

TESIS

**PENGOLAHAN CITRA DIGITAL UNTUK KLASIFIKASI PENYAKIT
DIABETES RETINOPATI MENGGUNAKAN EKSTRAKSI CIRI GLCM
DAN KNN CLASSIFIER**



Disusun oleh:

Nama : Yusuf Fadilla Rachman
NIM : 18.51.1139
Konsentrasi : Business Intelligence

**PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA**

2021

TESIS

**PENGOLAHAN CITRA DIGITAL UNTUK KLASIFIKASI PENYAKIT
DIABETES RETINOPATI MENGGUNAKAN EKSTRAKSI CIRI GLCM
DAN KNN CLASSIFIER**

**DIGITAL IMAGE PROCESSING FOR RETINOPATHY DIABETES
CLASSIFICATION BASED ON GLCM (GRAY LEVEL CO-OCCURANCE
MATRIX) FEATURE EXTRACTION AND KNN CLASSIFIER**

Diajukan untuk memenuhi salah satu syarat memperoleh derajat Magister



Disusun oleh:

Nama : Yusuf Fadlila Rachman
NIM : 18.51.1139
Konsentrasi : Business Intelligence

**PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA**

2021

HALAMAN PENGESAHAN

**PENGOLAHAN CITRA DIGITAL UNTUK KLASIFIKASI PENYAKIT DIABETES
RETINOPATI MENGGUNAKAN EKSTRAKSI CIRI GLCM DAN KNN CLASSIFIER**

**DIGITAL IMAGE PROCESSING FOR RETINOPATHY DIABETES
CLASSIFICATION BASED ON GLCM (GRAY LEVEL CO-OCCURANCE
MATRIX) FEATURE EXTRACTION AND KNN CLASSIFIER**

Dipersiapkan dan Disusun oleh

Yusuf Fadlila Rachman

18.51.1139

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Rabu, 2 Desember 2020

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer

Yogyakarta, 2 Desember 2020

Rektor

Prof. Dr. M. Suyanto, M.M.
NIK. 190302001

HALAMAN PERSETUJUAN

**PENGOLAHAN CITRA DIGITAL UNTUK KLASIFIKASI PENYAKIT DIABETES
RETINOPATI MENGGUNAKAN EKSTRAKSI CIRI GLCM DAN KNN CLASSIFIER**

**DIGITAL IMAGE PROCESSING FOR RETINOPATHY DIABETES
CLASSIFICATION BASED ON GLCM (GRAY LEVEL CO-OCCURANCE
MATRIX) FEATURE EXTRACTION AND KNN CLASSIFIER**

Dipersiapkan dan Disusun oleh

Yusuf Fadlila Rachman

18.51.1139

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Rabu, 2 Desember 2020

Pembimbing Utama

Dr. Kusrini, M.Kom.
NIK. 190302106

Pembimbing Pendamping

Hanif Al Fatta, M.Kom.
NIK. 190302096

Anggota Tim Penguji

Dr. Kusrini, M.Kom.
NIK. 190302106

Dr. Andi Sunyoto, M.Kom.
NIK. 190302052

Dr. Arief Setyanto, S.Si, M.T.
NIK. 190302036

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer

Yogyakarta, 2 Desember 2020
Direktur Program Pascasarjana

Dr. Kusrini, M.Kom.
NIK. 190302106

HALAMAN PERNYATAAN KEASLIAN TESIS

Yang bertandatangan di bawah ini,

Nama mahasiswa : Yusuf Fadlila Rachman
NIM : 18.51.1139
Konsentrasi : Business Intelligence

Menyatakan bahwa Tesis dengan judul berikut:

**PENGOLAHAN CITRA DIGITAL UNTUK KLASIFIKASI PENYAKIT
DIABETES RETINOPATI MENGGUNAKAN EKSTRAKSI CIRI GLCM
DAN KNN CLASSIFIER**

Dosen Pembimbing Utama : Dr. Kusnini, M. Kom

Dosen Pembimbing Pendamping : Hanif Al Fatta, M. Kom

1. Karya tulis ini adalah benar-benar ASLI dan BELUM PERNAH diajukan untuk mendapatkan gelar akademik, baik di Universitas AMIKOM Yogyakarta maupun di Perguruan Tinggi lainnya
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian SAYA sendiri, tanpa bantuan pihak lain kecuali arahan dari Tim Dosen Pembimbing
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab SAYA, bukan tanggung jawab Universitas AMIKOM Yogyakarta
5. Pernyataan ini SAYA buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka SAYA bersedia menerima SANKSI AKADEMIK dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi

Yogyakarta, 2 Desember 2020

Yang Menyatakan,



Yusuf Fadlila Rachman

HALAMAN PERSEMBAHAN

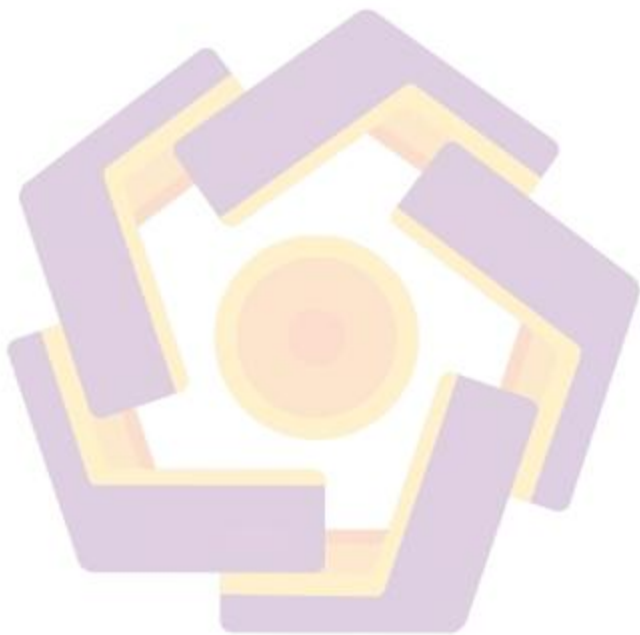
Pertama dan paling utama, saya ucapkan puji syukur kehadiran Allah SWT yang telah memberikan kemudahan dan kelancaran dalam proses pembuatan tugas akhir ini. Tugas Akhir ini saya persembahkan untuk :

1. Kedua orang tua dan keluarga yang senantiasa memberikan semangat dan doa, semoga selalu dalam lindungan-NYA.
2. Ibu Dr. Kusriani, M. Kom dan Bapak Hanif Al Fatta, M. Kom yang telah memberikan bimbingan aktif selama pelaksanaan penelitian, semoga mendapatkan banyak keberkahan dan dilancarkan segala urusannya.

Keluarga besar yang selalu mendukung dan memberikan semangat tanpa henti serta juga yang selalu ada dalam keadaan apapun. Serta semua pihak yang tidak bisa penulis sebutkan satu persatu.

HALAMAN MOTTO

"Life is Bore if you don't challenge yourself"



KATA PENGANTAR

Puji syukur penulis panjatkan kehadiran Allah SWT karena atas limpahan rahmat, hidayah serta inayah-NYA, penulis masih diberi kesempatan dan kemudahan untuk menyelesaikan tesis ini.

Tesis ini disusun dalam rangka memenuhi salah satu syarat kelulusan perguruan tinggi Program Studi Strata-2 Magister Teknik Informatika di Universitas Amikom Yogyakarta dan meraih gelar Magister Komputer (M. Kom). Selain itu tesis ini juga bertujuan untuk menambah pengetahuan tentang sistem pakar yang dibuat menggunakan teorema bayes kepada pembaca yang menggunakan web sebagai basis aksesnya.

Penulis juga mengucapkan terimakasih yang sebesar-besarnya kepada:

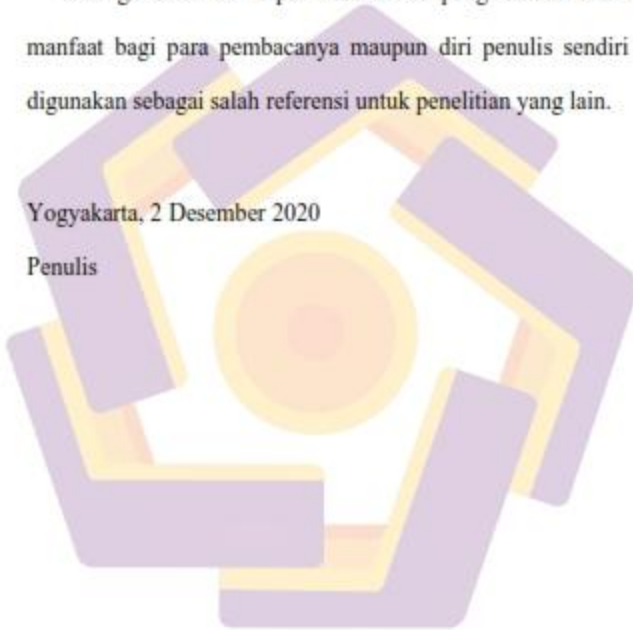
1. Bapak Prof. Dr. M. Suyanto, MM, selaku ketua Amikom Yogyakarta.
2. Ibu Dr. Kusriani, M. Kom dan Bapak Hanif Al Fatta, M. Kom selaku dosen pembimbing yang telah dengan sabar dalam memberikan masukan, saran, bantuan, dan bimbingan dalam menyelesaikan naskah skripsi ini.
3. Ibu Dr. Kusriani, M. Kom selaku Direktur Program Pascasarjana Universitas Amikom Yogyakarta.
4. Kedua Orang tua yang tak pernah lelah dalam memberikan support dan doanya.
5. Dosen Amikom Yogyakarta yang telah memberikan banyak ilmu dan pengalaman.

Semua pihak yang telah membantu penyelesaian skripsi ini yang tentunya sangat berharga dan tidak bisa disebutkan satu persatu. Penulis juga meminta maaf apabila dalam penyusunan tesis ini masih banyak kekurangan dan masih jauh untuk memberikan kata sempurna. Penulis juga dengan senang hati menerima kritik dan saran.

Semoga tesis ini dapat menambah pengetahuan dan memberikan manfaat bagi para pembacanya maupun diri penulis sendiri serta dapat digunakan sebagai salah referensi untuk penelitian yang lain.

Yogyakarta, 2 Desember 2020

Penulis



DAFTAR ISI

HALAMAN JUDUL.....	ii
HALAMAN PENGESAHAN.....	iii
HALAMAN PERSETUJUAN.....	iv
HALAMAN PERNYATAAN KEASLIAN TESIS.....	v
HALAMAN PERSEMBAHAN.....	vi
HALAMAN MOTTO.....	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	x
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR.....	xiv
INTISARI.....	xvi
<i>ABSTRACT</i>	xvii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang Masalah.....	1
1.2. Rumusan Masalah.....	4
1.3. Batasan Masalah.....	5
1.4. Tujuan Penelitian.....	5
1.5. Manfaat Penelitian.....	6
BAB II TINJAUAN PUSTAKA.....	7
2.1. Tinjauan Pustaka.....	7
2.2. Keaslian Penelitian.....	11

2.3. Landasan Teori.....	14
2.3.1 Retina	14
2.3.2. Retinopati Diabetes.....	14
2.3.3 Citra Digital	18
2.3.4 Pengolahan Citra Digital.....	19
2.3.5 Ekstraksi Ciri	21
2.3.6 Ekstraksi Ciri Statistik Orde Kedua.....	22
2.3.7. Gray Level Co-Occurrence (GLCM).....	25
2.3.8 Klasifikasi	28
2.3.9. K-Nearest Neighbors	29
2.3.10. Pengujian Sistem.....	30
BAB III Metode penelitian.....	32
3.1. Jenis, Sifat dan Pendekatan Penelitian.....	32
3.2. Metode Analisis Data.....	32
3.3. Alur Penelitian	33
3.4. Metode Pengumpulan Data.....	34
3.5. Data Preprocessing.....	34
3.6. Ekstraksi Ciri	35
3.7. Klasifikasi	36
3.8. Pengujian Sistem.....	36
BAB IV HASIL PENELITIAN DAN PEMBAHASAN	38

