhood Gray-tone Difference Matrix

Neighborhood Gray-tone Difference Matrix (NGTDM) digunakan untuk menggambarkan fitur lokal dengan menghitung parameter tekstur tingkat tinggi. *Gray-tone* merupakan tingkat kegelapan atau tingkat kecerahan obyek pada citra. Misalkan $f(k, l)$ *gray-tone* pada titik (k, l) yang mempunyai nilai *gray tone* ke- i , maka untuk mencari *gray-tone* rata-rata dari sekitaran (k, l) menggunakan persamaan:

$$\bar{A}_i = \bar{A}(k, l) = \frac{1}{W - 1} \left[\sum_{m=-d}^d \sum_{n=-d}^d f(k + m, l + n) \right]_{(m, n) \neq (0, 0)} \quad (2.11)$$

dengan d adalah jarak yang menentukan ukuran persekitaran dan $W = (2d + 1)^2$. Kemudian masukan ke- i pada NGTDM ditunjukkan oleh persamaan:

$$s(i) = \begin{cases} \sum |i - \bar{A}_i|, & \text{untuk } i \in N_i \text{ jika } N_i \neq 0, \\ 0 & \text{untuk } i \text{ yang lain} \end{cases} \quad (2.12)$$

dengan N_i merupakan himpunan semua piksel yang memiliki *gray tone* i (kecuali pada sekeliling area luasan d).

Karakteristik tekstur citra yang diturunkan dari NGTDM sesuai dengan karakteristik citra menurut persepsi manusia. Parameter-parameter yang diturunkan dari NGTDM [15] antara lain:

a. *Coarseness*

Coarseness atau kekasaran merupakan unsur paling penting dari karakteristik tekstur. Pada tekstur kasar pola yang terbentuk sangat besar. Akibatnya citra cenderung memiliki tingkat keseragaman intensitas lokal yang tinggi

sehingga perbedaan intensitas piksel yang bertetangga kecil. Perbedaan intensitas itu dihitung untuk mengetahui tingkat kekasaran citra. *Coarseness* dapat dihitung menggunakan persamaan berikut:

$$f_{cos} = \left(\varepsilon + \sum_{i=0}^{G_h} p_i s(i) \right)^{-1} \quad (2.13)$$

dengan,

G_h = nilai *gray-tone* tertinggi dari citra

ε = bilangan yang sangat kecil yang mencegah f_{cos} menjadi tak berhingga

p_i = probabilitas kookurensi nilai *gray-tone* ke i

Nilai p_i dapat dihitung menggunakan persamaan berikut:

$$p_i = N_i/n^2 \quad (2.14)$$

dengan $n = N - 2d$

b. *Contrast*

Sebuah citra dirasa memiliki kontras yang tinggi jika perbedaan intensitas piksel yang bertetangga terlihat jelas. Contohnya adalah ketika citra ditarik (*stretched*) muncul kotak-kotak kecil pada daerah yang memiliki kontras tinggi dan kotak-kotak kasar untuk daerah dengan tingkat keabuan yang sama. *Contrast* dapat dihitung menggunakan persamaan berikut:

$$f_{con} = \left[\frac{1}{N_g(N_g - 1)} \sum_{i=0}^{G_h} \sum_{j=0}^{G_h} p_i p_j (i - j)^2 \right] \left[\frac{i}{n^2} \sum_{i=0}^{G_h} s(i) \right] \quad (2.15)$$

dengan N_g merupakan total tingkat keabuan yang berbeda pada citra.

c. *Busyness*

Tekstur *busyness* adalah kondisi dimana terjadi perubahan intensitas yang sangat tinggi dengan cepat dari satu piksel ke tetangganya. Frekuensi perubahan intensitas mencerminkan tingkat *busyness* dari citra. Besarnya perubahan ini tergantung pada rentang skala keabuan dan

berhubungan dengan kontras. *Busyness* dapat dihitung menggunakan persamaan berikut:

$$f_{bus} = \left[\frac{\sum_{i=0}^{G_h} p_i s(i)}{\sum_{i=0}^{G_h} \sum_{j=0}^{G_h} i p_i - j p_j} \right], \quad p_i \neq 0, p_j \neq 0 \quad (2.16)$$

d. *Complexity*

Complexity atau kompleksitas mengacu pada isi informasi visual dari tekstur. Sebuah tekstur dianggap kompleks jika berisi informasi yang tinggi. Tekstur dengan perubahan intensitas yang cepat lebih kompleks daripada tekstur yang memiliki tingkat tinggi keseragaman lokal dalam intensitas. *Complexity* dapat dihitung menggunakan persamaan berikut:

$$f_{com} = \sum_{i=0}^{G_h} \sum_{j=0}^{G_h} \left(\frac{|i-j|}{n^2(p_i + p_j)} \right) (p_i s(i) + p_j s(j)), \quad (2.17)$$

$p_i \neq 0, p_j \neq 0$

e. *Texture Strength*

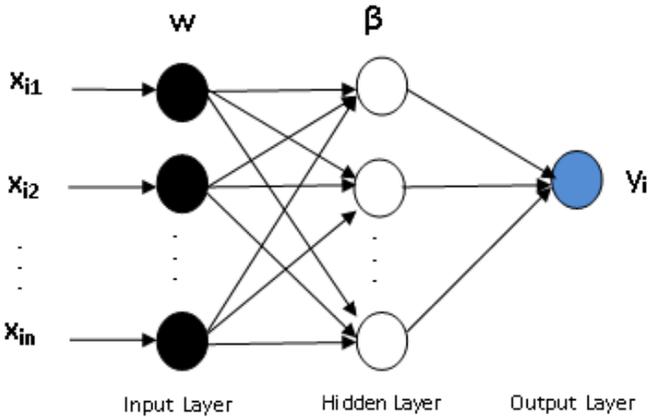
Sebuah tekstur dikatakan kuat jika memiliki pola yang terlihat jelas dan mudah didefinisikan sehingga terlihat menarik karena menyajikan informasi visual yang tinggi. Tekstur kasar dapat dibedakan dengan mengamati perbedaan kecil antara intensitas rata-rata mereka. Namun untuk tekstur *busy* harus terdapat perbedaan besar antar intensitas mereka. Oleh karena itu kekuatan tekstur bergantung kepada korelasi antara kekasaran dan kontras. *Texture Strength* dapat dihitung menggunakan persamaan berikut:

$$f_{str} = \frac{\sum_{i=0}^{G_h} \sum_{j=0}^{G_h} (p_i + p_j)(i-j)^2}{\varepsilon + \sum_{i=0}^{G_h} s(i)}, \quad p_i \neq 0, p_j \neq 0 \quad (2.18)$$

2.7 *Extreme Learning Machine*

Extreme Learning Machine merupakan algoritma pembelajaran yang sederhana untuk *Single Hidden Layer*

Feed forward Neural network (SNFLs). ELM hanya tersusun dari 1 *input layer*, 1 *hidden layer*, dan 1 *output layer*. Waktu pembelajaran ELM lebih cepat dari algoritma pembelajaran *feedforward* tradisional lainnya karena ELM secara acak memilih bobot dan secara analitik menentukan bobot *output* dari SNFLs [4].



Gambar 2. 5: Arsitektur *Extreme Learning Machine*

Untuk setiap pasangan input dan target (x_i, t_i) dengan $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in \mathbb{R}^n$ dan $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in \mathbb{R}^m$, standar SNFLs dengan \tilde{N} *hidden neuron* dan fungsi aktivasi $g(x)$ secara matematika dimodelkan sebagai persamaan berikut [16]:

$$\sum_{i=1}^{\tilde{N}} \beta_i g_i(x_j) = \sum_{i=1}^{\tilde{N}} \beta_i g(w_i \cdot x_j + b_i) = o_j, \quad (2.19)$$

$$j = 1, \dots, N$$

dengan $w_i = [w_{i1}, w_{i2}, \dots, w_{in}]^T$ merupakan bobot yang menghubungkan *input neuron* dengan *hidden neuron* ke- i , $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^T$ merupakan bobot yang menghubungkan *hidden neuron* ke- i dengan *output neuron*, dan b_i merupakan *threshold* dari *hidden neuron* ke- i .

Standar SFNLs dengan \tilde{N} *hidden neuron* dan fungsi aktivasi $g(x)$ dapat memperkirakan N sampel ini dengan *error* mendekati nol yang artinya $\sum_{j=1}^N \|o_j - t_j\| = 0$ dengan $\|\cdot\|$ merupakan norma *Euclid* di \mathbb{R} . Dengan kata lain terdapat β_i, w_i dan b_i sedemikian hingga

$$\sum_{i=1}^{\tilde{N}} \beta_i g(w_i \cdot x_j + b_i) = t_j, \quad j = 1, \dots, N \quad (2.20)$$

Persamaan diatas dapat ditulis sebagai berikut:

$$H\beta = T \quad (2.21)$$

dengan,

$$H = \begin{bmatrix} g(w_1 \cdot x_1 + b_1) & \cdots & g(w_{\tilde{N}} \cdot x_1 + b_{\tilde{N}}) \\ \vdots & \cdots & \vdots \\ g(w_i \cdot x_N + b_i) & \cdots & g(w_{\tilde{N}} \cdot x_N + b_{\tilde{N}}) \end{bmatrix}_{N \times \tilde{N}} \quad (2.22)$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_{\tilde{N}}^T \end{bmatrix}_{\tilde{N} \times m} \quad \text{dan} \quad T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N \times m} \quad (2.23)$$

dengan H merupakan matriks keluaran *hidden layer* dari *neural network* dan kolom ke- i dari matriks H merupakan vektor keluaran dari *hidden neuron* ke- i dengan *input* x_1, x_2, \dots, x_N .

Algoritma *Extreme Learning Machine* adalah sebagai berikut: Diberikan *training set* $\mathfrak{X} = \{(x_i, t_i) | x_i \in \mathbb{R}^n, t_i \in \mathbb{R}^m, i = 1, \dots, N\}$, fungsi aktivasi $g(x)$, dan *hidden layer* dengan jumlah neuron sebanyak \tilde{N} ,

Langkah 1: tetapkan bobot w_i dan bias b_i secara acak, dengan $i = 1, \dots, \tilde{N}$

Langkah 2: Hitung *hidden layer output* matriks H

Langkah 3: Hitung bobot *output* β :

$$\beta = H^\dagger T \quad (2.24)$$

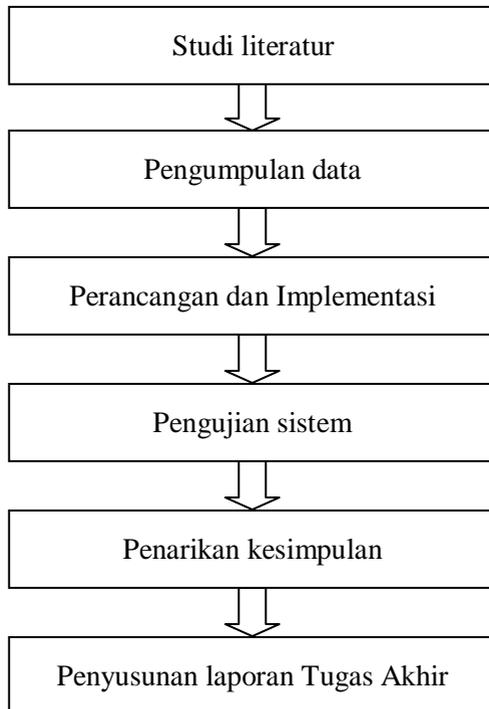
dengan H^\dagger merupakan *Moore-Penrose generalized inverse* dari H .

BAB III METODE PENELITIAN

Pada bab ini dijelaskan langkah-langkah yang digunakan dalam penyusunan Tugas Akhir. Disamping itu, dijelaskan pula prosedur dan proses pelaksanaan tiap-tiap langkah yang dilakukan dalam menyelesaikan Tugas Akhir.

3.1 Tahapan Penelitian

Gambaran tahap-tahap dalam penelitian pada Tugas Akhir ini disajikan sebagai diagram alir pada Gambar 3.1.



Gambar 3. 1: Diagram alir tahapan penelitian

3.2 Studi Literatur

Studi Literatur ini dilakukan untuk mengidentifikasi permasalahan dengan mencari referensi yang menunjang penelitian yang berupa Tugas Akhir, jurnal internasional, buku, maupun artikel yang berhubungan dengan topik Tugas Akhir ini.