4.1. Deskripsi Data.....	38
4.2. Preprocessing Data.....	39
4.3. Ekstraksi Ciri.....	41
4.4. Klasifikasi.....	50
4.5. Pengujian Sistem.....	63
BAB V PENUTUP.....	68
5.1. Kesimpulan.....	68
5.2. Saran.....	70
DAFTAR PUSTAKA.....	71
LAMPIRAN.....	74
Lampiran 1 Ekstraksi Ciri dan Preprocessing Data.....	75
Lampiran 3 Source Code GLCM.....	76
Lampiran 4 Weka Fitur Selection.....	78
Lampiran 5 Source Code Algoritma KNN.....	79
Lampiran 6 Hasil Klasifikasi KNN (Weka).....	80
Lampiran 7 Hasil Klasifikasi Menggunakan 4 Kelas.....	80
Lampiran 8 Hasil Klasifikasi Menggunakan 2 Kelas.....	81

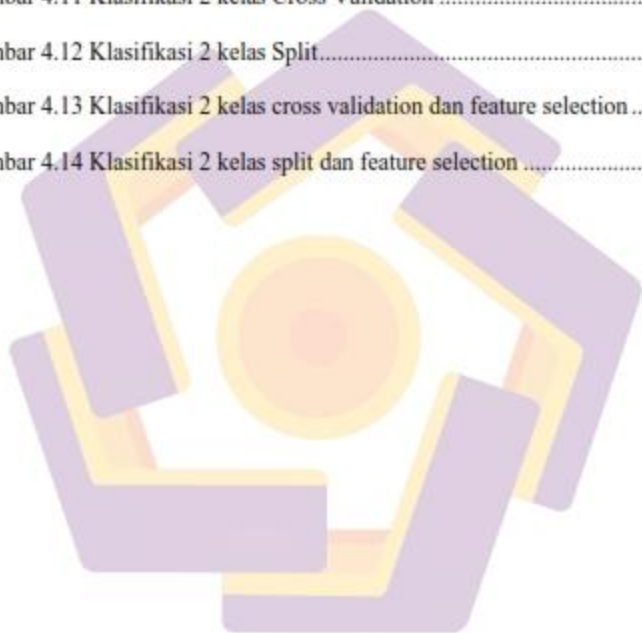
DAFTAR TABEL

Table 2.1. Tabel Literatur Review	11
Tabel 2.2. Area Kerja Matriks.....	26
Tabel 2.3. Normalisasi Matriks Simetris.....	27
Tabel 2.4. Contoh Confusion Matrix	30
Tabel 4.1. Sebaran Data	38
Tabel 4.2. Matriks Normalisasi.....	42
Tabel 4.3. Nilai piksel matriks GLCM.....	43
Tabel 4.4. Fitur GLCM	48
Tabel 4.5. Sebagian Hasil Akhir Ekstraksi Ciri Ordo Kedua	49
Tabel 4.6. Data Contoh Perhitungan KNN	51
Tabel 4.7 Hasil klasifikasi menggunakan cross validation	53
Tabel 4.8 Hasil klasifikasi menggunakan teknik split	54
Tabel 4.9 Hasil klasifikasi cross validation setelah fitur selection	56
Tabel 4.10 Hasil klasifikasi teknik split setelah fitur selection.....	57
Tabel 4.11 Perbandingan klasifikasi cross validation dan data testing.....	57
Tabel 4.12 Hasil Klasifikasi 2 Target kelas	60
Tabel 4.13 Hasil klasifikasi 2 kelas setelah fitur selection	62
Tabel 4.14 Metrik precision dan recall	63

DAFTAR GAMBAR

Gambar 2.1 Retina Normal	15
Gambar 2.2 Mild NPDR	16
Gambar 2.3 <i>Moderate NPDR</i>	17
Gambar 2.4 <i>Severe NPDR</i>	17
Gambar 2.5 PDR	18
Gambar 2.6 Macular Edema	18
Gambar 2.7 Matriks Citra Digital	19
Gambar 2.8 Pengolahan Citra Digital	19
Gambar 2.9. Matriks Kookurensi.....	26
Gambar 2.10. Hubungan spasial $d = 1$ dan $\theta=0^0$	26
Gambar 3.1. Analisis Data	33
Gambar 3.2. Alur Penelitian.....	33
Gambar 3.3. Data Preprocessing.....	35
Gambar 3.4. Alur Pembentukan Matriks Kookurensi (GLCM)	35
Gambar 3.5. Klasifikasi KNN.....	36
Gambar 3.6. Pengujian Sistem	37
Gambar 4.1. Hasil Konversi Citra <i>Grayscale</i>	39
Gambar 4.2. Optic Disc.....	40
Gambar 4.3. Remove Optic Disc	40
Gambar 4.4. Matriks A.....	41
Gambar 4.5. Hubungan Antar piksel	42

Gambar 4.6 Hasil fitur selection	50
Gambar 4.7 Grafik akurasi cross validation.....	53
Gambar 4.8 Grafik akurasi teknik split.....	54
Gambar 4.9 Grafik Cross Validation dan Fitur Selection	55
Gambar 4.10 Grafik Fitur Selection dan Split	56
Gambar 4.11 Klasifikasi 2 kelas Cross Validation	59
Gambar 4.12 Klasifikasi 2 kelas Split.....	59
Gambar 4.13 Klasifikasi 2 kelas cross validation dan feature selection.....	61
Gambar 4.14 Klasifikasi 2 kelas split dan feature selection	61



INTISARI

Diabetes Retinopati, merupakan salah satu penyakit yang timbul akibat dari komplikasi diabetes. Penyakit ini menyerang mata pasien hingga dapat mengalami kebutaan ketika memasuki tingkat keparahan yang tinggi. Terdapat 4 tingkat keparahan yang dapat dialami oleh penderita, yaitu normal, mild (rendah), moderate (sedang), dan severe (parah). Penelitian ini bertujuan untuk mengembangkan sebuah sistem yang dapat digunakan untuk mengidentifikasi penyakit, sekaligus tingkat keparahan penderita Retinopati Diabetes berdasarkan gambar retina mata pasien.

Penelitian ini terbagi menjadi 2 bagian, yang pertama adalah ekstraksi ciri dan yang kedua adalah klasifikasi. Proses ekstraksi ciri dilakukan menggunakan metode Grey Level Cooccurrence Matrix (GLCM). Sedangkan fitur citra yang digunakan adalah Energi, Homogenitas, Kontras, Korelasi, Angular Second Moment (ASM), dan Dissimilarity. Klasifikasi dilakukan menggunakan algoritma K-Nearest Neighbors dengan jumlah dataset sebanyak 600 citra digital retina.

Hasil akhir dari penelitian ini menunjukkan bahwa sistem memiliki akurasi sebesar 32,08% dalam melakukan identifikasi tingkat keparahan penderita Diabetes Retinopati dan 77,083% ketika mengidentifikasi penyakit Diabetes Retinopati.

Kata Kunci: Retinopati Diabetes, Ekstraksi Ciri, GLCM, Klasifikasi, KNN

ABSTRACT

Diabetes retinopathy, is a disease that arises from complications of diabetes. The disease attacks the patient's eyes so that he can become blind when it enters a high level of severity. There are 4 levels of severity that can be read by sufferers, namely normal, mild (low), moderate (severe), and severe (severe). This study aims to develop a system that can be used to identify disease, as well as the severity of people with diabetic retinopathy based on retina images of the eye.

This research is divided into 2 parts, the first is feature extraction and the second is classification. The feature extraction process is carried out using the Gray Level Cooccurrence Matrix (GLCM) method. Meanwhile, the image features used are Energy, Homogeneity, Contrast, Correlation, Angular Second Moment (ASM), and Dissimilarity. Classification is done using the K-Nearest Neighbors algorithm with a total dataset of 600 digital retinal images.

The final results of this study indicate that the system has an accuracy of 32.08% in detecting the severity of Diabetes Retinopathy sufferers and 77.083% as identification of Diabetes Retinopathy.

Keywords: *Diabetic Retinopathy, Feature Extraction, GLCM, Classification, KNN*



BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Teknologi informasi dan komunikasi telah berkembang cepat di Indonesia dan banyak dirasakan manfaatnya. Perkembangan teknologi informasi sendiri telah merambah di berbagai bidang, salah satunya di bidang medis. Akan tetapi, di Indonesia masih sedikit penelitian pada bidang medis yang berhubungan dengan laboratorium klinis (Usman, 2008).

Diabetes merupakan salah satu penyakit yang memiliki pertumbuhan paling cepat di dunia (Kingham, 2009). Menurut data WHO (*World Health Organization*), pada tahun 2013 terdapat 382 juta orang hidup dengan diabetes dan diperkirakan akan meningkat menjadi 592 juta orang pada tahun 2035. Penderita diabetes memiliki resiko terkena penyakit lain, seperti stroke, kerusakan syaraf, *retinopathy diabetic*, gagal ginjal, bahkan kematian. Berdasarkan data yang dimiliki RSUP Dr. Cipto Mangunkusumo Jakarta pada tahun 2011, resiko komplikasi tertinggi yang dialami penderita diabetes adalah *neuropati* (kerusakan syaraf) 54%, diabetes retinopati 33,4%, dan *proteinuria* 26,5% (Anonim, Waspada Diabetes Eat Well Live Well Situasi Dan Analisis Diabetes, 2014).

Salah satu penyakit akibat dari komplikasi penyakit diabetes adalah Diabetes Retinopati (DR). Diabetes Retinopati ini merusak bagian retina mata dan mengakibatkan terganggunya penglihatan penderita. Dampak paling parah dari komplikasi ini akan menyebabkan kebutaan permanen jika terlambat ditangani

(Dillak, R.Y., Harjoko, A, 2011). Penderita yang mengalami diabetes retinopati akan mengalami gejala antara lain: *microneurysm*, *hemorrhages*, *hard exudates*, *soft exudates*, dan *neovascularis*. Berdasarkan gejala-gejala yang muncul, diabetes retinopati dapat diklasifikasikan ke dalam 4 fase (tingkat keparahan), yaitu normal, *non-proliferative diabetic retinopathy* (NPDR), *proliferative diabetic retinopathy* (PDR), dan *macular edema* (ME) (Fadzil, 2011).

Diagnosa penyakit diabetes retinopati dapat dilakukan menggunakan citra retina (Lukmanto, 1995). Retina merupakan salah satu bagian mata yang dapat menunjukkan gejala penyakit dalam seperti diabetes, lemah jantung, kolesterol, dan sebagainya (Riliandi, 2013). Proses diagnosa dilakukan dengan melihat kondisi retina berdasarkan pembuluh arteri, pembuluh vena, makula, dan keempat kuadran retina. Proses diagnosa ini memerlukan waktu yang lama sehingga dapat menyebabkan lambatnya penanganan terhadap pasien. Untuk mengatasi masalah tersebut, maka perlu dikembangkan sebuah sistem yang dapat membantu dokter dalam proses diagnosa penyakit. Sistem harus mampu memproses citra digital retina secara cepat dan akurat (Dillak, R.Y., Harjoko, A, 2011).

Proses pengembangan sistem diagnosis diperlukan dua proses utama yaitu pengolahan citra dan klasifikasi. Dalam pengolahan citra, sebuah gambar (citra digital) akan diekstraksi fitur-fiturnya untuk mendapatkan informasi yang terkandung dalam citra tersebut. Informasi ini kemudian akan diolah dan menjadi dasar dari pembuatan sistem diagnosis. Terdapat beberapa metode yang digunakan dalam ekstraksi fitur, salah satunya adalah *Gray Level Co-Occurrence* (GLCM). Metode GLCM merupakan metode yang telah terbukti secara efektif menjadi

deskriptor tekstur (Siquera, 2013). GLCM merupakan sebuah matriks yang menggambarkan frekuensi munculnya pasangan piksel dalam jarak d dan orientasi sudut θ tertentu dalam citra. Jarak antar pasangan piksel direpresentasikan dalam derajat, sedangkan orientasi sudut diambil berdasarkan 4 arah, yaitu 0° , 45° , 90° , 135° (Surya, R. A., Fadhil, A., dan Yudhana, A, 2017).

Klasifikasi merupakan salah satu cabang ilmu dari *machine learning* yang memungkinkan untuk menganalisis kelompok data tertentu guna mendapatkan informasi atau pola tertentu. Terdapat beberapa algoritma yang dapat digunakan dalam klasifikasi, salah satunya adalah K-Nearest Neighbors (KNN). KNN termasuk pada *supervised learning* dimana hasil dari *query instance* yang baru, diklasifikasikan berdasarkan mayoritas dari kategori pada KNN. Tujuan dari KNN adalah untuk mengklasifikasikan obyek baru berdasarkan atribut dan *training sample*.