3.3 Pengumpulan Data

Pengumpulan Data merupakan tahap untuk mengumpulkan data yang diperlukan dalam pengerjaan Tugas Akhir ini yaitu citra fundus retina. Data citra fundus retina yang digunakan diambil dari dataset MESSIDOR (*Methods to Evaluating Segmentation and Indexing techniques Dedicated to Retinal Ophthalmology*). MESSIDOR merupakan sebuah penelitian yang didanai oleh Kementerian Penelitian dan Pertahanan Perancis dalam program *techno-vision* 2004 yang didukung oleh 11 konsorsium yang terdiri dari rumah sakit *ophthalmology*, laboratorium, dan universitas di Perancis[17]. Data citra fundus retina yang digunakan yaitu retina yang menandung *hard exudate* dan teridentifikasi NPDR tingkat sedang (*moderate*) dan NPDR parah (*severe*). Data ini nantinya akan dibagi ke dalam dua kelompok yaitu data *training* dan data *testing*.

3.4 Perancangan dan Implementasi

Pada tahap ini penulis membuat perancangan dan implementasi dari sistem klasifikasi tingkat keparahan NPDR yang alurnya digambarkan pada Gambar 3.2 agar mudah dipahami oleh pembaca. Implementasi tersebut dibuat dengan menggunakan Matlab R2015a untuk proses *training* dan *testing*. Tahap-tahap yang dilakukan adalah sebagai berikut :

- a. Segmentasi Exudate

Proses segmentasi dilakukan untuk menemukan daerah *hard exudates* pada citra fundus retina. Awalnya dilakukan proses perbaikan kualitas citra untuk

mempertegas daerah gelap dan terang pada citra fundus retina. Setelah itu dilakukan eliminasi *optic disk* untuk menghilangkan daerah palsu yang memiliki kesamaan ciri dengan *exudate*.

b. Ekstraksi Ciri

Pada tahap ini *exudate* yang telah tersegmentasi diekstrak fitur-fiturnya. Informasi yang diekstrak dari citra *exudate* tersegmentasi adalah ciri statistik orde kedua. Ciri statistik yang diekstrak pada tahap ini yaitu *angular second moment (energy)*, *contrast*, *correlation*, *dissimilarity*, *inverse difference moment*, *entropy*, *coarseness*, *busyness*, *complexity*, dan *texture strength*. Untuk mendapatkan semua fitur tersebut digunakan metode GLCM (*Gray Level Co-occurrence Matrix*) dan NGTDM (*Neighborhood Gray-tone Difference Matrix*). Hasil dari proses ekstraksi ciri ini berupa sebuah vektor berisi fitur-fitur yang diekstrak dari satu citra fundus retina.

c. Klasifikasi

Pada penelitian ini *Non-Proliferative Diabetic Retinopathy* diklasifikasikan kedalam 2 kelas yaitu *moderate* (sedang) dan *severe* (parah). Klasifikasi dilakukan berdasarkan karakteristik *hard exudate* yang ditemukan pada retina. Metode yang digunakan untuk mengklasifikasi tingkat keparahan *Non-Proliferative Diabetic Retinopathy* adalah *Extreme Learning Machine*. Pada proses ini terdapat dua tahap, yaitu tahap *training* dan *testing*. Pada tahap *training* diperlukan *input* berupa citra fundus retina yang teridentifikasi *Diabetic Retinopathy* serta tingkat keparahannya untuk memperoleh bobot dan bias. Kemudian dilakukan tahap *testing* menggunakan bobot yang diperoleh pada tahap *training* dengan *input* berupa citra fundus retina yang teridentifikasi *Diabetic Retinopathy* dan akan memberikan *output* berupa tingkat keparahannya.

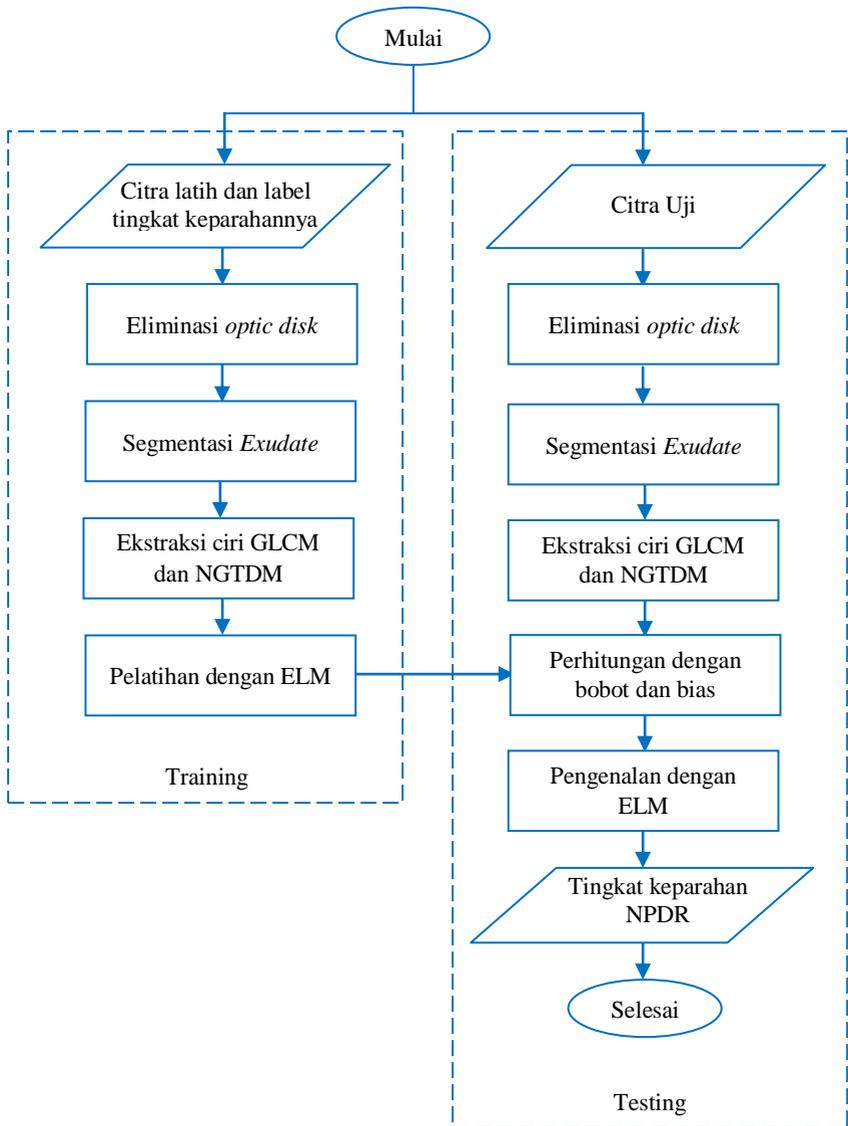
Tahap *training* dan *testing* menggunakan citra yang berbeda namun berasal dari *dataset* yang sama.

3.5 Pengujian Sistem

Pada tahap ini dilakukan pengujian untuk memeriksa apakah hasil *output* yang dihasilkan sudah sesuai yang diharapkan. Pada tahap *testing* diperlukan bobot yang dihasilkan dari tahap *training* dan *input* berupa citra fundus retina. *Output* yang dihasilkan berupa label tingkat keparahan NPDR. Hasil keluaran ini kemudian dihitung tingkat akurasi sebagai berikut:

$$AC = \frac{x}{n} \times 100\% \quad (3.1)$$

dengan AC adalah persentase akurasi, x adalah jumlah citra yang berada di kelas yang benar, dan n adalah jumlah seluruh citra yang diuji pada tahap *testing*.



Gambar 3. 2: Diagram alir proses secara umum

3.6 Penarikan Kesimpulan

Setelah implementasi dan pengujian selesai dilakukan, maka tahap selanjutnya adalah penarikan kesimpulan dari keseluruhan tahap yang sudah dilakukan dan pemberian saran sebagai masukan untuk pengembangan penelitian lebih lanjut.

3.7 Penyusunan Laporan Tugas Akhir

Setelah semua proses selesai dilakukan maka tahap terakhir adalah penyusunan laporan Tugas Akhir.

BAB IV

PERANCANGAN DAN IMPLEMENTASI

Bab ini menjelaskan rancangan dan implementasi sistem dari Tugas Akhir ini. Desain sistem menggambarkan proses rancang bangun secara terperinci dari awal tahap praproses data hingga proses konstruksi sistem menggunakan metode *Extreme Learning Machine*, serta penjelasan mengenai cara untuk mendapatkan data keluaran yang sesuai dengan tujuan dari penelitian Tugas Akhir ini.

4.1 Perancangan Sistem

Tampilan dari sistem klasifikasi tingkat keparahan NPDR dibangun dengan tampilan sederhana. *Software* yang digunakan untuk membangun sistem ini adalah MATLAB R2015a.

4.1.1 Praproses Citra

Obyek yang teliti dalam Tugas Akhir ini adalah *exudate* yang merupakan gejala penyakit *Diabetic Retinopathy*. Data masukan dalam sistem klasifikasi tingkat keparahan NPDR ini adalah citra fundus retina yang teridentifikasi *moderate* NPDR (NPDR tingkat sedang) dan *severe* NPDR (NPDR tingkat parah) dengan format .tif dan berupa citra RGB. Tahap praproses dilakukan untuk menentukan wilayah *exudate* pada citra fundus retina. Karena citra fundus retina yang digunakan merupakan citra yang tidak melalui proses kompresi sehingga memiliki ukuran yang cukup besar maka dilakukan proses *resize* normalisasi menjadi ukuran 384x576 piksel agar citra dapat diolah dengan mudah oleh sistem.

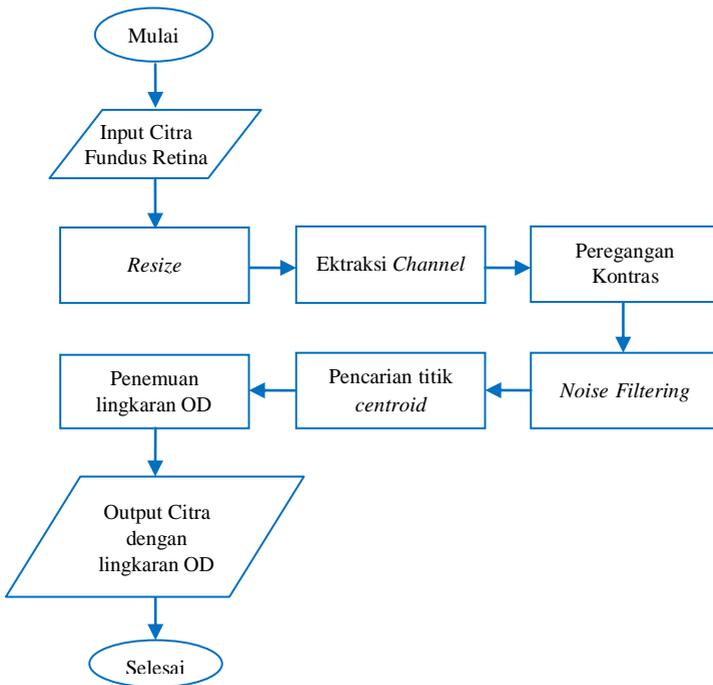
4.1.1.1 Eliminasi *Optic Disk*

Optic disc merupakan salah satu bagian utama pada retina. Sebelum dilakukan proses segmentasi *exudate*, lingkaran *optic disk* harus dieliminasi dari citra dengan tujuan

untuk menghilangkan daerah palsu yang memiliki kesamaan ciri dengan *exudate*. Proses eliminasi *optic disk* ditunjukkan pada Gambar 4.1.

Tahap-tahap yang dilakukan pada proses eliminasi *optic disk* yaitu:

1. Mengekstraksi citra RGB yang telah *resize* ke dalam *Red Channel* dan *Green Channel*.
2. Melakukan peregangkan kontras agar area-area gelap pada citra menjadi lebih terang.
3. Melakukan *filtering* untuk menghilangkan *noise* dari citra hasil peregangkan kontras.
4. Mencari titik *centroid* sebagai titik tengah lingkaran *optic disk*.
5. Menentukan posisi dan wilayah lingkaran *optic disk*



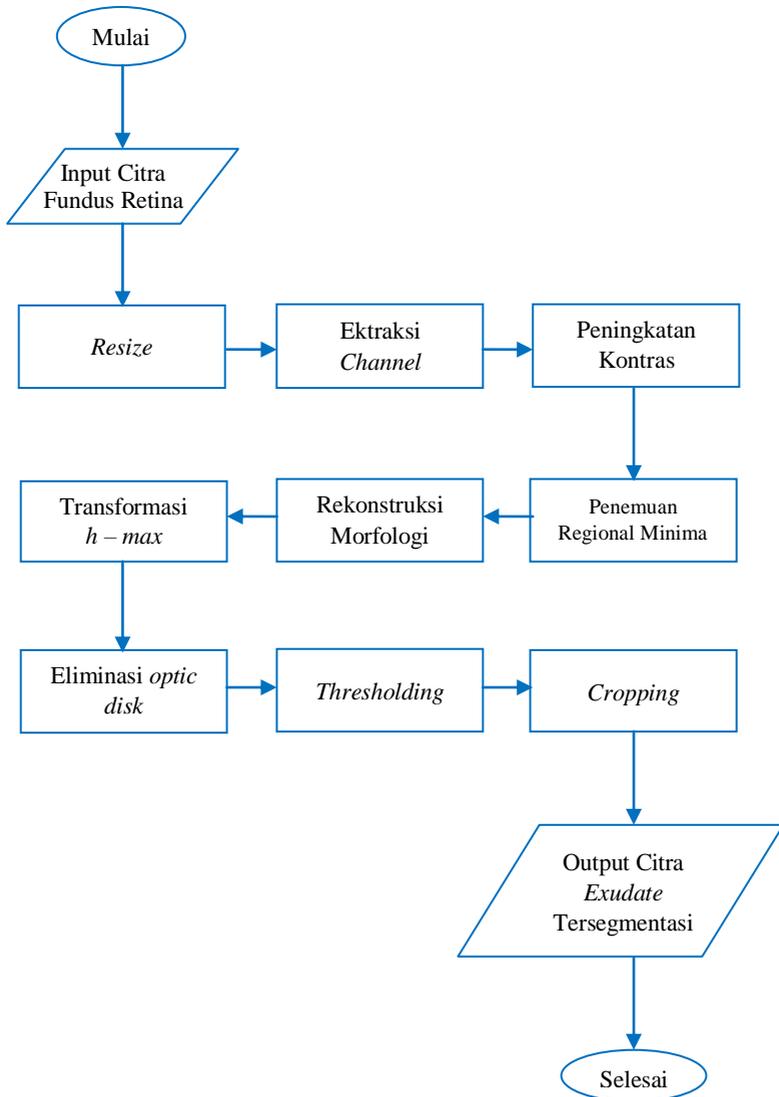
Gambar 4. 1: Diagram alir proses eliminasi *optic disk*

4.1.1.2 Segmentasi Exudate

Proses segmentasi dilakukan untuk mempertegas daerah *hard exudate* pada citra fundus retina sehingga dapat dideskripsikan dan dikenali. Citra fundus retina yang telah melalui proses eliminasi *optic disk* disegmentasi menggunakan metode *thresholding* Otsu untuk memisahkan daerah *exudate* dan *non-exudate*. Proses segmentasi *exudate* ditunjukkan ada Gambar 4.2.

Tahap-tahap yang dilakukan pada proses segmentasi *exudate* yaitu:

1. Mengubah citra RGB yang telah *dirresize* ke dalam citra *grayscale*
2. Mengekstraksi citra RGB yang telah *dirresize* ke dalam *Green Channel*.
3. Melakukan operasi morfologi peningkatan kontras untuk meningkatkan kontras
4. Menemukan regional minima untuk menandai area-area citra yang memiliki nilai intensitas yang rendah.
5. Mengubah citra hasil tahap sebelumnya menjadi citra *gray*
6. Melakukan rekonstruksi morfologi dari hasil penemuan regional minima
7. Melakukan transformasi h-maxima untuk menghilangkan area yang memiliki nilai intensitas yang rendah
8. Mensubstraksi citra hasil transformasi h-maxima dengan wilayah *optic disk* yang telah ditemukan pada proses eliminasi *optic disk*.
9. Melakukan *thresholding* untuk menentukan wilayah *exudate* dan *non-exudate*
10. Melakukan *cropping* citra hasil *thresholding* untuk mengembalikan nilai asli dari *exudate* yang telah ditandai.



Gambar 4. 2: Diagram alir proses segmentasi *exudates*

4.1.2 Ekstraksi Ciri

Informasi yang diekstrak dari citra *exudate* yang telah tersegmentasi merupakan ciri statistik orde kedua. Fitur-fitur ini didapatkan dengan menggunakan GLCM (*Gray Level Co-occurrence Matrix*) dan NGTDM (*Neighborhod Gray-tone Difference Matrix*).

4.1.2.1 Pembentukan GLCM

Pada GLCM, ciri statistik *angular second moment (energy)*, *contrast*, *correlation*, *dissimilarity*, *inverse difference moment (homogeneity)*, dan *entropy* dihitung untuk masing-masing orientasi sudut yaitu $0^\circ, 45^\circ, 90^\circ, 135^\circ$ kemudian dicari rata-ratanya. Pada tugas akhir ini, GLCM dibentuk dengan jarak $d = 1$ untuk masing-masing orientasi sudut.

Tahap-tahap yang dilakukan pada pembentukan matriks GLCM ini yaitu:

1. Menentukan matriks area kerja berdasarkan nilai keabuan tertinggi dari citra.
2. Menghitung jumlah kookurensi atau kemunculan bersama setiap pasangan nilai piksel untuk keseluruhan citra.
3. Menambahkan matriks kookurensi yang didapatkan dari tahap sebelumnya dengan matriks transposenya untuk menjadikannya simetris terhadap sumbu diagonal.
4. Melakukan normalisasi matriks yang dihasilkan pada tahap sebelumnya dengan membagi setiap elemen matriknya dengan jumlah seluruh elemen spasial.

Berikut contoh pembentukan matriks GLCM: Misal sebuah citra direpresentasikan oleh matriks A sebagai berikut:

$$A = \begin{bmatrix} 2 & 1 & 3 & 1 \\ 0 & 1 & 0 & 0 \\ 2 & 3 & 2 & 1 \\ 2 & 0 & 1 & 1 \end{bmatrix}$$

Matriks kookurensi akan dihitung dengan nilai $d = 1$ dan $\theta = 0^\circ$.

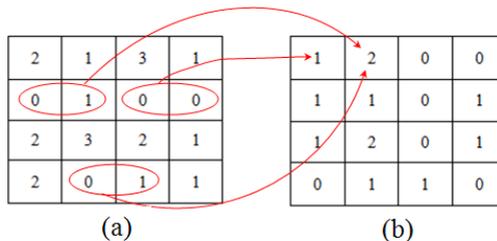
	0	1	2	3
0	0,0	0,1	0,2	0,3
1	1,0	1,1	1,2	1,3
2	2,0	2,1	2,2	2,3
3	3,0	3,1	3,2	3,3

Gambar 4. 3: Area kerja GLCM

Gambar 4.3 menunjukkan area kerja GLCM untuk menentukan posisi frekuensi kemunculan setiap pasangan nilai piksel citra.

Untuk setiap pasangan nilai piksel pada area kerja, dihitung jumlah kemunculannya pada matriks asal, sehingga didapatkan matriks kookurensi I sebagai berikut:

$$I = \begin{bmatrix} 1 & 2 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 2 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$



Gambar 4. 4: (a) Matriks asal, (b) Matriks kookurensi

Gambar 4.4 menunjukkan jumlah kemunculan pasangan nilai piksel dari matriks asal dan penempatan jumlah frekuensi kemunculannya sesuai dengan area kerja sehingga menghasilkan matriks kookurensi.

Matriks kookurensi yang didapat kemudian ditambahkan dengan matriks transposenya sehingga didapatkan matriks M sebagai berikut:

$$M = I + I^T = \begin{bmatrix} 2 & 3 & 1 & 0 \\ 3 & 2 & 2 & 2 \\ 1 & 2 & 0 & 2 \\ 0 & 2 & 2 & 0 \end{bmatrix}$$

Matriks yang telah simetris selanjutnya harus dinormalisasi, elemen-elemennya dinyatakan dengan probabilitas. Nilai elemen untuk masing-masing sel dibagi dengan jumlah seluruh elemen spasial. Sehingga didapatkan matriks GLCM sebagai berikut:

$$GLCM = \begin{bmatrix} 0,087 & 0,13 & 0,043 & 0 \\ 0,13 & 0,087 & 0,087 & 0,087 \\ 0,043 & 0,087 & 0 & 0,043 \\ 0 & 0,087 & 0,087 & 0 \end{bmatrix}$$

Matriks GLCM ini yang digunakan untuk menghitung ciri statistik dari citra *exudate* tersegmentasi.

Contoh perhitungan ciri menggunakan GLCM yaitu :

- *Angular Second Moment* (ASM) dihitung menggunakan Persamaan 2.5 sehingga didapat,

$$\begin{aligned} ASM &= 0,087^2 + 0,13^2 + 0,043^2 + \dots + 0,087^2 + 0^2 \\ &= 0,09233 \end{aligned}$$

- *Contrast* dihitung menggunakan Persamaan 2.6 sehingga didapat,

$$\begin{aligned} Contrast &= 0^2(0,087 + \dots + 0) + 1^2(0,13 + \dots + 0,087) \\ &\quad + 2^2(0,043 + \dots + 0,087) + 3^2(0 + 0) \\ &= 1,604 \end{aligned}$$

- *Correlation* dihitung menggunakan Persamaan 2.7. Nilai $\mu_x, \mu_y, \sigma_x \sigma_y$ dihitung terlebih dahulu dan didapat nilai $\mu_x = \mu_y = 0,2495$ dan $\sigma_x = 0,10277$ dan $\sigma_y = 0,10873$ sehingga didapat,

$$\begin{aligned} Correlation &= \frac{1}{(0,10277)(0,10873)} (0,087 + 0,087 \\ &\quad + \dots 0,087 + 0 - 16(0,2495)^2) \end{aligned}$$

$$= 66,311$$

- *Dissimilarity* dapat dihitung menggunakan Persamaan 2.8 sehingga didapat,

$$\text{Dissimilarity} = 0(0,087) + 1(0,13) + \dots + 3(0)$$

$$= 1,084$$
- *Inverse Difference Moment (IDM)* dihitung menggunakan Persamaan 2.9 sehingga didapat,

$$\text{IDM} = 0,087 + \frac{0,13}{2} + \frac{0,043}{5} + \dots + \frac{0,087}{2} + 0$$

$$= 0.508$$
- *Entropy* dihitung menggunakan Persamaan 2.10 sehingga didapat,

$$\text{Entropy} = -(0,087(\log 0,087) + 0,13(\log 0,13) + \dots$$

$$+ 0,087(\log 0,087) + 0(\log(0 + \epsilon)))$$

$$= 0.508$$

4.1.2.2 Pembentukan Entry dan Probabilitas NGTDM

Perhitungan ciri statistik orde kedua dengan pada NGTDM ini menggunakan nilai kegelapan (*gray-tone*) dari persekitaran sebuah titik atau piksel pada citra dengan mempertimbangkan probabilitas kemunculan nilai *graytone* tersebut.