Penelitian terdahulu yang dilakukan oleh Gladi et al (2013) memanfaatkan pengolahan citra untuk mendeteksi penyakit diabetes. Metode yang digunakan adalah transformasi *wavelet* dan metode klasifikasi LVQ (*Learning Vector Quantization*). Transformasi *wavelet* akan menghasilkan suatu citra aproksimasi yang mengandung informasi citra. Citra tersebut kemudian dijadikan vektor ciri sebagai input dari metode LVQ (Riliandi, 2013).

Penelitian lain yang dilakukan oleh P. Mohanaiah et al (2013) menggunakan metode GLCM dalam ekstraksi fitur pada tekstur gambar. Penggunaan metode GLCM berguna dalam ekstraksi fitur pada orde kedua dalam sebuah gambar. Proses ekstraksi fitur yang dilakukan berhasil mendapatkan 4 fitur yaitu *Angular Second*

Moment, Correlation, Inverse Difference Moment, dan Entropy. Hasil dalam penelitian ini menyatakan bahwa proses ekstraksi fitur yang dilakukan dengan metode GLCM mengalami penurunan waktu yang signifikan dibanding metode lain (Mohanaiah., 2013).

Penelitian yang dilakukan oleh Wahyudi S. (2016) menggunakan metode KNN dalam deteksi makulopati diabetes berdasarkan citra retina. Proses ekstraksi ciri dilakukan menggunakan metode Two Dimensional Linear Discriminant Analysis (2DLDA). Metode 2DLDA bertujuan untuk mendapatkan transformasi optimal berupa reduksi dimensi. Hasil yang didapat adalah presentasi akurasi mencapai 93,33% (Setiawan W. D., 2016).

Berdasarkan penelitian sebelumnya, maka usulan penelitian yang akan dilakukan adalah mengidentifikasi fase penyakit diabetes retinopati menggunakan ekstraksi fitur GLCM dan metode klasifikasi KNN. Perbedaan penelitian yang diusulkan dengan penelitian sebelumnya terletak pada analisis hasil klasifikasi berdasarkan ekstraksi ciri yang didapat. Pada penelitian ini akan dianalisa pengaruh fitur-fitur citra digital terhadap hasil klasifikasi.

1.2. Rumusan Masalah

Berdasarkan latar belakang masalah yang telah dijelaskan, maka rumusan masalah yang diambil pada penelitian ini adalah

1. Bagaimana hasil klasifikasi Retinopati Diabetes berdasarkan ekstraksi fitur-fitur citra digital?
2. Bagaimana hasil klasifikasi menggunakan teknik Cross Validation dan Split Percentage?

3. Berapa nilai akurasi, presisi, dan recall yang diperoleh dalam klasifikasi Retinopati Diabetes?
4. Bagaimana pengaruh penggunaan metode feature selection terhadap hasil klasifikasi Retinopati Diabetes?

1.3. Batasan Masalah

Batasan masalah pada penelitian ini adalah:

- a. Dataset yang digunakan berjumlah berjumlah 600 citra retina.
- b. Dataset citra digital yang digunakan didapatkan dari database *Messidor*, berasal dari departemen ophthalmologist.
- c. Data gambar diambil menggunakan *retinograph* memiliki resolusi 1440x960 dengan format TIFF.
- d. Ekstraksi ciri yang digunakan adalah ekstraksi ciri tekstur metode statistik orde kedua.
- e. Parameter ciri orde kedua yang digunakan, yaitu Correlation, Energy, Dissimilarity, Homogeneity, Contrast.
- f. Terdapat 4 kelas yang digunakan sebagai target klasifikasi yaitu, Normal, Mild, Moderate, Severe

1.4. Tujuan Penelitian

Tujuan dari penelitian yang penulis lakukan adalah sebagai berikut:

- a. Klasifikasi citra retina pasien Retinopati Diabetes berdasarkan ekstraksi ciri statistik orde kedua.
- b. Identifikasi tingkat keparahan pasien Retinopati Diabetes berdasarkan citra digital retina.

- c. Identifikasi data pasien sakit atau normal berdasarkan citra digital retina.
- d. Menganalisa hasil klasifikasi berdasarkan ciri statistik citra retina pasien Retinopati Diabetes.

1.5. Manfaat Penelitian

Manfaat yang didapatkan dari penelitian yang dilakukan adalah:

- a. Deteksi dan identifikasi tingkat keparahan yang dialami oleh penderita Retinopati Diabetes berdasarkan citra retina.
- b. Meringankan dan membantu kinerja dokter dalam diagnosa fase penyakit Retinopati Diabetes secara cepat dan akurat.



BAB II

TINJAUAN PUSTAKA

2.1. Tinjauan Pustaka

Penelitian berjudul “Klasifikasi Penyakit Diabetes Retinopati Berdasarkan Citra Digital Dengan Menggunakan Metode Wavelet dan Support Vector Machine” oleh Ratna Ayu Gitasari tahun 2015 (Gitasari, 2015). Penelitian ini menjelaskan tentang deteksi dan klasifikasi penyakit Retinopati Diabetes berdasarkan jumlah mikroanuerisma, dan lemak protein pada retina. Kelas yang digunakan dalam klasifikasi adalah normal, *mild*, *moderate*, dan *severe*. Ekstraksi ciri dilakukan menggunakan metode transformasi wavelet sedangkan klasifikasi dilakukan menggunakan metode SVM. Data yang digunakan berjumlah 60 citra. Hasil yang didapat pada penelitian ini adalah mencapai akurasi tertinggi 72% menggunakan SVM multiclass One-Against-All dengan parameter *skewness*.

Penelitian berjudul “Klasifikasi Citra Retina Menggunakan K-Nearest Neighbor untuk Mendeteksi Makulopati Diabetik oleh Wahyudi Setiawan dan Fitri Damayanti tahun 2016 (Setiawan W. D., 2016). Penelitian ini berisi tentang implementasi metode KNN untuk klasifikasi Makulopati Diabetes. Fitur yang digunakan didapatkan menggunakan metode ekstraksi ciri Two Dimensional Linear Discriminant Analysis (2DLDA). Data yang digunakan sebanyak 75 citra. Klasifikasi dilakukan dalam tiga kelas yaitu Edema Makula tingkat 1, tingkat 2, dan tingkat 3. Hasil dari penelitian ini menunjukkan tingkat akurasi maksimal hingga 93,33%.

Penelitian berjudul “Early Detection of Curable Diabetic Retinopathy from Retinal Images” oleh Darshankumar Gorasiya, Sanket Damaya, et al tahun 2019 (Gorasiya, 2019). Penelitian ini mengimplementasikan metode Convolutional Neural Network. Pre-processing pada penelitian ini dilakukan dengan metode *Contrast Limited Adaptive Histogram Equalization (CLAHE)*. Tujuan penelitian ini adalah untuk mengidentifikasi gejala awal Retinopati Diabetes menggunakan citra retina memakai metodologi *machine learning*. Data yang digunakan pada penelitian ini berjumlah 35,124 citra dan dibagi ke dalam 4 kelas yaitu No-RD, Mild, Moderate, Severe, Proliferative RD. Hasil yang didapat adalah model yang dibuat memiliki akurasi 89%.

Penelitian berjudul “Exudates Detection from Digital Fundus Image of Diabetic Retinopathy” oleh Namrata dan Shaveta Arora tahun 2015 (Namrata dan Arora, 2015). Penelitian ini bertujuan untuk mendeteksi darah putih (*exudate*) dari gambar fundus dan memberikan informasi tentang Non Proliferative Diabetic Retinopathy (NPDR). Kelas yang digunakan untuk klasifikasi adalah no RD, Mild, Moderate, dan Severe. Metode Pre-processing yang digunakan adalah CLAHE atau contrast enhancement. Metode median filter digunakan untuk mendeteksi *exudates* pada gambar. Hasil yang didapat adalah proses deteksi darah putih dapat dilakukan dengan menambahkan *thresholding* pada gambar menggunakan *median filter*.

Penelitian berjudul “Analysis of Retinal Fundus Images for Grading of Diabetic Retinopathy Severity” oleh M.H. Ahmad Fadzil, Lila Iznita Izhar, Hermawan Nugroho, dan Hanung Adi Nugroho tahun 2011 (Fadzil, 2011). Penelitian ini bertujuan untuk mendeteksi gejala awal Retinopati Diabetes dan

pengobatan yang efektif pada kasus yang parah. Objek yang digunakan adalah gambar fundus retina yang digunakan untuk menghitung foveal avascular zone (FAZ). Penelitian ini menggunakan 2 metode pre-processing *contrast enhancement* yaitu CLAHE dan *Independent Component Analysis (ICA)*. Metode *Gaussian Bayes* digunakan untuk mengklasifikasikan tingkat keparahan Retinopati Diabetes berdasarkan FAZ. Tingkat akurasi yang didapat pada penelitian ini adalah 97,46%.

Penelitian berjudul "Support Vector Machine untuk Klasifikasi Citra Jenis Daging Berdasarkan Tekstur Menggunakan Ekstraksi Ciri Gray Level Co-occurrence Matrices (GLCM)" oleh Neneng, Kusworo Adi, R. Rizal Isnanto tahun 2016 (Neneng, 2016). Tujuan penelitian ini adalah mengklasifikasi jenis daging berdasarkan teksturnya. Metode GLCM digunakan dalam proses ekstraksi ciri dan SVM digunakan dalam proses klasifikasi. Data yang digunakan sebanyak 1800 data terbagi menjadi 3 berdasarkan jarak pengambilan gambar yaitu 20 cm, 30 cm, dan 40 cm. Hasil yang didapat pada penelitian ini menunjukkan bahwa sistem berhasil melakukan klasifikasi jenis daging dengan akurasi sebesar 87,5% pada jarak pengambilan gambar 20 cm.

Penelitian berjudul "Image Texture Feature Extraction Using GLCM Approach" oleh P. Mohanaiah, P. Sathyanarayana, dan L. GuruKumar tahun 2013 (Mohanaiah., 2013). Penelitian ini bertujuan untuk mengimplementasikan metode GLCM untuk ekstraksi fitur sebuah citra digital. Terdapat 4 fitur yang digunakan yaitu *Angular Second Moment*, *Correlation*, *Inverse Difference Moment*, dan *Entropy*. Hasil penelitian ini adalah dengan melakukan ekstraksi fitur menggunakan

GLCM maka dapat mereduksi waktu proses saat konversi gambar dari RGB ke Gray Level Image.



2.2. Keaslian Penelitian

Table 2.1. Tabel Literatur Review

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
1	Klasifikasi Penyakit Diabetes Retinopati Berdasarkan Citra Digital Dengan Menggunakan Metode Wavelet dan Support Vector Machine	Ratna Ayu Gitasari. e-Proceeding of Engineering Vol 2 No 1 Telkom University. (2015)	Mengimplementasikan metode wavelet, metode ekstraksi ciri statistik, dan metode klasifikasi SVM dalam menentukan klasifikasi penyakit retinopati diabetes melalui citra retina	Akurasi yang didapat adalah 72% menggunakan parameter ekstraksi ciri <i>skewness</i> , didekomposisi wavelet level 5 dan metode One against one.	Terjadi penurunan akurasi sistem ketika citra ditambahkan dengan <i>noise</i> . Dari akurasi awal 72% menjadi 34% ketika ditambahkan <i>noise</i> .	Perbedaan dengan penelitian yang akan dilakukan terletak pada penggunaan metode ekstraksi ciri (wavelet) dan klasifikasi (SVM). Penelitian yang akan dilakukan menggunakan GLCM dan KNN
2	Klasifikasi Citra Retina Menggunakan K-Nearest Neighbor untuk Mendeteksi Makulopati Diabetik	Wahyudi Setiawan, Fitria Damayanti. Seminar Nasional Sains dan Teknologi Fakultas Teknik Universitas Muhammadiyah Jakarta. (2016)	Mendeteksi tingkat edema makula penyebab makulopati diabetik menggunakan KNN	Akurasi maksimal yang didapat pada penelitian ini sebesar 93,33%	Nilai akurasi bergantung pada jumlah data training. Perlu dilakukan uji coba dengan jumlah data uji tetap dan data latih berbeda	Penelitian ini menggunakan <i>Edema Makula</i> sebagai variabel untuk deteksi makulopati diabetik. Ekstraksi ciri menggunakan Two Dimensional Linear Discriminant Analysis

Tabel 2.1. Tabel Literatur Review (lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
3	Early Detection of Curable Diabetic Retinopathy from Retinal Images	Darshankumar Gorasiya, Sanket Dayama, et all (2019)	Mengidentifikasi gejala awal Retinopati Diabetes secara efisien berdasarkan citra.	Akurasi yang dicapai sebesar 89% disebabkan oleh berkurangnya kompleksitas komputasi yang digunakan	Penggunaan metode yang berbeda pada tahap pre-processing dan klasifikasi dapat dilakukan	Penelitian ini menggunakan metode deep learning CNN. Menggunakan Contrast dan Histogram sebagai metode preprocessing
4	Exudates Detection from Digital Fundus Image of Diabetic Retinopathy	Namrata dan Shaveta Arora. International Journal of Advances in Biology (IJAB) Vol 2. No 4. (2015)	Mendeteksi darah putih dan memberikan informasi tentang Non Proliferative Diabetic Retinopathy berdasarkan gambar digital retina	Metode yang digunakan berhasil mendeteksi darah putih menggunakan gambar retina	Tidak disertai penjelasan tentang metode klasifikasi yang digunakan. Akurasi tidak ditunjukkan	Menggunakan metode CLAHE untuk pre processing data. Thresholding untuk menghapus optical disc. Dan memakai median filter untuk deteksi fitur (darah putih)
5	Analysis of Retinal Fundus Images for Grading of Diabetic Retinopathy Severity	M.H. Ahmad Fadzil, et all. International Federation for Medical and Biological Engineering (2011)	Mendeteksi gejala Retinopati Diabetes dan pengobatan yang efektif pada kasus yang parah	Tingkat rata-rata akurasi yang didapat pada penelitian ini adalah 97,46%.	Menggunakan cross validation untuk memvalidasi metode yang digunakan berdasarkan pakar	Penelitian ini menggunakan Foveal Avascular Zone dalam penentuan tingkat keparahan retinopati diabetes. Menggunakan gaussian bayes sebagai metode klasifikasi.