Tahap-tahap yang dilakukan pada pembentukan matriks NGTDM ini yaitu:

1. Menentukan jarak atau ukuran persekitaran d .
2. Untuk setiap nilai *graytone* ke- i dihitung rata-rata nilai *gray-tone* dari persekitarannya menggunakan Persamaan (2.11).
3. Menghitung jumlah kemunculan dari masing-masing nilai *graytone*.
4. Menghitung probabilitas kemunculan setiap nilai *graytone*.
5. Untuk setiap nilai *graytone* ke- i dihitung *entry* $S(i)$ menggunakan Persamaan (2.12)

$$S(1) = \left|1 - \frac{12}{8}\right| + \left|1 - \frac{15}{8}\right| = 1,375$$

sehingga untuk setiap nilai *graytone* dari citra *input* didapat $S(i)$ seperti ditunjukkan pada Gambar 4.5(b). Nilai probabilitas dihitung menggunakan Persamaan 2.14 sehingga didapatkan,

$$P = \begin{bmatrix} 0.011342 \\ 0.015123 \\ 0.0075614 \\ 0.0056711 \\ 0.0037807 \\ 0.0037807 \end{bmatrix}$$

Contoh perhitungan ciri menggunakan NGTDM yaitu :

- *Coarseness* dihitung menggunakan Persamaan 2.13. sehingga didapatkan,

$$f_{cos} = \frac{1}{\varepsilon + (0.011342)(5,625) + \dots + (0.0037807)(3,5)}$$

$$= 8,481$$

- *Contrast* dapat dihitung menggunakan Persamaan 2.15, dan diperoleh nilai $Ng = 6$ dan $n^2 = 529$ sehingga didapatkan,

$$f_{con} = \frac{1}{6 * 5} \left((0.011342^2) 0 + (0.011342)(0.015123) 1 + \dots \right. \\ \left. + (0.0037807^2) 0 \right) \left(\frac{1}{529} (5,625 + \dots + 3,5) \right)$$

$$= 0,024509$$

- *Busyness* dapat dihitung menggunakan Persamaan 2.16, sehingga didapat,

$$f_{bus} = \left(\frac{(0.011342)(5,625) + \dots + (0.0037807)(3,5)}{((0(0.011342) - 0(0.011342)) + \dots + (5(0.0037807) - 5(0.0037807)))} \right)$$

$$= 0.58844$$

- *Complexity* dapat dihitung menggunakan Persamaan 2.17, sehingga didapat,

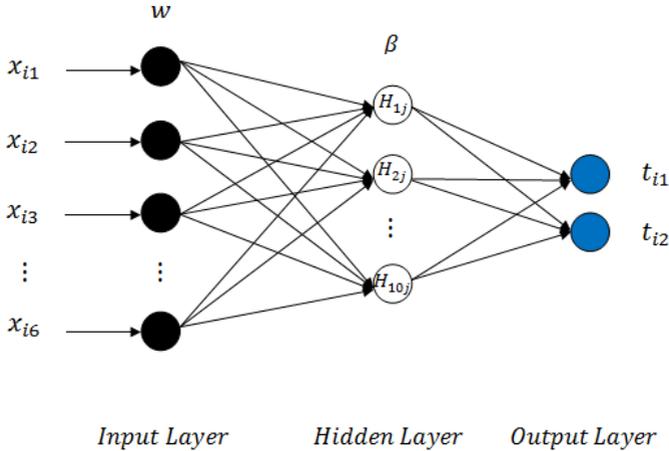
$$\begin{aligned}
f_{com} &= \left(\frac{1}{529}\right) \left(\frac{0}{0.011342 + 0.011342}\right) ((0.011342)(5,625) \\
&+ (0.011342)(5,625)) \\
&+ \left(\frac{1}{0.011342 + 0.015123}\right) ((0.011342)(5,625) \\
&+ (0.015123)(1,375)) + \dots \\
&+ \left(\frac{0}{0.0037807 + 0.0037807}\right) ((0.0037807)(2,75) \\
&+ (0.0037807)(3,5)) \\
&= 0.37869
\end{aligned}$$

- *Texture Strength* dapat dihitung menggunakan Persamaan 2.18, sehingga didapat,

$$\begin{aligned}
f_{str} &= \frac{0 + (0.011342 + 0.015123) + \dots + 0}{\varepsilon + 5,625 + \dots + 3,5} \\
&= 0.22443
\end{aligned}$$

4.1.3 Desain Arsitektur Extreme Learning Machine

Desain arsitektur *Extreme Learning Machine* pada Tugas Akhir ini ditunjukkan pada Gambar 4.6. *Input* untuk jaringan berupa karakteristik dari 105 citra yang berbeda yang masing-masing berisi ciri statistik (*angular second moment (energy)*, *contrast*, *correlation*, *dissimilarity*, *inverse difference moment (homogeneity)*, *entropy*) karakteristik *exudate* yang merupakan hasil dari ekstraksi ciri. Jumlah node pada *layer* pertama sama dengan jumlah ciri statistik yang menjadi *input* yaitu 6 node. Jumlah ini menghasilkan hasil paling optimal pada proses klasifikasi. *Layer* kedua merupakan *hidden layer* yang menghasilkan hasil optimal dengan 10 node. Hasil keluaran dari *hidden layer* ini akan menjadi masukan untuk *layer output* yang terdiri dari dua buah node.



Gambar 4. 6: Desain Arsitektur Extreme Learning Machine yang digunakan pada Tugas Akhir ini

4.1.4 Pemodelan Extreme Learning Machine

Extreme Learning Machine memiliki model matematis yang berbeda dari jaringan syaraf tiruan *feedforward*. Model matematis dari ELM lebih sederhana dan efektif [18]. Dengan 105 jumlah pasangan *input* dan target *output* yang berbeda (x_i, t_i) , dengan $x_i = [x_{i1}, x_{i2}, \dots, x_{iN}]^T$, N adalah jumlah ciri statistik yang didapatkan dari hasil ekstraksi ciri dan $t_i = [t_{i1}, t_{i2}]^T$ dengan jumlah *hidden* sebanyak 10 dan fungsi aktivasi $g(x)$ maka pada Tugas Akhir ini berdasarkan Persamaan (2.19) didapatkan model matematis ELM sebagai berikut:

$$\sum_{i=1}^{10} \beta_i g_i(x_j) = \sum_{i=1}^{10} \beta_i g(w_i \cdot x_j + b_i) = o_j, \quad j = 1, 2, \dots, 105$$

dengan,

- $w_i = [w_{i1}, w_{i2}, \dots, w_{i10}]^T$ merupakan vektor bobot yang menghubungkan *hidden node* ke- i dan *input node*

- b. $\beta_i = [\beta_{i1}, \beta_{i2}]^T$ merupakan vektor bobot yang menghubungkan *hidden node* ke- i dan *output node*
- c. b_i merupakan bias dari *hidden node* ke- i
- d. $w_i \cdot x_j$ merupakan *inner product* dari w_i dan x_j

Konstruksi model ELM pada Tugas Akhir ini mengacu pada desain arsitektur yang telah dibuat. Model berupa persamaan umum fungsi transfer antara *layer* satu dengan *layer* lainnya. Berikut penjabaran model pada setiap *layer* arsitektur jaringan.

4.1.4.1 Input Layer

Layer ini merupakan *layer* pertama pada jaringan. Jumlah node pada *layer* sesuai dengan jumlah ciri statistik hasil dari ekstraksi ciri citra fundus retina. Satu ciri statistik mewakili satu node.

Input layer dan *hidden layer* dihubungkan oleh vektor bobot w yang nilainya ditentukan secara acak. Bias yang terhubung dengan node-node pada *hidden layer* juga ditentukan secara acak. Fungsi aktivasi yang digunakan pada *layer* ini adalah *sigmoid bipolar*, ditunjukkan oleh persamaan berikut:

$$f(x) = \frac{2}{1+e^{-x}} - 1 \quad (4.1)$$

dengan x merupakan bilangan *real* dan e adalah konstanta *euler* yang nilainya sekitar 2,7183. Fungsi aktivasi ini memiliki *range* antara -1 sampai 1 . Fungsi aktivasi ini mendapatkan hasil paling optimal pada proses klasifikasi.

Berdasarkan Persamaan (4.2) dapat dirumuskan suatu fungsi umum sebagai berikut:

$$\begin{aligned} H_{i,j} &= g(w_i \cdot x_j + b_i) \\ &= \frac{2}{1 + e^{-(w_i \cdot x_j + b_i)}} - 1 \end{aligned} \quad (4.2)$$

Keterangan variabel dan indeks pada persamaan-persamaan di atas adalah sebagai berikut:

H : matriks output pada *hidden layer*

- w : vektor bobot yang menghubungkan *hidden node* dan *input node*
 x : matriks *input*
 b : bias yang terhubung dengan *hidden node*
 i : indeks jumlah node pada *hidden layer*
 j : indeks jumlah citra input
 $w \cdot x$: *inner product* dari w dan x

4.1.4.2 Hidden Layer

Layer ini merupakan *layer* kedua dari jaringan dan terdiri dari 10 buah node. *Hidden layer* dan *output layer* dihubungkan oleh vektor bobot β . Berdasarkan Persamaan (2.21) dengan $N = 105$ dan $\tilde{N} = 10$ dapat dituliskan suatu persamaan berikut:

$$H\beta = T \quad (4.3)$$

dengan,

$$H = \begin{bmatrix} g(w_1 \cdot x_1 + b_1) & \cdots & g(w_{10} \cdot x_1 + b_{10}) \\ \vdots & \ddots & \vdots \\ g(w_1 \cdot x_{105} + b_1) & \cdots & g(w_{10} \cdot x_{105} + b_{10}) \end{bmatrix}$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_{10}^T \end{bmatrix} \text{ dan } T = \begin{bmatrix} T_1^T \\ \vdots \\ T_{105}^T \end{bmatrix}$$

Keterangan variabel dan indeks pada persamaan di atas adalah sebagai berikut:

- H : matriks output pada *hidden layer*
 w : vektor bobot yang menghubungkan *hidden node* dan *input node*
 x : matriks *input*
 b : bias yang terhubung dengan *hidden node*
 $w \cdot x$: *inner product* dari w dan x

β : vektor bobot yang menghubungkan *hidden node* dan *output node*

T : matriks target

Bobot *output* yang berhubungan dengan *hidden layer* dapat ditentukan dari persamaan berikut:

$$\beta = H^{\dagger}T \quad (4.4)$$

dengan H^{\dagger} merupakan *Moore-Penrose generalized inverse* dari H .

4.1.4.3 Output Layer

Setiap node yang ada di hidden layer dihubungkan dengan *output layer* melalui vektor bobot β . Jumlah node *output* disesuaikan dengan jumlah kelas untuk klasifikasi. *Output layer* pada Tugas Akhir ini terdiri dari 2 buah node. Hasil dari keluaran pada *output layer* ini mewakili tingkat keparahan NPDR. Nilai *output* dari layer ini dihitung menggunakan Persamaan (4.3).

4.1.5 Klasifikasi dengan Extreme Learning Machine

Dalam Tugas Akhir ini *Non-Proliferative Diabetic Retinopathy* diklasifikasikan kedalam 2 kelas yaitu *moderate* (sedang) dan *severe* (parah). Klasifikasi dilakukan berdasarkan karakteristik *hard exudate* yang ditemukan pada retina. Proses klasifikasi ini terdiri dari dua tahap yaitu proses *training* dan *testing*. Proses *training* memerlukan *input* berupa citra fundus retina dan tingkat keparahannya. Proses ini menghasilkan bobot w dan bias b yang ditentukan secara acak serta bobot β yang dihitung dengan menggunakan Persamaan (4.4). Bobot dan bias yang dihasilkan pada proses ini digunakan untuk klasifikasi NPDR pada proses *testing*. Proses *testing* memerlukan *input* berupa citra fundus retina dan memberikan *output* berupa tingkat keparahan dari citra *input*. Berikut penjabaran proses klasifikasi pada tiap tahap.

4.1.5.1 Proses *Training*

Pada Tugas Akhir ini data *training* yang digunakan terdiri dari 105 pasangan ciri statistik karakteristik *exudate* dari citra fundus retina yang teridentifikasi *Diabetic Retinopathy* serta tingkat keparahannya. Ciri statistik *exudate* diwakili oleh matriks *input* x dan tingkat keparahannya diwakili oleh matriks target t yang terdiri dari 2 elemen dengan nilai 1 atau -1 . Representasi tingkat keparahan NPDR dalam matriks t dapat dilihat pada Tabel 4.1.

Tabel 4. 1. Representasi tingkat keparahan NPDR dalam Matriks t

No	Tingkat keparahan	Representasi dalam Matriks t
1	<i>Moderate</i>	[1, 1]
2	<i>Severe</i>	[-1, 1]

Tahap-tahap yang dilakukan pada proses *training* yaitu:

1. Menentukan bobot w dan bias b secara acak
2. Untuk setiap citra input lakukan eliminasi *optic disk* dari citra fundus retina
3. Melakukan segmentasi *exudate*
4. Mengekstraksi ciri citra *exudate* tersegmentasi menggunakan GLCM dan NGTDM
5. Setelah semua citra diekstraksi ciri, dihitung matriks H menggunakan Persamaan (4.2)
6. Menghitung bobot β menggunakan Persamaan (2.24)

4.1.5.2 Proses *Testing*

Pada proses ini citra fundus retina harus melalui tahap pra-proses terlebih dahulu untuk menghilangkan *optic disk* dan memisahkan *exudate* dari *background* (segmentasi). *Exudate* yang telah tersegmentasi diekstraksi dengan menggunakan metode GLCM dan NGTDM agar diperoleh ciri statistiknya. Ciri statistik ini yang menjadi masukkan pada ELM.