Tabel 2.1. Tabel Literatur Review (lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
6	Support Vector Machine untuk Klasifikasi Citra Jenis Daging Berdasarkan Tekstur Menggunakan Ekstraksi Ciri Gray Level Co-Occurrence Matrices (GLCM)	Neneng, Kusworo adi, R. Rizal Isnanto. Jurnal Sistem Informasi Bisnis Undip. (2016)	Mengklasifikasi jenis daging berdasarkan tekstur	Sistem menghasilkan akurasi 87,5% pada jarak pengambilan 20 cm, jarak piksel tetangga (d) = 2, dan orientasi sudut (θ) = 135 derajat.	Memperkecil jarak pengambilan gambar mungkin akan memperjelas tekstur pada gambar, sehingga dapat meningkatkan akurasi sistem	Perbedaan terletak pada objek yang penelitian (citra daging) dan metode klasifikasi (SVM) yang digunakan.
7	Image Texture Feature Extraction Using GLCM Approach	P. Mohananiyah, P. Sathyanarayana, L. GuruKumar. International Journal of Scientific and Research Publications (2013)	Mengimplementasikan metode GLCM untuk ekstraksi fitur sebuah citra digital.	Hasil penelitian ini adalah dengan melakukan ekstraksi fitur menggunakan GLCM maka dapat mereduksi waktu proses saat konversi gambar dari RGB ke Gray Level Image	Kesimpulan pada paper menyatakan bahwa penggunaan GLCM lebih baik daripada metode lain. Tetapi tidak disertai data atau perbandingan	Penelitian ini melakukan ekstraksi fitur pada gambar menggunakan 4 fitur yaitu Angular Second Moment, Correlation, Inverse Difference Moment, dan Entropy.

2.3. Landasan Teori

2.3.1 Retina

Retina merupakan lapisan tipis yang sensitif terhadap cahaya dan terletak di bagian belakang bola mata (Lusiani, 2019). Retina memiliki sel batang (basilus) dan sel kerucut (konus) yang bertugas menerima cahaya dan memberikan visualisasi atas apa yang dilihat oleh mata. Retina memiliki sel fotoreseptor yang berguna untuk mendeteksi warna dan intensitas cahaya dari suatu benda yang dilihat mata. Retina kemudian menerima dan memproses informasi yang dikumpulkan sel fotoreseptor dan menyampaikannya ke otak.

Retina memiliki peran yang sangat vital dalam penglihatan, sehingga terjadinya gangguan pada retina sangat mengganggu kinerja penglihatan. Terdapat beberapa jenis penyakit yang dapat menyerang retina (Willy, 2019), yaitu: Ablasio Retina, Retinopati Diabetes, Retinitis Pigmentosa, Retinoblastoma, dan Degenerasi Makula.

2.3.2. Retinopati Diabetes

Retinopati diabetes merupakan gangguan yang terjadi pada mata, yang disebabkan oleh penyakit diabetes. Retinopati diabetes memicu penyumbatan pada pembuluh darah pada bagian retina mata. Retina merupakan bagian mata yang berfungsi mengambil citra (gambar) dan mengirim gambar ke otak. Dalam menjalankan fungsinya, retina membutuhkan asupan darah dari pembuluh darah di sekitarnya. Akibat dari kadar gula yang tinggi pada penderita diabetes, maka pembuluh darah tersebut akan tersumbat. Tekanan darah yang tinggi menyebabkan kerusakan pada pembuluh darah mata, sehingga mempermudah bocornya cairan

atau darah pada retina. Fase yang paling parah pada retinopati diabetes dapat mengakibatkan kebutaan (Rahman, K.,Kartasasmitta, A.S., Heksan, 2013).

Komplikasi retinopati diabetes dapat terjadi tanpa disertai gejala yang dirasakan oleh penderita dikarenakan sifat penyebarannya yang lambat. Gejala yang ditimbulkan oleh komplikasi Retinopati Diabetes biasanya berhubungan dengan pembuluh darah, penglihatan menurun secara bertahap, tampak bercak hitam pada penglihatan, tampak noda melayang pada penglihatan, penglihatan berbayang, sulit membedakan warna, dan nyeri pada mata.

Retinopati Diabetes terbagi menjadi beberapa fase, yaitu fase normal, fase non-proliferative diabetic retinopathy (NPDR), fase proliferative diabetic retinopathy (PDR), dan macular edema (ME) (Fadzil, 2011). Berikut adalah penjelasan masing-masing fase Retinopati Diabetes:

A. Fase Normal

Citra retina pada mata normal tidak terdeteksi adanya mikro aneurisma dan pendarahan. Contoh citra retina normal dapat dilihat pada gambar 2.1

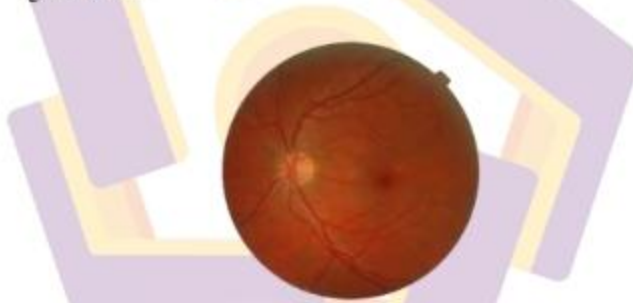


Gambar 2.1 Retina Normal

B. Fase Non-Proliferative Diabetic Retinopathy (NPDR)

Fase NPDR merupakan fase awal pada Retinopati Diabetes. Fase ini terjadi ketika pembuluh darah di retina mengalami perubahan. Perubahan ini tidak dapat terlihat maupun dideteksi oleh mata telanjang. Fase ini memiliki ciri yaitu munculnya titik-titik (*cotton wool spot*), pendarahan retina dan kelainan pada mikrovaskular (Namrata and Aurora, 2015). Fase NPDR terbagi menjadi 3 yaitu *mild*, *moderate*, dan *severe*.

- *Mild NPDR*: jumlah mikroaneurisma antara 0 sampai 5 dan tidak muncul pendarahan. Contoh gambar retina fase mild NPDR dapat dilihat pada gambar 2.2



Gambar 2.2 Mild NPDR

- *Moderate NPDR*: Jumlah mikro aneurisma antara 5 hingga 15 atau jumlah pendarahan sebesar 5. Tidak terjadi neovaskularisasi. Contoh gambar fase *moderate* dapat dilihat pada gambar 2.3.



Gambar 2.3 *Moderate NPDR*

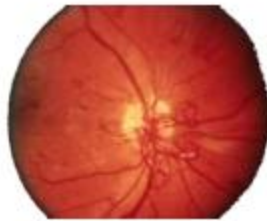
- Severe NPDR: jumlah mikro aneurisma lebih dari 15. Pendarahan lebih dari 5 tempat. Terjadi neovaskularisasi. Contoh gambar retina *severe NPDR* dapat dilihat pada gambar 2.4.



Gambar 2.4 *Severe NPDR*

C. Fase Proliferative Diabetic Retinopathy (PDR)

Fase selanjutnya dari Retinopati Diabetes adalah Proliferative Diabetic Retinopathy (PDR). Pada fase ini terjadi penyumbatan di pembuluh darah retina. Akibat dari penyumbatan ini maka akan tumbuh pembuluh darah baru. Akan tetapi pembuluh darah baru ini sangat rentan dan mudah bocor. Kebocoran pembuluh darah ini akan menyebabkan kebutaan. Contoh gambar retina fase PDR dapat dilihat pada gambar 2.5.



Gambar 2.5 PDR

E. Fase Macular Edema (ME)

Fase lain dari Retinopati Diabetes adalah Macular Edema. Pada fase ini terjadi pembengkakan pada makula (titik tengah retina) yang menyebabkan bocornya cairan pada retina. Contoh gambar Macular Edema dapat dilihat pada gambar 2.6.



Gambar 2.6 Macular Edema

2.3.3 Citra Digital

Citra digital adalah sebuah gambar 2 dimensi yang ditampilkan pada layar komputer sebagai himpunan atau diskrit nilai digital yang disebut pixel. Proses memperbaiki kualitas citra dengan menggunakan perangkat komputer agar mudah diinterpretasi oleh manusia atau mesin. Suatu citra didefinisikan sebagai fungsi $f(x,y)$ berukuran M piksel dan N piksel, dengan x (baris) dan y (kolom) adalah koordinat spasial, sedangkan amplitudo f di titik koordinat (x,y) dinamakan

intensitas atau tingkat keabuan dari citra pada titik tersebut. Apabila nilai x, y dan nilai amplitudo f secara keseluruhan berhingga dan bernilai diskrit, maka citra tersebut adalah citra digital (Gonzalez, R.C. & Woods, R.E., 2010). Citra digital $M \times N$ dapat dituliskan dalam bentuk matriks berukuran M baris dan N kolom seperti pada gambar 2.7.

$$f = [f(x, y)] = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0, M-1) \\ f(1,0) & f(1,1) & \dots & f(1, M-1) \\ \dots & \dots & \dots & \dots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1, M-1) \end{bmatrix}$$

Gambar 2.7 Matriks Citra Digital

Indeks baris (x) dan indeks kolom (y) menyatakan suatu koordinat titik pada citra. Masing masing titik koordinat (x, y) disebut piksel sedangkan $f(x, y)$ disebut intensitas atau tingkat keabuan pada titik (x, y).

2.3.4 Pengolahan Citra Digital

Pengolahan citra dapat dilakukan dengan berbagai cara diantaranya adalah representasi citra, peningkatan kualitas citra, restorasi citra, analisis citra, rekonstruksi citra, dan kompresi citra (Gonzalez, R.C. & Woods, R.E., 2010). Tujuan pengolahan citra digital adalah memperbaiki kualitas citra menggunakan perangkat komputer sehingga dapat diinterpretasi oleh manusia atau mesin. Diagram pengolahan citra dapat dilihat pada gambar 2.8.



Gambar 2.8 Pengolahan Citra Digital

Citra digital memiliki elemen-elemen penyusun (Sutoyo, 2009):

- **Kecerahan:** merupakan intensitas cahaya yang dipancarkan piksel dari citra sehingga dapat ditangkap oleh penglihatan manusia. Kecerahan pada suatu titik dalam citra merupakan intensitas rata-rata dari suatu area yang melingkupinya.
- **Kontras:** merupakan komposisi sebaran gelap terang. Citra yang baik memiliki komposisi gelap terang yang merata.
- **Kontur:** merupakan keadaan yang ditimbulkan oleh perubahan intensitas pada piksel yang bertetangga. Perubahan ini akan menyebabkan mata dapat menangkap tepian objek pada suatu citra.
- **Warna:** merupakan persepsi visual terhadap panjang gelombang cahaya yang dipantulkan oleh objek.
- **Bentuk:** merupakan representasi citra dalam 3 dimensi
- **Tekstur:** merupakan karakteristik suatu objek yang menjadikannya unik atau beda dari objek lainnya. Tekstur dicirikan sebagai distribusi spasial dari derajat keabuan di dalam sekumpulan piksel-piksel yang bertetangga. Tekstur terbentuk dari susunan piksel yang membentuk pola-pola yang teratur.

Teknik pengolahan citra dapat dibagi menjadi 3 tingkatan yaitu pengolahan tingkat rendah, menengah dan tinggi. Pengolahan tingkat rendah merupakan pengolahan tahap dasar meliputi pengurangan noise, perbaikan citra, dan restorasi gambar. Pengolahan tingkat menengah meliputi segmentasi citra, deskripsi objek, dan klasifikasi objek secara terpisah. Sedangkan pengolahan tingkat tinggi mulai dilakukan analisis pada citra untuk mendapatkan informasi pada citra. Berdasarkan

ketiga tingkatan tersebut, maka teknik yang dapat dilakukan pada proses pengolahan citra (Basuki, 2005):

- Image enhancement: proses perbaikan citra dengan meningkatkan kualitas citra baik kontras ataupun kecerahan
- Image restoration: proses memperbaiki model citra sesuai bentuk yang sesuai
- Color Image processing: proses ini secara sederhana adalah mempertajam warna pada suatu citra. Teknik yang adalah image enhancement, image restoration atau lainnya.
- Wavelet multiresolution processing: prose dimana citra akan diterjemahkan ke dalam beberapa resolusi
- Image compression: proses pengubahan ukuran pada citra
- Morphologic processing: proses penggalan informasi pada suatu citra
- Segmentation: proses membedakan atau memisah suatu objek dengan objek lainnya pada suatu citra. Contoh nya adalah memisahkan antara foto objek dengan latar belakangnya
- Object recognition: proses pengenalan objek-objek yang ada pada citra

2.3.5 Ekstraksi Ciri

Ciri/fitur merupakan karakteristik unik yang membedakan antara satu objek dengan yang lain. Fitur pada objek dapat dibedakan menjadi dua bagian yaitu fitur alami dan buatan. Fitur alami merupakan bagian dari gambar, contohnya tingkat kecerahan dan tepi objek. Sedangkan fitur buatan diperoleh dengan mengolah gambar dengan operasi tertentu, contohnya histogram atau tingkat keabuan

(Coltelli, Primo and Gualtieri, Paulo, 1990). Tujuan dari ekstraksi ciri adalah mendapatkan informasi yang terkandung pada suatu citra untuk dijadikan pembeda antara satu citra dengan citra yang lain.

Terdapat beberapa metode ekstraksi fitur yang dapat digunakan antara lain metode statistik, metode spektral, dan metode struktural. Metode statistik bekerja dengan menghitung statistik distribusi derajat keabuan (*Histogram*) untuk mengukur tingkat kontras, granularitas, dan tingkat kekasaran suatu daerah berdasarkan hubungan ketetanggaan antar piksel pada citra digital. Ekstraksi ciri statistik terbagi menjadi 2 macam, yaitu orde pertama dan orde kedua. Metode Spektral bekerja berdasarkan fungsi autokorelasi suatu daerah pada domain transformasi fourier dalam mendeteksi periodisitas tekstur. Metode Struktural menggunakan deskripsi primitif terstruktur dan aturan sintaktis.

2.3.6 Ekstraksi Ciri Statistik Orde Kedua

Salah satu teknik untuk memperoleh ciri statistik orde dua adalah dengan menghitung probabilitas hubungan ketetanggaan antara dua piksel pada jarak dan orientasi sudut tertentu. Pendekatan ini bekerja dengan membentuk sebuah matriks kookurensi dari data citra, dilanjutkan dengan menentukan ciri sebagai fungsi dari matriks antara tersebut (Prasetyo, 2011). Matriks kookurensi berguna untuk menentukan ciri pada suatu citra. Berikut adalah ciri/fitur-fitur citra yang dapat dicari menggunakan matriks kookurensi.

1. Entropy

Entropy menunjukkan distribusi derajat keabuan suatu citra secara acak. Semakin acak distribusi derajat keabuan semakin tinggi nilai entropy yang

dihasilkan. Nilai entropi mengindikasikan kompleksitas citra, semakin tinggi nilai entropi berarti semakin kompleks citra tersebut (Nalband,Saif., Prince,Amalin., Agrawal,Anita , 2017). Entropi memiliki rentang nilai antara -14 hingga 14 pada tengah gambar, dan -15 hingga 15 pada tepi gambar. Nilai entropi dapat dicari menggunakan persamaan 8.

$$Entropy = \sum_{i_1} \sum_{i_2} p(i_1, i_2) \log p(i_1, i_2) \quad (8)$$

2. Energi (*Angular Second Moment*)

Energi dapat disebut juga dengan Angular Second Moment. Energi merupakan fitur pada citra yang merepresentasikan tekstur suatu objek. Proses analisa tekstur dapat menggunakan matriks co-occurrence. Matriks co-occurrence adalah penggambaran frekuensi munculnya pasangan dua piksel dengan intensitas tertentu dalam jarak dan sudut tertentu dalam citra (Basuki, 2005). Energi digunakan untuk mengukur konsentrasi pasangan intensitas pada matriks co-occurrence. Energi memiliki nilai rentang nilai antara 0 hingga 1. Semakin besar nilai energi berarti pasangan piksel pada matriks co-occurrence terpusat pada beberapa koordinat, sedangkan nilai energi mengecil jika letak pasangan piksel menyebar. Nilai energi dapat dicari menggunakan persamaan 9.

$$Energi = \sum_{i_1} \sum_{i_2} p^2(i_1, i_2) \quad (9)$$

3. Kontras

Kontras merupakan tingkat penyebaran piksel kedalam intensitas warna (Purnomo, M.H. dan Muntasa, A, 2010). Kontras terbagi menjadi 3 macam, yaitu kontras rendah, normal, dan tinggi. Kontras rendah dapat terjadi karena kurangnya pencahayaan atau kurangnya bidang dinamika sensor citra. Kontras rendah

ditunjukkan oleh kurva histogram yang sempit akibat penyebaran intensitas yang tidak merata. Kontras normal ditunjukkan dengan lebar kurva histogram yang tidak terlalu lebar ataupun sempit. Kontras tinggi memiliki kurva histogram yang lebar dengan sebaran intensitas gelap terang yang merata pada keseluruhan skala intensitas. Nilai kontras dapat dihitung menggunakan persamaan 10.

$$\text{Kontras} = \sum_{i_1} \sum_{i_2} (i_1 - i_2)^2 p(i_1, i_2) \quad (10)$$

4. Homogenitas (Inverse Difference Moment)

Homogenitas dapat disebut juga Inverse Difference Moment. Homogenitas digunakan untuk mengukur tingkat homogen variasi intensitas pada citra. Homogenitas memiliki rentang nilai antara 0 hingga 1. Semakin besar nilai homogenitas berarti tingkat variasi intensitas citra mengecil. Nilai homogenitas mencapai maksimal jika semua elemen pada gambar sama. Homogenitas dapat didefinisikan dengan persamaan 11:

$$\text{Homogenitas} = \sum_{i_1} \sum_{i_2} \frac{p(i_1, i_2)}{1 + |i_1 - i_2|} \quad (11)$$

5. Korelasi

Korelasi digunakan untuk mengukur ketergantungan linier level keabuan pada piksel tetangga. Korelasi biasanya digunakan untuk mengukur tingkat deformasi, perubahan, atau peregangan aliran optik. Korelasi memiliki nilai antara -1 hingga 1. Korelasi dapat didefinisikan dengan persamaan 12:

$$\text{Korelasi} = \sum_{i_1} \sum_{i_2} \frac{(i_1 - \mu_{i'}) \cdot (j - \mu_{j'}) \cdot p(i, j)}{\sigma_{i'} \cdot \sigma_{j'}} \quad (12)$$

2.3.7. Gray Level Co-Occurrence (GLCM)

Proses perhitungan ciri tekstur pada ekstraksi citra digital dilakukan berdasarkan distribusi kombinasi intensitas piksel pada posisi tertentu. Kombinasi dari masing-masing intensitas piksel ini dibedakan melalui statistik orde pertama, kedua, dan statistik orde yang lebih tinggi. Salah satu cara untuk melakukan ekstraksi statistik orde kedua adalah metode GLCM (R.M. Haralick, K. Shanmugam, dan I. Dinstein, 1973).

GLCM (Gray Level Co-Occurrence Matrix) merupakan suatu matriks yang menggambarkan frekuensi kemunculan pasangan dua piksel dengan intensitas tertentu dalam jarak dan arah tertentu dalam citra (Prasetyo, 2011). Dalam proses analisis citra berdasarkan distribusi statistik dari intensitas pikselnya, dapat dilakukan ekstraksi pada fitur teksturnya (P. P. Pullaperuma, A. T. Dharmaratne, 2013). Kookurensi berarti jumlah kejadian pada satu level nilai piksel bertetangga dengan satu level nilai piksel lain dalam jarak d dan orientasi sudut θ . Bentuk orientasi sudut berdasarkan empat arah sudut dengan interval 45° yaitu 0° , 45° , 90° , dan 135° , sedangkan jarak antar piksel sebesar 1 piksel (Eleyan, A. and Demirel, H., 2011). Matriks kookurensi merupakan matrik bujursangkar yang memiliki elemen sebanyak kuadrat jumlah level intensitas piksel pada citra. Setiap elemen pada matriks kookurensi (i, j) memiliki peluang kejadian bernilai i bertetangga dengan j pada jarak d dan memiliki orientasi sudut θ . Perhitungan matriks kookurensi dilakukan dengan nilai $d=1$ dan $\theta=0^\circ$. Jumlah kemunculan pasangan (i, j) dihitung pada keseluruhan matriks dan pengisian nilai pada matriks berdasarkan

jumlah kookurensi pada posisi sel yang bersesuaian. Pembuatan matriks kookurensi seperti pada gambar 2.10.

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

Gambar 2.9. Matriks Kookurensi

Pada matriks diatas, memiliki empat aras keabuan, maka jumlah piksel tetangga dan nilai piksel referensi pada area kerja matriks berjumlah 4. Ilustrasi diatas dapat dilihat pada tabel 2.2.

Tabel 2.2. Area Kerja Matriks

Nilai Piksel Tetangga	0	1	2	3
Nilai Piksel Referensi				
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

Hubungan spasial untuk $d = 1$ dan $\theta=0^0$ pada matriks dapat digambarkan pada gambar 2.10.

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

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

Gambar 2.10. Hubungan spasial $d = 1$ dan $\theta=0^0$

Dengan memakai sudut orientasi $\theta=0^\circ$ maka akan dicari hubungan piksel-piksel secara horizontal. Maka dari itu angka 2 pada (0,0) berarti jumlah pasangan (0,0) pada matriks asal berjumlah 2. Setelah mendapat matriks kookurensi, kemudian ditambahkan dengan matriks transposenya agar simetris terhadap sumbu diagonal.

Matriks I + Matriks I^T = I simetris

$$\begin{bmatrix} 2 & 2 & 1 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 2 & 0 & 0 & 0 \\ 2 & 2 & 0 & 0 \\ 1 & 0 & 3 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 4 & 2 & 1 & 0 \\ 2 & 4 & 0 & 0 \\ 1 & 0 & 6 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix}$$

Langkah berikutnya adalah melakukan normalisasi matriks yang telah simetris. Maka hasil akhirnya dapat dilihat pada tabel 2.3.

Tabel 2.3. Normalisasi Matriks Simetris

0,1667	0,0833	0,0042	0
0,0833	0,1667	0	0
0,0042	0	0,2500	0,0042
0	0	0,0042	0,0833

Berdasarkan contoh diatas, maka langkah-langkah yang dilakukan pada metode GLCM adalah:

1. Membentuk matriks awal GLCM berdasarkan pasangan 2 piksel yang berjajar sesuai dengan arah 0° , 45° , 90° , dan 135° .
2. Menjumlahkan matriks awal GLCM dengan nilai transposenya, sehingga menghasilkan matriks yang simetris.
3. Normalisasi matriks GLCM dengan membagi tiap elemen matriks dengan jumlah pasangan piksel.