Tahap-tahap yang dilakukan pada proses *testing* yaitu:

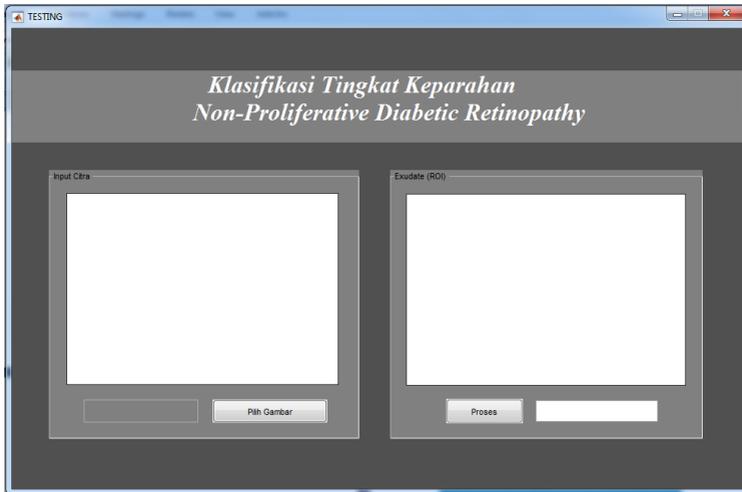
1. Mengeliminasi *optic disk* dari citra *input*.
2. Melakukan segmentasi *exudate*
3. Mengekstraksi ciri citra *exudate* tersegmentasi menggunakan GLCM dan NGTDM
4. Mengambil nilai bobot w dan bias b pada proses *training*.
5. Menghitung matriks H menggunakan Persamaan (4.2)
6. Menghitung bobot β menggunakan Persamaan (2.23)
7. Menghitung nilai *output*

4.2 Implementasi

Perancangan program yang telah dibangun selanjutnya diimplementasikan dengan menggunakan MATLAB R2015a. Pembahasan dalam implementasi sistem meliputi implementasi antarmuka (*interface*) sistem, implementasi tahap praproses, implementasi eliminasi *optic disk*, implementasi segmentasi *exudate*, implementasi pembentukan GLCM, implementasi perhitungan ciri statistik menggunakan GLCM, implementasi pembentukan *entry* dan probabilitas NGTDM, implementasi perhitungan ciri statistik menggunakan NGTDM dan terakhir implementasi proses *training* dan *testing* pada klasifikasi menggunakan metode *Extreme Learnin Machine*.

4.2.1 Implementasi User Interface

Desain *user interface* klasifikasi tingkat keparahan NPDR ini dapat dilihat pada Gambar 4.7. Pada desain *user interface* terdapat menu Pilih gambar dan Proses. Citra yang digunakan sebagai masukan dalam Tugas Akhir ini adalah file yang berekstensi .tif yang diambil langsung dari direktori komputer.



Gambar 4. 7: Desain *user interface* klasifikasi tingkat keparahan NPDR

4.2.2 Implementasi praproses citra

Citra fundus retina yang menjadi data *input* memiliki ukuran yang besar dan berbeda-beda. Oleh karena itu pada tahap praproses citra *resize* menjadi ukuran 384x576 agar seluruh citra yang diolah memiliki ukuran yang seragam. Kode program tahap praproses ini diimplementasikan menjadi sebuah program berikut:

```
im = imresize(im, [384 576]);
```

4.2.2.1 Implementasi *Eliminasi Optic Disk*

Setelah melalui proses *resize*, dilakukan ekstraksi *channel* pada citra. Kemudian dilakukan peregangkan kontras dan *noise filtering*. Setelah itu dicari *centroid* dari citra untuk menentukan posisi lingkaran *optic disk*. Kode program proses eliminasi *optic disk* diimplementasikan menjadi sebuah program berikut:

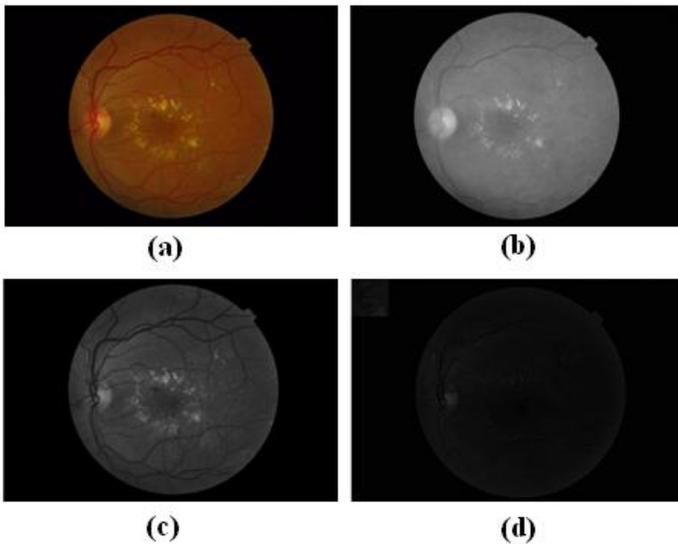
```

val=ceknilaiv(img);
if val<0.29
    im=img(:,:,1); %red
else
    im=img(:,:,2); %green
end
CL=adapthisteq(im);
hasil=Contrasstreching(CL,10,120,0,150);
im=medfilt2(hasil,[15 15]);
max_v=max(max(im));
final=ismember(im,max_v);
L=bwlabel(final,4);
s=regionprops(L,'Area');
area_values=[s.Area];
idx=find(area_values>0);
bw2=ismember(L,idx);
bw3=bwdist(bw2,'euclidean')<=11;
L2=bwlabel(bw3,4);
s=regionprops(L2,'Area');
area_values=[s.Area];
idx=find(area_values==max(area_values));
bw4=ismember(L2,idx);
L3=bwlabel(bw4,4);
s=regionprops(L3,'Area','centroid');
r=30;
circle=MidpointCircle(L3,r,s.Centroid(2),s.Centroid(1),1);
circlefilled=imfill(circle);
finalOD=~circlefilled;

```

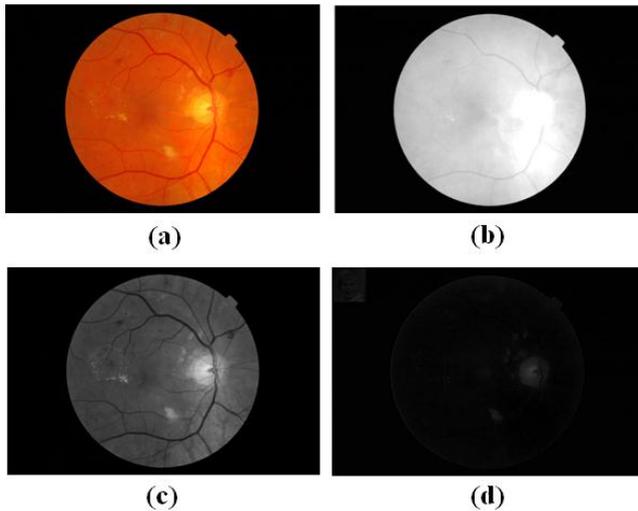
Citra fundus retina yang digunakan sebagai *input* memiliki tingkat kecerahan yang berbeda sehingga ekstraksi *channel* dilakukan ke dalam *green channel* atau *red channel* saja. Karena jika diubah ke dalam *blue channel* citra cenderung berwarna gelap sehingga sulit untuk diteliti. Penentuan *channel* untuk setiap citra dilakukan berdasarkan rata-rata nilai histogramnya. Citra yang memiliki tingkat

kecerahan rendah diubah ke dalam *red channel*. Hal ini dilakukan karena *optic disk* merupakan daerah retina yang berwarna terang sehingga untuk mempertegas wilayah *optic disk* citra diubah ke *red channel*. Namun untuk citra yang memiliki tingkat kecerahan yang tinggi harus diubah ke dalam *green channel*, karena jika diubah ke dalam *red channel* maka wilayah *optic disk* akan tertelan oleh wilayah di sekitarnya yang memiliki tingkat kecerahan yang lebih tinggi.



Gambar 4. 8: (a) Citra asli, (b) *Red Channel*, (c) *Green Channel*, (d) *Blue Channel*

Gambar 4.8(a) menunjukkan citra fundus retina yang memiliki tingkat kecerahan rendah. Sedangkan Gambar 4.8(b), 4.8(c) dan 4.8(d) masing-masing menunjukkan hasil ekstraksi *red channel*, *green channel* dan *blue channel* dari citra. Lingkaran *optic disk* lebih terlihat jelas pada citra *red*.



Gambar 4. 9: (a) Citra asli, (b) *Red Channel*, (c) *Green Channel*, (d) *Blue Channel*

Gambar 4.9(a) menunjukkan citra fundus retina yang memiliki tingkat kecerahan tinggi. Gambar 4.9(a), 4.9(b) dan 4.9(c) masing-masing menunjukkan hasil ekstraksi *red channel*, *green channel* dan *blue channel* dari citra. Karena citra asli memiliki tingkat kecerahan yang tinggi maka pada *red channel* wilayah *optic disk* tertelan oleh wilayah di sekitarnya yang juga memiliki nilai intensitas yang tinggi. Sedangkan pada *green channel* wilayah *optic disk* lebih terlihat jelas

4.2.2.2 Implementasi Segmentasi Exudate

Tahap awal dari proses ini adalah mengubah citra berwarna menjadi citra *grayscale* dan mengekstraksi citra berwarna ke dalam bentuk *green channel*. Ekstraksi channel hanya dilakukan untuk *green channel* karena exudate terlihat lebih terlihat jelas di *green channel* seperti ditunjukkan pada

gambar 4.8(c) dan 4.9(c). Kode program proses segmentasi *exudate* ini diimplementasikan menjadi sebuah program berikut:

```
gray=rgb2gray(citral);
green = citral(:, :, 2);
```

Selanjutnya dilakukan operasi morfologi berupa peningkatan kontras citra pada *green channel* dengan melakukan *top-hat filtering* yaitu melakukan operasi morfologi *opening* diikuti dengan melakukan substraksi citra awal dengan citra hasil *opening*. *Top-hat filtering* mampu memperbaiki iluminasi yang tidak merata pada citra. *Top-hat filtering* membuat *background* citra dan daerah *non exudate* yang nilai intensitasnya rendah menjadi lebih gelap sehingga daerah *exudate* terlihat lebih menonjol. Setelah itu, dilakukan *image adjustment* untuk memperjelas tampilan sekaligus meningkatkan kontras citra. Kode program proses peningkatan kontras menggunakan operasi morfologi diimplementasikan menjadi sebuah program berikut:

```
se =(strel('disk',10));
I2 = imtophat(green,se);
I3 = imadjust(I2);
```

Citra yang telah ditingkatkan kontrasnya kemudian dicari *regional minima*. Penemuan *regional minima* ini bertujuan untuk menandai piksel-piksel yang memiliki nilai intensitas rendah. Hasil dari proses ini berupa citra biner sehingga harus diubah ke betuk *gray* agar bisa digunakan pada rekonstruksi operasi morfologi. Kode program proses penemuan *regional minima* dari citra dan proses rekonstruksi citra diimplementasikan menjadi sebuah program berikut:

```
I4 = imregionalmin(I3);
grayImage = 255 * uint8(I4);
I5 = imreconstruct(grayImage, I3);
```

Karakteristik *exudate* adalah area yang memiliki intensitas yang tinggi. Oleh karena itu, setelah dilakukan rekonstruksi morfologi selanjutnya dilakukan transformasi *h-maxima* untuk menghilangkan area yang memiliki nilai intensitas rendah. Citra yang dihasilkan dari proses ini masih memiliki wilayah *optic disk* karena *optic disk* memiliki nilai intensitas yang tinggi. Oleh karena itu, dilakukan penghilangan lingkaran *optic disk* pada citra hasil transformasi *h-maxima*. Kode program proses transformasi *h-maxima* dan penghilangan wilayah *optic disk* diimplementasikan menjadi sebuah program berikut:

```
b = imhmax(I5, 50);
I7=immultiply(b, OD);
```

Proses *thresholding* dilakukan setelah wilayah *optic disk* dihilangkan. Tujuannya adalah untuk menghilangkan area-area yang nilainya kurang dari nilai *threshold* yang diberikan pada metode otsu. Citra di *threshold* menggunakan tiga nilai ambang dan dicari nilai intensitas yang terbesar dan terkecil. Area dengan nilai intensitas yang rendah dihilangkan sehingga menyisakan area-area yang memiliki nilai intensitas yang tinggi. Kemudian area-area yang tersisa ini ditandai sebagai *exudate* dalam citra biner. Kode program proses *thresholding* diimplementasikan menjadi sebuah program berikut:

```

Nvals=3;
thresh = multithresh(I9,Nvals);
valuesMax = [thresh max(I9(:))];
[citra2,index] = imquantize(I9, thresh,
valuesMax);
valuesMin = [thresh min(citra2(:))];
citra3 = valuesMin(index);
citra4 = I9-citra3;
cit=immultiply(gray,citra4);

```

Terakhir, dilakukan *cropping* untuk mengembalikan nilai asli dari *exudate* yang telah ditandai. Kode program proses *cropping* diimplementasikan menjadi sebuah program berikut:

```

OUT=zeros(m,n);
for i=1:m
    for j=1:n
        if(cit(i,j)==255)
            OUT(i,j)=gray(i,j);
        end
    end
end
end

```

4.2.3 Implementasi Proses Ekstraksi Ciri dengan GLCM dan NGTDM

4.2.3.1 Implementasi Pembentukan GLCM

Matriks GLCM yang digunakan untuk menghitung ciri statistik citra pada Tugas Akhir ini dibentuk dengan orientasi sudut yaitu $0^\circ, 45^\circ, 90^\circ, 135^\circ$ dengan jarak $d = 1$ untuk masing-masing orientasi sudut kemudian dicari rata-ratanya. Kode program proses pembentukan GLCM ini diimplementasikan menjadi sebuah program berikut:

```

if deg==0
    GLCM2 = graycomatrix(citral,'Offset',[0
pixdist], 'symmetric', true);
elseif deg==45
    GLCM2 = graycomatrix(citral,'Offset',[-
pixdist pixdist], 'symmetric', true);
elseif deg==90
    GLCM2 = graycomatrix(citral,'Offset',[-
pixdist 0], 'symmetric', true);
else
    GLCM2 = graycomatrix(citral,'Offset',[-
pixdist -pixdist], 'symmetric', true);
end

```

4.2.3.2 Implementasi Perhitungan Ciri Statistik

Menggunakan GLCM

Perhitungan ciri statistik dari citra dengan GLCM dilakukan menggunakan matriks yang telah terbentuk pada tahap sebelumnya. Ciri statistik *angular second moment (energy)* dihitung menggunakan Persamaan (2.5), *contrast* dihitung menggunakan Persamaan (2.6), *correlation* dihitung menggunakan Persamaan (2.7), *dissimilarity* dihitung menggunakan Persamaan (2.8), *inverse difference moment (homogeneity)* dihitung menggunakan Persamaan (2.9), dan *entropy* dihitung menggunakan Persamaan (2.10). Kode program proses perhitungan ciri statistik dengan GLCM ini diimplementasikan menjadi sebuah program berikut:

```

mul_contr = abs(i_matrix - j_matrix).^2;
mul_dissi = abs(i_matrix - j_matrix);
for k = 1:size_glcm_3
    contrs(k)=sum(sum(mul_contr.*glcm(:,:,k)
    ));
    dissi(k)=sum(sum(mul_dissi.*glcm(:,:,k)
    ));
    energ(k)=sum(sum(glcm(:,:,k).^2));
    entro(k)=-sum(sum((glcm(:,:,k).*
    log(glcm(:,:,k) + eps))));
    homom(k)=sum(sum((glcm(:,:,k)./(1 +
    mul_dissi))));
    u_x(k)=sum(sum(i_matrix.*glcm(:,:,k)));
    u_y(k)=sum(sum(j_matrix.*glcm(:,:,k)));
    s_x(k)=(sum(sum(((i_matrix -
    u_x(k)).^2).* glcm(:,:,k))))^0.5;
    s_y(k)=(sum(sum(((j_matrix -
    u_y(k)).^2).* glcm(:,:,k))))^0.5;
    corm(k)=sum(sum(((i_matrix - u_x(k)).*
    (j_matrix - u_y(k)).*glcm(:,:,k))));
    corrm(k) = corm(k) / (s_x(k)*s_y(k));
end

```

4.2.3.3 Implementasi Pembentukan Entry dan Probabilitas NGTDM

Entry dan probabilitas ini digunakan untuk menghitung ciri statistik dari citra *exudate* tersegmentasi. Kode program implementasi untuk proses pembentukan *entry* dan probabilitas NGTDM ini diimplementasikan menjadi sebuah program berikut:

```

for i=1:graylevels
    awl=0;
    for k=(1+d):(m-d)
        for l=(1+d):(n-d)
            if (im(k,l)==(i-1))
                awl=awl+1;
            end
        end
    end
    N(i)=awl;
end
for i=1:graylevels
    hsl=0;
    [k,l]=find(citra==(i-1));
    ki=size(k,l);
    for x=1:ki
        if (k(x)>1 && k(x)<m && l(x)>1 &&
            l(x)<n)
            a=Akl(citra,W,k(x),l(x),d);
            hsl=hsl+abs((i-1)-a);
        end
    end
    S(i)=hsl;
end

Nn=sum(N)-2*d;
R=1/(Nn^2);
P=N*R;

```

4.2.3.4 Implementasi Perhitungan ciri statistik menggunakan NGTDM

Ciri statistik yang diekstrak menggunakan NGTDM antara lain: *coarsness*, *contrasts*, *bussyness*, *complexity* dan *texture strength*. *Coarsness* dihitung menggunakan Persamaan (2.13). Kode program perhitungan ciri statistik *coarsness* diimplementasikan menjadi sebuah program berikut:

```
Coars = 1 / (eps + (sum(P.*S)));
```

Contrast dihitung menggunakan Persamaan (2.15). Kode program perhitungan ciri statistik *contrasts* diimplementasikan menjadi sebuah program berikut:

```
A=0;
for i=1:graylevels
    for j=1:graylevels
        A=A+(P(i)*P(j)*(((i-1)-(j-1))^2));
    end
end
B=sum(S);
Contr=(1/(Ng*(Ng-1)))*A*R*B;
```

Bussyness dihitung menggunakan Persamaan (2.16). Kode program perhitungan ciri statistik *bussyness* diimplementasikan menjadi sebuah program berikut:

```
C=0;
for i=1:graylevels
    for j=1:graylevels
        if(P(i)==0 && P(j)==0)
            continue
        else
            C=C+abs((i-1)*P(i)-(j-1)*P(j));
        end
    end
end
Buss=sum(P.*S)/C;
```

Complexity dihitung menggunakan Persamaan (2.17). Kode program perhitungan ciri statistik *complexity* diimplementasikan menjadi sebuah program berikut:

```

D=0;
for i=1:graylevels
    for j=1:graylevels
        if(P(i)==0 && P(j)==0)
            continue
        else
            D=D+((abs((i-1) - (j-1)) /
(P(i)+P(j)))*((P(i)*S(i)) + (P(j)*S(j))));
        end
    end
end
Comp=R*D;

```

Texture strength dihitung menggunakan Persamaan (2.18). Kode program perhitungan ciri statistik *texture strength* diimplementasikan menjadi sebuah program berikut:

```

E=0;
for i=1:graylevels
    for j=1:graylevels
        if(P(i)==0 && P(j)==0)
            continue
        else
            E=E+((P(i)+P(j))*((i-1)-(j-1))^2));
        end
    end
end
Strngth=E/(eps+B);

```

4.2.4 Implementasi Proses Klasifikasi Menggunakan Extreme Learning Machine

4.2.4.1 Implementasi Proses training

Pada proses *training*, setelah seluruh data *training* dimasukkan ke dalam sistem klasifikasi, kemudian sistem menentukan nilai bobot w dan bias b . Nilai w (bobot yang menghubungkan *input layer* dan *hidden layer*) dan b

merupakan bilangan antara -1 sampai 1 yang ditentukan secara acak. Kode program penentuan nilai w dan b diimplementasikan menjadi sebuah program berikut:

```
W=-1 + (1+1)* rand(hid,n);
b=-1 + (1+1)* rand(hid,1);
```

Setelah bobot dan bias terbentuk selanjutnya dihitung nilai matriks H , yaitu hasil keluaran dari *hidden layer*. Matriks H dihitung menggunakan Persamaan (4.2). Kode program perhitungan nilai matriks H pada proses *training* diimplementasikan menjadi sebuah program berikut:

```
for i=1:m
    for j=1:hid
        w=W(j,:);
        x=X(i,:);
        c=(w*x');
        H(i,j)=fungsi_aktivasi(c+b(j));
    end
end
```

Invers dari Matriks H ini digunakan untuk menghitung nilai bobot β (bobot yang menghubungkan *hidden layer* dan *output layer*) menggunakan Persamaan (2.24). Kode program perhitungan nilai matriks β diimplementasikan menjadi sebuah program berikut:

```
H2=pinv(H);
B = H2*T;
```

4.2.4.2 Implementasi Proses Testing

Data masukan pada proses ini adalah ciri statistik dari citra fundus retina yang telah melalui tahap eliminasi *optic disk* dan segmentasi *exudate*. Bobot w dan bias b yang digunakan pada proses ini sama dengan bobot dan bias yang

digunakan pada saat *training*. Selanjutnya, dihitung nilai matriks H menggunakan persamaan (4.2). Kode program perhitungan nilai matriks H pada proses *testing* diimplementasikan menjadi sebuah program berikut:

```

for i=1:m
    for j=1:hid
        w=W(i, :);
        c=(w*X');
        g(i)=fungsi_aktivasi(c+b(i));
    end
end

```

Setelah matriks H didapat selanjutnya dihitung nilai *output* menggunakan persamaan (2.19). Karena jumlah node pada *hidden layer* berjumlah 10 maka Persamaan (2.19) dapat dituliskan sebagai berikut:

$$o = \sum_{i=1}^{10} \beta_i g(w_i \cdot x + b_i)$$

Matriks x diperoleh dari hasil ekstraksi ciri citra. Banyaknya elemen matriks x tergantung dari jumlah ciri statistik yang dihitung pada proses ekstraksi ciri. Matriks *output* o terdiri dari 2 elemen yang digunakan untuk menentukan tingkat keparahan dari citra *input*. Untuk setiap $o_i \in o$, jika $o_i < 0$ maka $o_i = -1$ dan jika $o_i > 0$ maka $o_i = 1$. Hasil klasifikasi ini ditentukan dari Tabel 4.1 dengan melihat label tingkat keparahan yang sesuai dengan matriks o . Implementasi program untuk menghitung nilai *output* pada proses *testing* adalah sebagai berikut:

```
o=g*B;
```


BAB V

PENGUJIAN DAN PEMBAHASAN

Bab ini menjelaskan mengenai proses pengujian yang dilakukan terhadap sistem klasifikasi tingkat keparahan NPDR dengan menggunakan *Extreme Learning Machine*. Hasil pengujian kemudian dibahas untuk mengetahui kerja sistem secara keseluruhan. Selanjutnya dijelaskan mengenai hasil pengujian terhadap sistem yang telah diimplementasikan pada bab sebelumnya, yaitu pengujian tahap praproses data meliputi eliminasi *optic disk* dan segmentasi *exudate* serta pengujian tahap klasifikasi dengan metode *Extreme Learning Machine* dengan beberapa variasi pengujian untuk mengetahui keakuratan sistem. Bab ini diakhiri dengan pembahasan hasil pengujian yang telah dilakukan.

5.1 Lingkungan Pengujian Sistem

Lingkungan pengujian dari sistem klasifikasi tingkat keparahan NPDR meliputi perangkat keras dan lunak komputer. Detail dari perangkat keras dan lunak yang digunakan dapat dilihat pada Tabel 5.1.

Tabel 5. 1 Spesifikasi perangkat keras dan perangkat lunak

Lingkungan	Spesifikasi	
Perangkat Keras	Processor	Intel® Core™ i3–3110M CPU @ 2.40GHz 2.40 GHz
	RAM	2.00GB
Perangkat Lunak	Sistem Operasi	Windows 7 32-bit
	Tools	Matlab R2015a

5.2 Pengujian Tahap Praproses Data

Pengujian ini dilakukan terhadap proses-proses pada tahap pengolahan citra. Pengujian bertujuan untuk mengetahui bahwa proses-proses pada tahap pengolahan citra sudah benar.

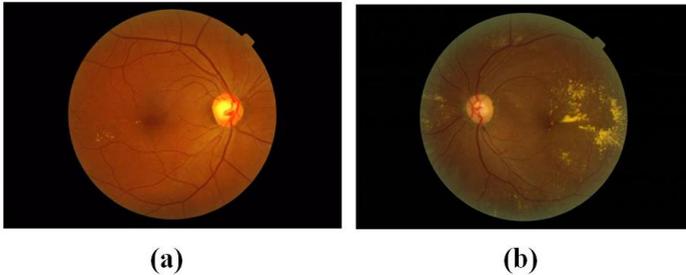
Pada tahap ini terdapat 2 proses yaitu eliminasi *optic disk* dan segmentasi *exudate*.