4. Melakukan ekstraksi ciri, misalnya: *Contrast, Homogeneity, Energy, dan Entropy.*

2.3.8 Klasifikasi

Dalam terminologi pembelajaran mesin, klasifikasi dianggap sebagai salah satu metode pembelajaran, dimana kelompok data pelatihan yang diidentifikasi benar-benar tersedia (Alpaydin, 2010). Klasifikasi terdiri dari prediksi sebuah hasil tertentu berdasarkan input yang diberikan. Untuk memprediksi kemungkinan hasil tersebut, algoritma memproses data training yang berisi sebuah data atribut dan hasil masing-masing, biasanya disebut *goal* atau atribut prediksi. Algoritma mencoba untuk menemukan hubungan antar atribut yang mungkin digunakan untuk memprediksi hasil. Berikutnya algoritma yang diberikan kelompok data yang tidak terlihat disebut *prediction set*, yang berisi atribut yang sama, kecuali untuk atribut prediksi yang belum diketahui. Algoritma menganalisis input dan menghasilkan sebuah prediksi. Tingkat keakuratan dari prediksi tersebut menggambarkan seberapa baik algoritma itu. Contoh: pada database medis, set training akan memiliki rekaman informasi pasien yang relevan sebelumnya, dimana atribut prediksi adalah apakah pasien memiliki masalah jantung atau tidak.

Berdasarkan konferensi internasional tentang data mining, IEEE ICDM pada tanggal 22 desember 2006, telah disepakati bahwa terdapat 10 besar algoritma data mining yang dapat digunakan untuk menyelesaikan permasalahan klasifikasi. Adapun ke 10 algoritma tersebut adalah C4.5, K-Means, SVM, Apriori, EM Algorithm, PageRank, K-Nearest Neighbors, Naïve Bayes, CART, dan Adaboost (Wu, Xindong., Kumar, Vipin., et all, 2006).

2.3.9. K-Nearest Neighbors

Dalam pengenalan pola, algoritma *K-Nearest Neighbors* (KNN) merupakan metode non parametrik yang digunakan untuk klasifikasi dan regresi (Altman,1992). KNN merupakan pembelajaran berbasis *instance* atau *lazy learning*, dimana fungsi hanya didekati secara lokal dan semua perhitungan di tunda hingga proses klasifikasi. Algoritma KNN dapat digunakan untuk memberikan bobot untuk kontribusi tiap *neighbors*, sehingga kontribusi *neighbors* terdekat lebih besar daripada *neighbor* yang jauh.

KNN melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Data pembelajaran diproyeksikan pada k ruang berdimensi banyak, dimana masing-masing dimensi merepresentasikan fitur dari data. Ruang ini dibagi menjadi bagian-bagian berdasarkan klasifikasi data pembelajaran. Sebuah titik pada kelas c , jika kelas c merupakan klasifikasi yang paling banyak ditemui pada k buah tetangga terdekat titik tersebut. Algoritma KNN memiliki kelebihan yaitu pelatihan sangat cepat, sederhana dan mudah dipelajari, tahan terhadap data pelatihan yang memiliki derau, dan efektif terhadap data pelatihan yang besar (Suguna N, 2010).

Penentuan nilai k yang baik bergantung pada data yang digunakan. Umumnya, nilai k yang lebih besar mengurangi efek *noise* pada klasifikasi, namun menyebabkan batasan antar kelas sedikit berbeda. Jarak antar data yang digunakan dalam algoritma KNN dapat dihitung menggunakan rumus jarak *Euclidean Distance* dan *Manhattan Distance*. Rumus jarak yang biasa digunakan untuk KNN adalah *Euclidean Distance*, yang merepresentasikan cara berpikir manusia tentang

jarak pada kehidupan nyata (Larose, 2005). *Euclidean Distance* digunakan karena sesuai dengan karakteristik data yang *continue* dan memiliki dimensi yang jelas dan relevan. Untuk mencari nilai *euclidean* dapat digunakan persamaan 15.

$$d_{\text{Euclidean}}(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{k=1}^n (\mathbf{x}_k - \mathbf{y}_k)^2} \quad (15)$$

Dimana x_k dan y_k merupakan atribut ke- k dari x dan y secara berturut - turut.

2.3.10. Pengujian Sistem

Salah satu pengujian algoritma dilakukan dengan menggunakan *confusion matrix*. *Confusion matrix* merupakan salah satu metode yang digunakan untuk mengukur kinerja suatu metode klasifikasi. *Confusion matrix* memiliki 4 istilah sebagai representasi hasil klasifikasi yaitu, *True Positive (TP)*, *True Negative (TN)*, *False Positive (FP)* dan *False Negative*. Nilai *True Positive (TP)* merupakan data positif yang terdeteksi benar. Nilai *True Negative (TN)* merupakan data positif yang terdeteksi salah. *False Positive (TP)* merupakan data negatif yang terdeteksi benar, dan *False Negative (FP)* merupakan data negative yang terdeteksi salah. Ilustrasi diatas dapat dilihat pada tabel 2.4.

Tabel 2.4. Contoh Confusion Matrix

Kelas	Klasifikasi Positif	Klasifikasi Negatif
Positif	TP (True Positif)	FP (False Positif)
Negatif	TN (True Negatif)	FN (False Negatif)

Berdasarkan tabel 2.4. diatas, nantinya akan dapat diperoleh nilai *precision*, *recall*, *accuracy*, *true positive rate*, dan *false positive rate*. Nilai *accuracy* menggambarkan seberapa akurasi tingkat klasifikasi yang dihasilkan. Nilai *recall*

menunjukkan berapa persen data kategori positif yang terklasifikasikan dengan benar oleh sistem. Nilai *precision* menggambarkan jumlah data positif yang diklasifikasikan secara benar dengan total data yang diklasifikasi positif. Nilai *true positive rate* adalah suatu kondisi dimana suatu model secara benar mengklasifikasi data positif. Sedangkan *false positive rate* adalah suatu kondisi ketika suatu model secara tidak tepat mengklasifikasi data positif. Persamaan *accuracy*, *precision*, *recall*, *FP Rate*, dan *TP Rate* dapat dilihat pada persamaan 16, 17, 18, 19, dan 20.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \quad (16)$$

$$\text{Precision} = \frac{TP}{TP+FP} \times 100\% \quad (17)$$

$$\text{Recall} = \frac{TP}{TP+FN} \times 100\% \quad (18)$$

$$\text{FP Rate} = \frac{FP}{FP+TN} \times 100\% \quad (19)$$

$$\text{TP Rate} = \frac{TP}{TP+FN} \times 100\% \quad (20)$$

BAB III

METODE PENELITIAN

3.1. Jenis, Sifat dan Pendekatan Penelitian

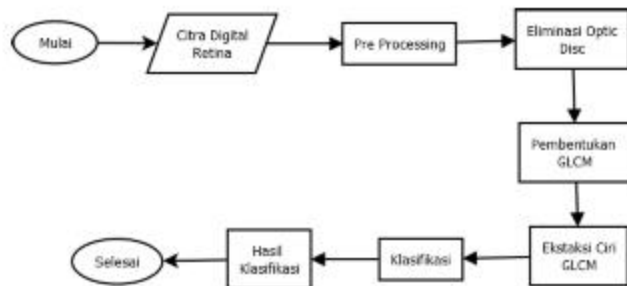
Penelitian yang akan diusulkan berjenis eksperimen. Penelitian dilakukan menggunakan data citra digital retina. Citra digital retina diambil menggunakan kamera fundus. Citra digital yang didapatkan kemudian diolah menggunakan metode GLCM untuk mengekstraksi fitur-fitur citra yang digunakan pada proses klasifikasi.

Penelitian ini bersifat deskriptif yaitu menjelaskan bagaimana rumusan masalah yang digunakan pada penelitian ini dapat terselesaikan.

Penelitian ini menggunakan pendekatan kuantitatif berupa citra digital retina yang telah melalui tahap pre-processing dan menghasilkan matriks numerik.

3.2. Metode Analisis Data

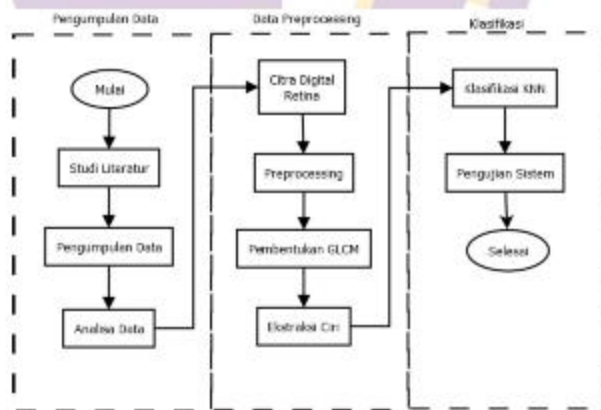
Metode analisis data yang digunakan adalah analisa kuantitatif. Proses klasifikasi citra yang dilakukan menghasilkan 4 kelas yaitu: kelas normal, kelas *Non-Proriferative Diabetic Retinopathy* (Mild), kelas *Proliferative Diabetic Retinopathy* (Moderate), dan kelas Macular Edema (Severe). Data citra retina yang digunakan untuk klasifikasi akan melalui tahap *pre-processing*. Tahap pre-processing bertujuan untuk menghilangkan *noise* pada gambar dan merubah menjadi *grayscale*. Data hasil preprocessing kemudian akan memasuki tahap ekstraksi ciri untuk mendapatkan fitur-fitur citra yang digunakan dalam klasifikasi. Flowchart analisis data dapat dilihat pada gambar 3.1.



Gambar 3.1. Analisis Data

3.3. Alur Penelitian

Proses pelaksanaan penelitian ini terbagi menjadi tiga bagian, yaitu Pengumpulan data, Ekstraksi Ciri, dan Klasifikasi. Pengumpulan data dilakukan secara manual dengan menggunakan dataset *Messidor*. Ekstraksi ciri dilakukan untuk mendapatkan fitur-fitur gambar yang digunakan untuk klasifikasi. Metode yang digunakan pada tahap ekstraksi ciri adalah *Grey Level Co-Occurrence*. Metode klasifikasi yang digunakan adalah *K-Nearest Neighbors*. Alur penelitian yang digunakan dapat dilihat pada gambar 3.2.



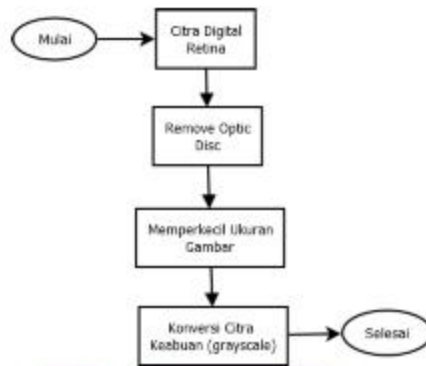
Gambar 3.2. Alur Penelitian

3.4. Metode Pengumpulan Data

Dataset yang digunakan pada penelitian ini berasal dari database *Messidor*. *Messidor* merupakan sebuah program penelitian yang dibiayai oleh Departemen Penelitian dan Pertahanan Prancis pada tahun 2004 (Anonim, *adcis.net*, 2004). *Messidor* menyediakan dataset yang dapat di gunakan untuk tujuan penelitian khususnya penelitian dengan topik Retinopati Diabetes. Data yang digunakan berupa citra digital retina yang diambil menggunakan kamera fundus 3CCD color video Topcon TRC NW6 Non-mydratic Retinograph dengan sudut pandang 45 derajat. Total data yang digunakan sebanyak 1200 data. Data yang berhasil dikumpulkan kemudian akan melalui tahap preprocessing dan ekstraksi ciri sebelum kemudian dilakukan klasifikasi.

3.5. Data Preprocessing

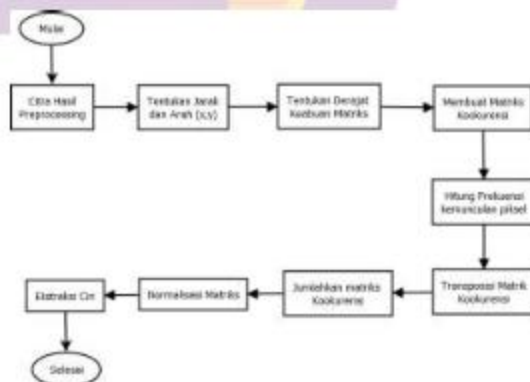
Tahap preprocessing dilakukan dengan tujuan untuk memperbaiki kualitas citra dan membentuk citra sesuai dengan yang dibutuhkan. Tahap ini dimulai dengan menghilangkan optic disc bertujuan untuk mendapatkan bagian citra yang akan digunakan untuk klasifikasi. Proses selanjutnya adalah melakukan reduksi ukuran gambar bertujuan untuk meringankan proses komputasi yang akan dilakukan. Proses terakhir adalah konversi citra ke dalam *grayscale* (keabuan). Hasil akhir dari tahap ini berupa gambar dengan citra *grayscale*. Citra ini kemudian akan masuk tahap ekstraksi ciri menggunakan metode GLCM. Alur *pre-processing data* dapat dilihat pada gambar 3.3.