1. Pengujian eliminasi *optic disk*

Pengujian proses ini bertujuan untuk menghilangkan wilayah *optic disk* sebelum *exudate* di segmentasi. Hasil pengujian proses ini ditunjukkan pada Gambar 5.2

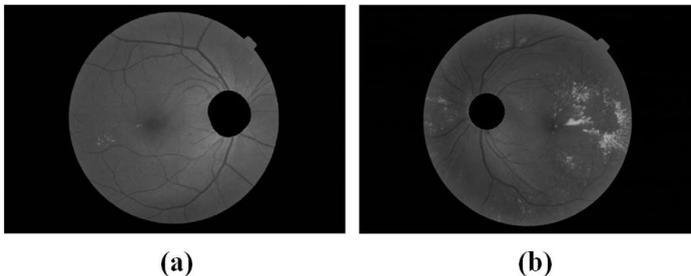
2. Pengujian proses segmentasi *exudate*

Pengujian proses ini bertujuan untuk menemukan wilayah *exudate* pada citra fundus retina. Hasil pengujian proses ini ditunjukkan pada Gambar 5.3.



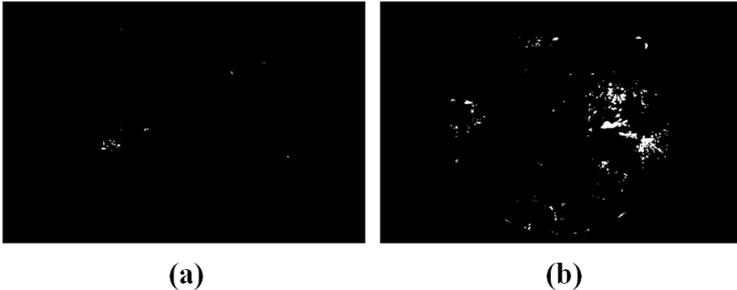
Gambar 5. 1: Citra input (a) *moderate* NPDR, (b) *severe* NPDR

Gambar 5.1(a) dan 5.1(b) masing-masing menunjukkan citra fundus retina dengan *moderate* NPDR dan *severe* NPDR yang belum diproses.



Gambar 5. 2: Pengujian proses eliminasi *optic disk* pada citra: (a) *moderate* NPDR, (b) *severe* NPDR

Gambar 5.2(a) dan 5.2(b) masing-masing menunjukkan hasil eliminasi *optic disk* dari citra fundus retina yang teridentifikasi *moderate* NPDR dan *severe* NPDR.



Gambar 5. 3: Pengujian proses segmentasi *exudate* pada citra: (a) *moderate* NPDR, (b) *severe* NPDR

Gambar 5.3(a) dan 5.3(b) masing-masing menunjukkan hasil segmentasi *exudate* dari citra fundus retina yang teridentifikasi *moderate* NPDR dan *severe* NPDR.

5.3 Pengujian Tahap Klasifikasi Menggunakan *Extreme Learning Machine*

Pengujian sistem dilakukan terhadap data *training* dan data *testing*. Data *training* sebanyak 105 citra retina yang terdiri dari 45 citra teridentifikasi *moderate* NPDR dan 60 citra teridentifikasi *severe* NPDR. Data *testing* sebanyak 34 citra retina yang terdiri dari 17 citra teridentifikasi *moderate* NPDR dan 17 citra teridentifikasi *severe* NPDR. Pengujian proses ini dilakukan untuk menentukan apakah suatu citra fundus retina merupakan retina yang teridentifikasi *moderate* NPDR atau *severe* NPDR

Pada penelitian ini pengujian dilakukan pada kasus 3 kasus yaitu ekstraksi ciri menggunakan GLCM saja, ekstraksi ciri menggunakan NGTDM saja, dan ekstraksi ciri menggunakan GLCM dan NGTDM. Masing-masing kasus diuji menggunakan fungsi aktivasi yang berbeda yaitu

sigmoid biner, *sigmoid bipolar*, dan *softsign*. Penggunaan fungsi aktivasi dan parameter ciri statistik yang digunakan berbeda-beda, artinya untuk setiap kasus model *Extreme Learning Machine* yang digunakan akan berbeda. Tingkat akurasi model dihitung dengan menggunakan Persamaan (3.1).

5.3.1 Pengujian dengan Ekstraksi Ciri Statistik Menggunakan GLCM

Ciri statistik yang digunakan sebagai *input* yaitu *angular second moment (energy)*, *contrast*, *correlation*, *dissimilarity*, *inverse difference moment (homogeneity)* dan *entropy* yang dihitung menggunakan GLCM.

Dengan menggunakan fungsi aktivasi *sigmoid biner*, untuk data *testing* didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 29 dari total 34 citra. Contoh penghitungan akurasi untuk data *testing* adalah sebagai berikut:

$$\text{Akurasi} = \frac{29}{34} \times 100\% = 85,29 \%$$

Untuk data *training* didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 100 dari total 105 citra. Contoh penghitungan akurasi untuk data *training* adalah sebagai berikut:

$$\text{Akurasi} = \frac{100}{105} \times 100\% = 95,24 \%$$

Dengan menggunakan fungsi aktivasi *sigmoid bipolar*, untuk data *testing* didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 29 dari total 34 citra dan untuk data *training* didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 102 dari total 105 citra.

Sedangkan dengan menggunakan fungsi aktivasi *softsign*, untuk data *testing* didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat

sebanyak 29 dari total 34 citra dan untuk data *training* didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 101 dari total 105 citra. Rincian hasil pengujian ini disajikan pada Tabel 5.2 dengan M adalah *moderate* dan S adalah *severe*. Hasil perhitungan akurasi disajikan dalam Tabel 5.3.

Tabel 5. 2 Hasil pengujian dengan ekstraksi ciri GLCM

Fungsi Aktivasi	Citra Input Hasil Output	Data Testing		Data Training	
		M	S	M	S
<i>Sigmoid Biner</i>	M	13	1	43	3
	S	4	16	2	57
<i>Sigmoid Bipolar</i>	M	14	2	44	2
	S	3	15	1	58
<i>Softsign</i>	M	16	4	43	2
	S	1	13	2	58

Tabel 5. 3 Hasil perhitungan akurasi dengan ekstraksi ciri GLCM

Fungsi Aktivasi	Data	Akurasi	Akurasi Total
<i>Sigmoid Biner</i>	<i>Training</i>	95,24 %	90,26 %
	<i>Testing</i>	85,29 %	
<i>Sigmoid Bipolar</i>	<i>Training</i>	97,14 %	91,22 %
	<i>Testing</i>	85,29 %	
<i>Softsign</i>	<i>Training</i>	96,19 %	90,74 %
	<i>Testing</i>	85,29 %	

5.3.2 Pengujian dengan Ekstraksi Ciri Statistik Menggunakan NGTDM

Ciri statistik yang digunakan sebagai *input* yaitu *coarsness*, *contrasts*, *bussyness*, *complexity* dan *texture strength* yang dihitung menggunakan NGTDM. Dengan menggunakan fungsi aktivasi *sigmoid biner*, untuk data *testing* didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 28 dari total 34 citra dan untuk *training* dan didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 100 dari total 105 citra.

Dengan menggunakan fungsi aktivasi *sigmoid bipolar*, untuk data *testing* didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 28 dari total 34 citra dan untuk data *training* dan didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 93 dari total 105 citra.

Sedangkan dengan menggunakan fungsi aktivasi *softsign*, untuk data *testing* didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 29 dari total 34 citra dan untuk data *training* dan didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 101 dari total 105 citra. Rincian hasil pengujian ini disajikan pada Tabel 5.4 dan hasil perhitungan akurasinya disajikan dalam Tabel 5.5.

Tabel 5. 4 Hasil pengujian dengan ekstraksi ciri NGTDM

Fungsi Aktivasi	Citra Input Hasil Output	Data Testing		Data Training	
		M	S	M	S
<i>Sigmoid Biner</i>	M	12	1	42	2
	S	5	16	3	58
<i>Sigmoid Bipolar</i>	M	14	3	43	10
	S	3	14	2	50
<i>Softsign</i>	M	13	1	42	1
	S	4	16	3	59

Tabel 5. 5 Hasil perhitungan akurasi dengan ekstraksi ciri NGTDM

Fungsi Aktivasi	Data	Akurasi	Akurasi Total
<i>Sigmoid Biner</i>	<i>Training</i>	95,24 %	88,79 %
	<i>Testing</i>	82,35 %	
<i>Sigmoid Bipolar</i>	<i>Training</i>	88,57 %	85,46 %
	<i>Testing</i>	82,35 %	
<i>Softsign</i>	<i>Training</i>	96,19 %	90,74 %
	<i>Testing</i>	85,29 %	

5.3.3 Pengujian dengan Ekstraksi Ciri Statistik Menggunakan GLCM dan NGTDM

Ciri statistik yang digunakan sebagai *input* yaitu *angular second moment (energy)*, *contrast*, *correlation*, *dissimilarity*, *inverse difference moment (homogeneity)*, *entropy*, *coarsness*, *bussyness*, *complexity* dan *texture strength* yang dihitung menggunakan GLCM dan NGTDM. Dengan

menggunakan fungsi aktivasi *sigmoid biner*, untuk data *testing* didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 24 dari total 34 citra dan untuk data *training* dan didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 81 dari total 105 citra.

Dengan menggunakan fungsi aktivasi *sigmoid bipolar*, untuk data *testing* didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 27 dari total 34 citra dan data *training* dan didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 96 dari total 105 citra.

Sedangkan dengan menggunakan fungsi aktivasi *softsign*, untuk data *testing* didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 26 dari total 34 citra dan data *training* dan didapatkan jumlah citra retina yang diklasifikasikan ke tingkat keparahan yang tepat sebanyak 100 dari total 105 citra. Rincian hasil pengujian ini disajikan pada Tabel 5.6 dan hasil perhitungan akurasinya disajikan dalam Tabel 5.7.

Tabel 5. 6 Hasil pengujian dengan ekstraksi ciri GLCM dan NGTDM

Fungsi Aktivasi	Citra Input Hasil Output	DataTesting		Data Training	
		M	S	M	S
<i>Sigmoid Biner</i>	M	7	0	22	1
	S	10	17	23	59
<i>Sigmoid Bipolar</i>	M	13	3	45	9
	S	4	14	0	51
<i>Softsign</i>	M	9	0	42	2
	S	8	17	3	58

Tabel 5. 7 Hasil perhitungan akurasi dengan ekstraksi ciri GLCM dan NGTDM

Fungsi Aktivasi	Data	Akurasi	Akurasi Total
<i>Sigmoid Biner</i>	<i>Training</i>	77,14 %	73,86 %
	<i>Testing</i>	70,59 %	
<i>Sigmoid Bipolar</i>	<i>Training</i>	91,43 %	85,42 %
	<i>Testing</i>	79,41 %	
<i>Softsign</i>	<i>Training</i>	95,24 %	85,85 %
	<i>Testing</i>	76,47 %	

5.4 Pembahasan Hasil Pengujian

Penyebab utama rendahnya akurasi sistem klasifikasi tingkat keparahan NPDR berdasarkan *hard exudate* menggunakan *Extreme Learning Machine* adalah proses segmentasi yang tidak sempurna. Banyaknya wilayah *non-exudate* yang ikut tersegmentasi membuat kualitas informasi (ciri) pada proses ekstraksi ciri mengurangi tingkat akurasi sistem. Selain itu penentuan penggunaan fungsi aktivasi dan banyaknya node pada tiap *layer* pada model *Extreme Learning Machine* juga menentukan tingkat akurasi sistem.

BAB VI

KESIMPULAN DAN SARAN

Pada bab ini berisi tentang beberapa kesimpulan yang dihasilkan berdasarkan penelitian yang telah dilaksanakan. dan saran yang dapat digunakan jika penelitian ini dikembangkan.

6.1 Kesimpulan

Berdasarkan uji coba pada bab sebelumnya, dapat diambil kesimpulan sebagai berikut :

1. Penelitian ini telah berhasil mengklasifikasikan tingkat keparahan NPDR berdasarkan *hard exudate* menggunakan *Extreme Learning Machine* dengan melakukan eliminasi *optic* disk dan segmentasi *exudate* terlebih dahulu. Karakteristik citra *exudate* yang telah tersegmentasi diekstrak menggunakan GLCM dan digunakan sebagai masukan pada proses pelatihan dan pengenalan pada metode *Extreme Learning Machine*.
2. Sistem berhasil mencapai tingkat akurasi tertinggi menggunakan fungsi aktivasi *sigmoid bipolar* pada model *Extreme Learning Machine* yaitu sebesar 91,22 % untuk ekstraksi ciri menggunakan GLCM. Sedangkan untuk ekstraksi ciri menggunakan NGTDM dan gabungan GLCM dan NGTDM didapatkan tingkat akurasi tertinggi sebesar 90,74 % dan 85,85 % dengan menggunakan fungsi aktivasi *softsign*.
3. Keakuratan sistem dipengaruhi oleh fungsi aktivasi yang digunakan dan banyaknya ciri statistik yang dihasilkan pada proses ekstraksi ciri. Segmentasi *exudate* juga berpengaruh pada keakuratan sistem. Banyaknya daerah *non-exudate* yang ikut tersegmentasi akan memberikan hasil klasifikasi yang kurang akurat.