Gambar 3.3. Data Preprocessing

3.6. Ekstraksi Ciri

Tahap ekstraksi ciri dilakukan menggunakan metode *Grey Level Co-Occurrence (GLCM)*. Input yang digunakan pada tahap ini berasal dari hasil pre-processing sebelumnya. Hasil yang didapatkan pada tahap ini berupa fitur-fitur citra yaitu korelasi, energy, entropy, kontras, dan homogenitas. Fitur ini kemudian akan digunakan sebagai inputan untuk tahap klasifikasi. Alur dari metode GLCM dapat dilihat pada gambar 3.4.



Gambar 3.4. Alur Pembentukan Matriks Kookurensi (GLCM)

3.7. Klasifikasi

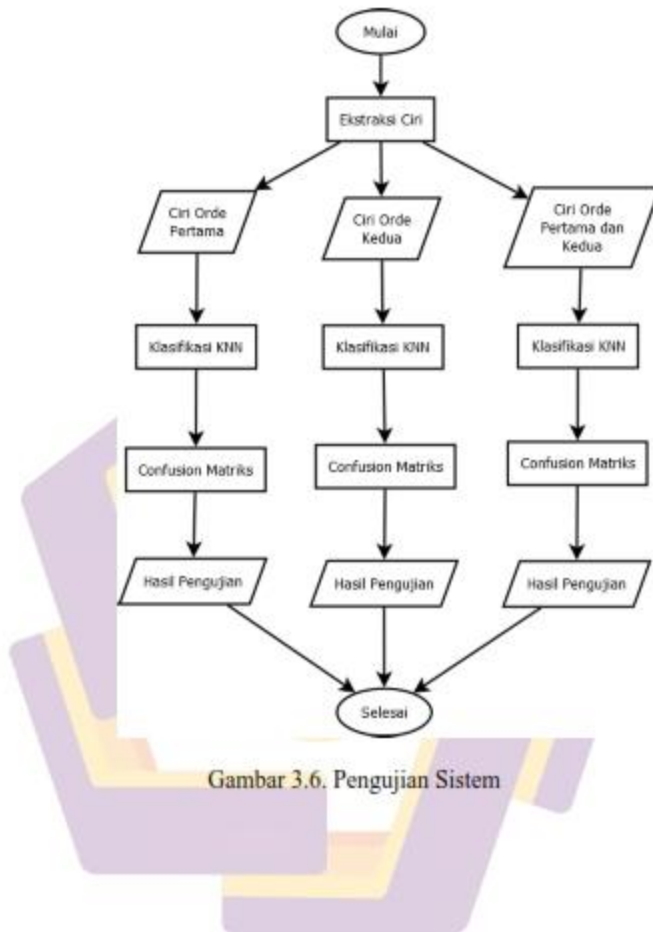
Proses klasifikasi dilakukan menggunakan metode *K-Nearest Neighbor* (KNN). Inputan yang digunakan pada tahap ini merupakan fitur-fitur citra yang telah didapatkan pada tahap ekstraksi ciri. Klasifikasi yang dilakukan akan membagi data kedalam 4 kelas yaitu normal, NPDR, PDR, dan ME. Alur klasifikasi metode KNN dapat dilihat pada gambar 3.5.



Gambar 3.5. Klasifikasi KNN

3.8. Pengujian Sistem

Pengujian algoritma dilakukan dengan menggunakan *confusion matrix*, dengan melakukan 2 skenario percobaan menggunakan teknik *cross-validation* dan *percentage split*. Skenario pertama akan dilakukan pengujian hasil klasifikasi berdasarkan ciri statistik orde pertama. Skenario kedua akan dilakukan pengujian hasil klasifikasi citra berdasarkan ciri statistik orde kedua. Pengujian dilakukan menggunakan teknik *cross validation* dan metode split dengan proporsi 60% data training dan 40% data testing. Alur dari pengujian sistem dapat dilihat pada gambar 3.6.



Gambar 3.6. Pengujian Sistem

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

4.1. Deskripsi Data

Data yang digunakan pada penelitian ini didapatkan dari database *Messidor*. *Messidor* merupakan sebuah program penelitian yang dibiayai oleh Departemen Penelitian dan Pertahanan Prancis pada tahun 2004 (Anonim, *adcis.net*, 2004). *Messidor* menyediakan dataset yang dapat di gunakan untuk tujuan penelitian khususnya penelitian dengan topik Retinopati Diabetes. Data yang digunakan berupa citra digital retina yang diambil menggunakan kamera fundus 3 CCD color video Topcon TRC NW6 Non-mydratic Retinograph dengan sudut pandang 45 derajat. Total data yang digunakan sebanyak 600 data.

Data yang digunakan terbagi menjadi 4 kelas yaitu Normal, Non-Poliferative Diabetic (NPDR) atau Mild, Poliferative (PDR) atau Moderate, dan Mocular Edema (ME) atau Severe. Setiap gambar memiliki resolusi sebesar 2440x1880 piksel. Jumlah sebaran data tiap kelas dapat dilihat pada tabel 4.1.

Tabel 4.1. Sebaran Data

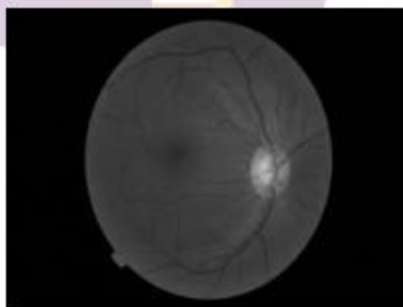
No	Kelas	Jumlah Gambar
1	Normal	150
2	Mild	150
3	Moderate	150
4	Severe	150

4.2. Preprocessing Data

Data yang telah dikumpulkan kemudian akan melalui tahap preprocessing dan ekstraksi ciri sebelum kemudian dilakukan klasifikasi. Tahap preproses ini bertujuan untuk merubah data mentah, menjadi data yang dapat digunakan untuk klasifikasi. Preprocessing data terdiri dari beberapa tahapan, yaitu memperkecil ukuran gambar, merubah gambar menjadi citra keabuan dan menghilangkan optikal disk.

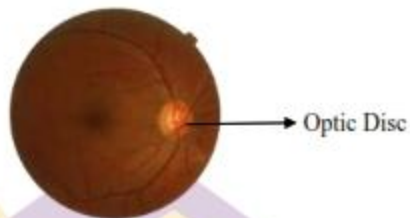
Proses pertama adalah memperkecil ukuran gambar. Data gambar retina yang digunakan memiliki ukuran yang cukup besar. Oleh karena itu, untuk meringankan beban komputasi maka gambar retina akan diperkecil resolusinya, tanpa mengurangi kualitas dan informasi yang terkandung dalam gambar.

Proses selanjutnya adalah merubah gambar dalam bentuk derajat keabuan (*grayscale*). Perubahan ini bertujuan untuk memperoleh ciri-ciri statistik yang dimiliki gambar. Proses konversi dilakukan dengan bantuan library OpenCV. Hasil dari proses konversi keabuan dapat dilihat pada gambar 4.1.



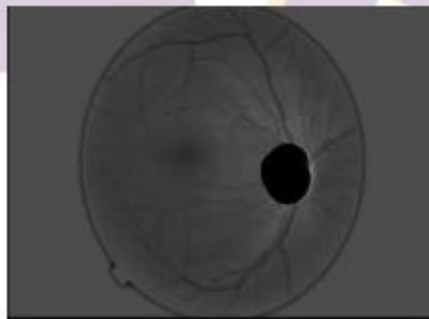
Gambar 4.1. Hasil Konversi Citra *Grayscale*

Tahap preproses data yang terakhir adalah menghapus optikal disk. Tahap ini bertujuan untuk menghilangkan objek yang tidak terpakai ketika perhitungan ekstraksi ciri. Contoh optik disk dapat dilihat pada gambar 4.2.



Gambar 4.2. Optic Disc

Proses penghapusan optik disk diawali dengan mendeteksi ada dan tidaknya kontur dalam image. Jika program mendeteksi adanya kontur berbentuk lingkaran (bola) dalam gambar, maka kontur tersebut akan dihilangkan menggunakan *masking*. Masking adalah sebuah cara menghilangkan sebuah objek pada gambar dengan menimpa (menutupi) objek tersebut. Hasil akhir tahap preprocessing data dapat dilihat pada gambar 4.3.



Gambar 4.3. Remove Optic Disc

4.3. Ekstraksi Ciri

Proses ekstraksi ciri dilakukan untuk mendapatkan informasi-informasi yang terkandung pada sebuah gambar. Informasi ini digunakan sebagai inputan pada proses klasifikasi. Proses ekstraksi dilakukan menggunakan matriks co-okurensi dengan metode GLCM. GLCM mampu menentukan ciri dalam merepresentasikan dalam merepresentasikan karakteristik tekstur sebuah citra (Siquera, 2013). Fitur citra yang diambil adalah, *kontras, korelasi, homogenitas, energi, angular second moment (asm), dan dissimilarity*. Fitur ini diambil berdasarkan dokumentasi library scikit-image image processing python. Fitur tersebut akan dihitung berdasarkan sudut 0, 45, 90, dan 135 derajat.

- **Ekstraksi ciri Gray Level Co-occurrence Matrix**

Ekstraksi ciri orde kedua dilakukan menggunakan metode Grey Level Co-occurrence Matrix (GLCM). Matriks ini dibuat dengan mengidentifikasi jumlah hubungan antara piksel gambar. Contoh membuat matriks ko-okurensi adalah sebagai berikut.

Diberikan contoh sebuah matriks A merupakan representasi dari sebuah gambar yang memiliki derajat keabuan 0-4. Matriks tersebut akan diubah menjadi matriks kookurensi berdasarkan aturan pada tabel 2.

4	0	3	2	1
0	2	4	2	0
0	3	0	4	1
1	1	2	3	0
2	4	1	0	2

Gambar 4.4. Matriks A

Kemudian akan dicari hubungan antar piksel pada sudut 0 derajat dan jarak ($d = 1$). Maka hasilnya adalah sebagai berikut

	0	1	2	3	4
0	0	0	2	2	1
1	1	1	1	0	0
2	1	1	0	1	1
3	2	0	1	0	0
4	1	2	1	0	0

Gambar 4.5. Hubungan Antar piksel

Berdasarkan gambar diatas jumlah hubungan antara piksel (0,0) adalah 0. Jumlah hubungan piksel (0,1) adalah 0, begitu seterusnya. Langkah berikutnya adalah dengan mengalikan matriks koonkurensi dengan inversnya.

$$\begin{bmatrix} 0 & 0 & 2 & 2 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 2 & 0 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 1 & 2 & 1 \\ 0 & 1 & 1 & 0 & 2 \\ 2 & 1 & 0 & 1 & 1 \\ 2 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 3 & 4 & 2 \\ 1 & 2 & 2 & 0 & 2 \\ 3 & 2 & 0 & 2 & 2 \\ 4 & 0 & 2 & 0 & 0 \\ 2 & 2 & 2 & 0 & 0 \end{bmatrix}$$

Langkah terakhir adalah normalisasi matriks tersebut. Hasil matriks normalisasi dapat dilihat pada tabel 4.2.

Tabel 4.2. Matriks Normalisasi

0	0,0263	0,0789	0,1052	0,0526
0,0263	0,0526	0,0526	0	0,0526
0,0789	0,0526	0	0,0526	0,0526
0,1053	0	0,0526	0	0
0,0526	0,0526	0,0526	0	0

Matriks yang sudah ternormalisasi kemudian akan digunakan untuk mencari nilai ekstraksi ciri orde kedua. Nilai 0 tidak diikutkan dengan dalam perhitungan.

Untuk memudahkan perhitungan akan dibuat sebuah tabel yang berisi nilai piksel yang digunakan.