6.2 Saran

Ada beberapa hal yang penulis sarankan untuk pengembangan penelitian selanjutnya :

1. Melakukan segmentasi *exudate* dengan menggunakan metode lain yang lebih akurat.
2. Menambah ruang lingkup penelitian dengan menambahkan tingkat keparahan NPDR dan menambah gejala NPDR seperti *microaneuriysms* dan *haemorrhages*.

DAFTAR PUSTAKA

- [1] <http://www.who.int/campaigns/world-health-day/2016/en/> diakses pada Senin, 9 Februari 2017 pukul 11.00
- [2] <http://www.who.int/diabetes/global-report/en/> diakses pada Senin, 9 Februari 2017 pukul 11.00
- [3] Soewondo, P., Soegondo, S., Suastika, K., Pranoto, A., Soeatmadji, D.W., dan Tjokroprawiro, A. The DiabCare Asia 2008 study - Out-comes on control and complications of type 2 diabetic patients in Indonesia. **Med J Indones. 2010;19(4):235-43.**
- [4] Vandarkuzhali, T., dan Ravichandran, DR.C.S. (2005). “ELM Based Detection of Abnormality in Retinal Image of Eye Due to Diabetic Retinopathy”. **Journal of Theoretical an Applied Information Technology Vol. 66**, No. 2, Hal. 423–428.
- [5] Tjandrasa, H., Putra, R. E., Wijaya, A. Y., dan Arieshanti, I. “Classification of Non-Proliferative Diabetic Retinopathy Based on Hard Exudate Usig Soft Margin SVM”. **International Conference on Control System, Computing and Engineering, IEEE 2013.** Hal. 376-380.
- [6] Gowthaman R. (2014). “Automatic Identification and Classification of Microaneurysms for Detection of Diabetic Retinopathy”. **International Journal of Research in Engineering and Technology Vol. 3**, Hal. 464-473.
- [7] Pratt, H., Coenen, F., Broadbent, D. M., Harding, S. P., dan Zheng, Y. (2016). “Convolutonal Neural Network for Diabetic Retinopathy”. **International Conference On Medical Imaging Understanding and Analysis 2016, MIUA 2016.**
- [8] The Royal College of Ophthalmologist. (2012). *Diabetic Retinopathy Guideline*. London.

- [9] Vislisel J, dan Oetting T.A. *Diabetic Retinopathy : from one medical student to another*. EyeRounds.org. Sept. 1,2010; Tersedia di: <http://www.EyeRounds.org/tutorials/diabetic-retinopathy-med-students/>
- [10] Gonzales, R. C dan Woods, R. E. (2002). *Digital Image Processing 2/E*. New Jersey: Prentice-Hall, Inc.
- [11] <http://informatika.stei.itb.ac.id/~rinaldi.munir/Buku/Pengolahan%20Citra%20Digital/> diunduh pada kamis, 3 November 2016 pukul 16.05
- [12] Jain, A.K.(1989). *Fundamental of Digital Image Processing*. New Jersey: Prentice-Hall, Inc.
- [13] Acherya, Tinku dan Ray, A.K. (2005). *Image Processing Principles and Applications*. Arizona: A John Wiley & Sons, Inc.
- [14] Haralick, R. M., Shanmugam, K., dan Dinstein, I. (1973). "Textural Feature for Image Classification". **IEEE Transactions on System, Man and Cybernetics Vol. SMC-3**, Hal. 610-621.
- [15] Amadasun, M., dan King, R. (1989). "Textural Features Corresponding to Textural Properties". **IEEE Transactions on System, Man and Cybernetics Vol. 19**, Hal. 1264-1274.
- [16] Huang, G. B., Zhu, Q. Y., dan Siew, C. K. (2006). "Extreme Learning Machine: A new Theory and Applications". **Neurocomputing Vol. 70**, Hal. 489-501.
- [17] MESSIDOR: *Methods for Evaluating Segmentation and Indexing techniques Dedicated to Retinal Ophthalmology*. (2004). [online] <http://www.adcis.net/en/Download-Third-Party/Messidor.html> diakses pada tanggal 9 Februari 2017 pukul 07.21
- [18] Fikriya, Z. A. (2017). "Implementasi *Extreme Learning Machine* untuk Pengenalan Objek Citra Digital". Tugas

Akhir. Departemen Matematika. Institut Teknologi
Sepuluh Nopember, Surabaya.

LAMPIRAN A
Pengujian dengan Menggunakan Fungsi *Sigmoid*
Bipolar

1. Hasil Klasifikasi Tingkat Keparahan NPDR dengan Ekstraksi Ciri Menggunakan GLCM
 Ciri statistik yang diekstrak menggunakan GLCM antara lain: *angular second moment (energy)*, *contrast*, *correlation*, *dissimilarity*, *inverse difference moment (homogeneity)*, dan *entropy*.

Tabel A. 1. Hasil klasifikasi tingkat keparahan NPDR dengan ekstraksi ciri menggunakan GLCM

No	Nama Citra	Tingkat Keparahan	Hasil Klasifikasi Tingkat Keparahan	
1	01M	Moderate	Moderate	Benar
2	02S	Severe	Severe	Benar
3	05S	Severe	Severe	Benar
4	06S	Severe	Severe	Benar
5	08M	Moderate	Moderate	Benar
6	09M	Moderate	Moderate	Benar
7	09S	Severe	Severe	Benar
8	10S	Severe	Severe	Benar
9	12S	Severe	Severe	Benar
10	14S	Severe	Severe	Benar
11	17M	Moderate	Moderate	Benar
12	19M	Moderate	Moderate	Benar
13	19S	Severe	Moderate	Salah
14	23M	Moderate	Moderate	Benar
15	25M	Moderate	Severe	Salah
16	34M	Moderate	Moderate	Benar
17	36M	Moderate	Moderate	Benar
18	41S	Severe	Severe	Benar
19	43M	Moderate	Severe	Salah

20	47M	Moderate	Moderate	Benar
21	48M	Moderate	Severe	Salah
22	55M	Moderate	Moderate	Benar
23	56M	Moderate	Moderate	Benar
24	56S	Severe	Severe	Benar
25	57S	Severe	Moderate	Salah
26	59M	Moderate	Moderate	Benar
27	60S	Severe	Severe	Benar
28	61M	Moderate	Moderate	Benar
29	61S	Severe	Severe	Benar
30	62M	Moderate	Moderate	Benar
31	62S	Severe	Severe	Benar
32	69S	Severe	Severe	Benar
33	75S	Severe	Severe	Benar
34	77S	Severe	Severe	Benar

2. Hasil Klasifikasi Tingkat Keparahan NPDR dengan Ekstraksi Ciri Menggunakan NGTDM

Ciri statistik yang diekstrak menggunakan NGTDM antara lain: *coarsness*, *contrast*, *bussyness*, *complexity*, dan *texture strength*.

Tabel A. 2. Hasil klasifikasi tingkat keparahan NPDR dengan ekstraksi ciri menggunakan NGTDM

No	Nama Citra	Tingkat Keparahan	Hasil Klasifikasi Tingkat Keparahan	
1	01M	Moderate	Moderate	Benar
2	02S	Severe	Severe	Benar
3	05S	Severe	Severe	Benar
4	06S	Severe	Severe	Benar
5	08M	Moderate	Moderate	Benar
6	09M	Moderate	Moderate	Benar
7	09S	Severe	Severe	Benar
8	10S	Severe	Severe	Benar

9	12S	Severe	Severe	Benar
10	14S	Severe	Moderate	Salah
11	17M	Moderate	Severe	Salah
12	19M	Moderate	Moderate	Benar
13	19S	Severe	Moderate	Salah
14	23M	Moderate	Moderate	Benar
15	25M	Moderate	Moderate	Benar
16	34M	Moderate	Moderate	Benar
17	36M	Moderate	Moderate	Benar
18	41S	Severe	Severe	Benar
19	43M	Moderate	Severe	Salah
20	47M	Moderate	Moderate	Benar
21	48M	Moderate	Severe	Salah
22	55M	Moderate	Moderate	Benar
23	56M	Moderate	Moderate	Benar
24	56S	Severe	Severe	Benar
25	57S	Severe	Moderate	Salah
26	59M	Moderate	Moderate	Benar
27	60S	Severe	Severe	Benar
28	61M	Moderate	Moderate	Benar
29	61S	Severe	Severe	Benar
30	62M	Moderate	Moderate	Benar
31	62S	Severe	Severe	Benar
32	69S	Severe	Severe	Benar
33	75S	Severe	Severe	Benar
34	77S	Severe	Severe	Benar

3. Hasil Klasifikasi Tingkat Keparahan NPDR dengan Ekstraksi Ciri Menggunakan GLCM dan NGTDM

Ciri statistik yang diekstrak menggunakan GLCM dan NGTDM antara lain: *angular second moment (energy)*, *contrast*, *correlation*, *dissimilarity*, *inverse difference moment (homogeneity)*, *entropy*, *coarsness*, *bussyness*, *complexity*, dan *texture strength*.

Tabel A. 3.. Hasil klasifikasi tingkat keparahan NPDR dengan ekstraksi ciri menggunakan GLCM dan NGTDM

No	Nama Citra	Tingkat Keparahan	Hasil Klasifikasi Tingkat Keparahan	
1	01M	Moderate	Moderate	Benar
2	02S	Severe	Severe	Benar
3	05S	Severe	Severe	Benar
4	06S	Severe	Severe	Benar
5	08M	Moderate	Moderate	Benar
6	09M	Moderate	Moderate	Benar
7	09S	Severe	Severe	Benar
8	10S	Severe	Severe	Benar
9	12S	Severe	Severe	Benar
10	14S	Severe	Severe	Benar
11	17M	Moderate	Severe	Salah
12	19M	Moderate	Moderate	Benar
13	19S	Severe	Moderate	Salah
14	23M	Moderate	Moderate	Benar
15	25M	Moderate	Severe	Salah
16	34M	Moderate	Moderate	Benar
17	36M	Moderate	Moderate	Benar
18	41S	Severe	Severe	Benar
19	43M	Moderate	Severe	Salah
20	47M	Moderate	Moderate	Benar
21	48M	Moderate	Severe	Salah
22	55M	Moderate	Moderate	Benar
23	56M	Moderate	Moderate	Benar
24	56S	Severe	Severe	Benar
25	57S	Severe	Moderate	Salah
26	59M	Moderate	Moderate	Benar
27	60S	Severe	Severe	Benar
28	61M	Moderate	Moderate	Benar
29	61S	Severe	Moderate	Salah
30	62M	Moderate	Moderate	Benar

31	62S	Severe	Severe	Benar
32	69S	Severe	Severe	Benar
33	75S	Severe	Severe	Benar
34	77S	Severe	Severe	Benar

BIODATA PENULIS



Penulis bernama lengkap **Dinda Ulima Rizky Yani**, lahir di Surabaya, 17 November 1995. Jenjang pendidikan formal yang ditempuh oleh penulis dimulai dari TK Cempaka (2000-2001), SDN Babatan I/456 (2001-2007), SMPN 16 Surabaya (2007-2010), SMAN 1 Surabaya (2010-2013). Setelah lulus dari SMAN 1 Surabaya pada tahun

2013, penulis melanjutkan studi ke jenjang S1 di departemen Matematika di Institut Teknologi Sepuluh Nopember (ITS) Surabaya. Di departemen Matematika ITS penulis mengambil bidang minat Ilmu Komputer. Selama mengikuti perkuliahan di ITS, penulis turut aktif dalam beberapa kegiatan kemahasiswaan sebagai staff Bidang Personalia UKM KOPMA dr. Angka ITS pada tahun 2014, staff Departemen Dalam Negeri Himatika ITS Periode 2014/2015, dan sebagai staff ahli Sport and Art Development Department Himatika ITS Periode 2015/2016. Selain aktif dalam beberapa kegiatan kemahasiswaan, penulis juga mengikuti Kerja Praktek di PT. Telekomunikasi Indonesia Tbk pada tahun 2016. Informasi lebih lanjut mengenai Tugas Akhir ini dapat ditujukan ke penulis melalui email: dindaulima@gmail.com.