Tabel 4.3. Nilai piksel matriks GLCM

GLCM	Nilai	GLCM	Nilai	GLCM	Nilai	GLCM	Nilai	GLCM	Nilai
(1,2)	0,0263	(2,1)	0,0263	(3,1)	0,0789	(4,1)	0,1053	(5,3)	0,0526
(1,3)	0,0789	(2,2)	0,0526	(3,2)	0,0526	(4,3)	0,0526		
(1,4)	0,1052	(2,3)	0,0526	(3,4)	0,0526	(5,1)	0,0526		
(1,5)	0,0526	(2,5)	0,0526	(3,5)	0,0526	(5,2)	0,0526		

Berikut proses perhitungan nilai kontras, angular second moment, korelasi, entropi dan hoogenitas:

a. Kontras

$$\begin{aligned}
 \text{Kontras} &= \sum_{i_1} \sum_{i_2} (i_1 - i_2)^2 p(i_1, i_2) \\
 &= \{(1-2)^2 p(1,2) + (1-3)^2 p(1,3) + (1-4)^2 p(1,4) + (1-5)^2 p(1,5) + (2-1)^2 p(2,1) + \\
 &\quad (2-2)^2 p(2,2) + (2-3)^2 p(2,3) + (2-5)^2 p(2,5) + (3-1)^2 p(3,1) + (3-2)^2 p(3,2) + (3- \\
 &\quad 4)^2 p(3,4) + (3-5)^2 p(3,5) + (4-1)^2 p(4,1) + (4-3)^2 p(4,3) + (5-1)^2 p(5,1) + (5-2)^2 \\
 &\quad p(5,2) + (5-3)^2 p(5,3)\} \\
 &= \{0,0263 + 4*0,0789 + 9*0,1052 + 16*0,0526 + 0,0263 + 0 + 0,0526 + \\
 &\quad 9*0,0526 + 4*0,0789 + 0,0526 + 0,0526 + 4*0,0526 + 9*0,1053 + 0,0526 + \\
 &\quad 9*0,0526 + 4*0,0526 + 4*0,0526\} \\
 &= 5,8395
 \end{aligned}$$

b. Angular Second Moment

$$\text{ASM} = \sum_{i_1} \sum_{i_2} p^2(i_1, i_2)$$

$$\begin{aligned}
&= \{ p(1,2)^2 + p(1,3)^2 + p(1,4)^2 + p(1,5)^2 + p(2,1)^2 + p(2,2)^2 + p(2,3)^2 + p(2,5)^2 \\
&+ p(3,1)^2 + p(3,2)^2 + p(3,4)^2 + p(3,5)^2 + p(4,1)^2 + p(4,3)^2 + p(5,1)^2 + p(5,2)^2 + \\
&p(5,3)^2 \} \\
&= \{ (0,0263)^2 + (0,0789)^2 + (0,1053)^2 + (0,0526)^2 + (0,0263)^2 + (0,0526)^2 + \\
&(0,0526)^2 + (0,0526)^2 + (0,0526)^2 + (0,0789)^2 + (0,0526)^2 + (0,0526)^2 + \\
&(0,0526)^2 + (0,1053)^2 + (0,0526)^2 + (0,0526)^2 + (0,0526)^2 + (0,0526)^2 \} \\
&= 0,06648
\end{aligned}$$

c. Entropi

$$\begin{aligned}
\text{Entropi} &= - \sum_{i_1} \sum_{i_2} p(i_1, i_2) \log p(i_1, i_2) \\
&= - \{ p(1,2) \log p(1,2) + p(1,3) \log p(1,3) + p(1,4) \log p(1,4) + p(1,5) \log p(1,5) \\
&+ p(2,1) \log p(2,1) + p(2,2) \log p(2,2) + p(2,3) \log p(2,3) + p(2,5) \log p(2,5) + \\
&p(3,1) \log p(3,1) + p(3,2) \log p(3,2) + p(3,4) \log p(3,4) + p(3,5) \log p(3,5) + \\
&p(4,1) \log p(4,1) + p(4,3) \log p(4,3) + p(5,1) \log p(5,1) + p(5,2) \log p(5,2) + \\
&p(5,3) \log p(5,3) \} \\
&= - \{ 0,0263 * \log(0,0263) + 0,0789 * \log(0,0789) + 0,1053 * \log(0,1053) + \\
&0,0526 * \log(0,0526) + 0,0263 * \log(0,0263) + 0,0526 * \log(0,0526) + \\
&0,0526 * \log(0,0526) + 0,0526 * \log(0,0526) + 0,0789 * \log(0,0789) + \\
&0,0526 * \log(0,0526) + 0,0526 * \log(0,0526) + 0,0526 * \log(0,0526) + \\
&0,1053 * \log(0,1053) + 0,0526 * \log(0,0526) + 0,0526 * \log(0,0526) + \\
&0,0526 * \log(0,0526) + 0,0526 * \log(0,0526) \} \\
&= 1,20342
\end{aligned}$$

d. Homogenitas

$$\text{Homogenitas} = \sum_{i_1} \sum_{i_2} \frac{p(i_1, i_2)}{1 + |i_1 - i_2|}$$

$$\begin{aligned}
&= \{p(1,2)/1+|1-2| + p(1,3)/1+|1-3| + p(1,4)/1+|1-4| + p(1,5)/1+|1-5| + \\
&p(2,1)/1+|2-1| + p(2,2)/1+|2-2| + p(2,3)/1+|2-3| + p(2,5)/1+|2-5| + p(3,1)/1+|3- \\
&1| + p(3,2)/1+|3-2| + p(3,4)/1+|3-4| + p(3,5)/1+|3-5| + p(4,1)/1+|4-1| + \\
&p(4,3)/1+|4-3| + p(5,1)/1+|5-1| + p(5,2)/1+|5-2| + p(5,3)/1+|5-3| \\
&= 0,0263/2 + 0,0789/3 + 0,1053/4 + 0,0526/5 + 0,0263/2 + 0,0526/1 + 0,0526/2 \\
&+ 0,0526/4 + 0,0789/3 + 0,0526/2 + 0,0526/2 + 0,0526/3 + 0,1053/4 + 0,0526/2 \\
&+ 0,0526/5 + 0,0526/4 + 0,0526/3 \\
&= 0,37173
\end{aligned}$$

e. Korelasi

Untuk menghitung nilai korelasi, terlebih dulu akan dicari nilai dari μ_i' , μ_j' , σ_i' , σ_j' . μ_i' dan μ_j' merupakan nilai mean dari i dan j. Sedangkan σ_i' dan σ_j' merupakan nilai standard deviasi dari i dan j.

$$\begin{aligned}
\mu_i' &= \sum_{i_1} \sum_{i_2} i * p(i, j) \quad \text{nilai Mean piksel } i \\
&= \{(1*0,0263) + (1*0,0789) + (1*0,1053) + (1*0,0526) + (2*0,0263) + \\
&(2*0,0526) + (2*0,0526) + (2*0,0526) + (3*0,0789) + (3*0,0526) + (3*0,0526) \\
&+ (3*0,0526) + (4*0,1053) + (4*0,0526) + (5*0,0526) + (5*0,0526) + \\
&(5*0,0526)\} \\
&= 2,7619
\end{aligned}$$

$$\begin{aligned}
\sigma_i' &= \sum_{i_1} \sum_{i_2} p(i, j)(i - \mu_i')^2 \quad \text{nilai Standard Deviasi piksel } i \\
&= \{(0,0263*(1-2,7619)^2 + (0,0789*(1-2,7619)^2 + (0,1053*(1-2,7619)^2 + \\
&(0,0526*(1-2,7619)^2 + (0,0263*(2-2,7619)^2 + (0,0526*(2-2,7619)^2 + \\
&(0,0526*(2-2,7619)^2 + (0,0526*(2-2,7619)^2 + (0,0789*(3-2,7619)^2 + \\
&(0,0526*(3-2,7619)^2 + (0,0526*(3-2,7619)^2 + (0,0526*(3-2,7619)^2 +
\end{aligned}$$

$$\begin{aligned}
 & (0,1053*(4-2,7619)^2 + (0,0526*(4-2,7619)^2 + (0,0526*(5-2,7619)^2 + \\
 & (0,0526*(5-2,7619)^2 + (0,0526*(5-2,7619)^2 \\
 & = 1,9692
 \end{aligned}$$

$$\begin{aligned}
 \mu_{j'} &= \sum_{i_1} \sum_{i_2} j * p(i, j) \quad \text{nilai Mean piksel } j \\
 &= \{(2*0,0263) + (3*0,0789) + (4*0,1053) + (5*0,0526) + (1*0,0263) + \\
 & (2*0,0526) + (3*0,0526) + (5*0,0526) + (1*0,0789) + (2*0,0526) + (4*0,0526) \\
 & + (5*0,0526) + (1*0,1053) + (3*0,0526) + (1*0,0526) + (2*0,0526) + \\
 & (3*0,0526)\} \\
 &= 27616
 \end{aligned}$$

$$\begin{aligned}
 \sigma_{j'} &= \sum_{i_1} \sum_{i_2} p(i, j)(j - \mu_{j'})^2 \quad \text{nilai Standard Deviasi piksel } i \\
 &= \{(0,0263*(2-2,7616)^2 + (0,0789*(3-2,7616)^2 + (0,1053*(4-2,7616)^2 + \\
 & (0,0526*(5-2,7616)^2 + (0,0263*(1-2,7616)^2 + (0,0526*(2-2,7616)^2 + \\
 & (0,0526*(3-2,7616)^2 + (0,0526*(5-2,7616)^2 + (0,0789*(1-2,7616)^2 + \\
 & (0,0526*(2-2,7616)^2 + (0,0526*(4-2,7616)^2 + (0,0526*(5-2,7616)^2 + \\
 & (0,1053*(1-2,7616)^2 + (0,0526*(3-2,7616)^2 + (0,0526*(1-2,7616)^2 + \\
 & (0,0526*(2-2,7616)^2 + (0,0526*(3-2,7616)^2) \\
 & = 1,9694
 \end{aligned}$$

$$\begin{aligned}
 \text{Korelasi} &= \sum_{i_1} \sum_{i_2} \frac{(i - \mu_{i'}) * (j - \mu_{j'}) * p(i, j)}{\sigma_{i'} * \sigma_{j'}} \\
 &= \{((1-2,7619)*(2-2,7616)*0,0263)/(1,9692*1,9694) + ((1-2,7619)*(3- \\
 & 2,7616)*0,0789)/(1,9692*1,9694) + ((1-2,7619)*(4- \\
 & 2,7616)*0,1052)/(1,9692*1,9694) + ((1-2,7619)*(5- \\
 & 2,7616)*0,0526)/(1,9692*1,9694) + ((2-2,7619)*(1- \\
 & 2,7616)*0,0526)/(1,9692*1,9694) + ((2-2,7619)*(2-2,7616)*0,0526)/(1,9692*1,9694) \\
 & + ((2-2,7619)*(3-2,7616)*0,0526)/(1,9692*1,9694) + ((2-2,7619)*(4-2,7616)*0,0526)/(1,9692*1,9694) \\
 & + ((2-2,7619)*(5-2,7616)*0,0526)/(1,9692*1,9694) + ((3-2,7616)*(1-2,7616)*0,0789)/(1,9692*1,9694) \\
 & + ((3-2,7616)*(2-2,7616)*0,0789)/(1,9692*1,9694) + ((3-2,7616)*(3-2,7616)*0,0789)/(1,9692*1,9694) \\
 & + ((3-2,7616)*(4-2,7616)*0,0789)/(1,9692*1,9694) + ((3-2,7616)*(5-2,7616)*0,0789)/(1,9692*1,9694) \\
 & + ((4-2,7616)*(1-2,7616)*0,1052)/(1,9692*1,9694) + ((4-2,7616)*(2-2,7616)*0,1052)/(1,9692*1,9694) \\
 & + ((4-2,7616)*(3-2,7616)*0,1052)/(1,9692*1,9694) + ((4-2,7616)*(4-2,7616)*0,1052)/(1,9692*1,9694) \\
 & + ((4-2,7616)*(5-2,7616)*0,1052)/(1,9692*1,9694) + ((5-2,7616)*(1-2,7616)*0,0526)/(1,9692*1,9694) \\
 & + ((5-2,7616)*(2-2,7616)*0,0526)/(1,9692*1,9694) + ((5-2,7616)*(3-2,7616)*0,0526)/(1,9692*1,9694) \\
 & + ((5-2,7616)*(4-2,7616)*0,0526)/(1,9692*1,9694) + ((5-2,7616)*(5-2,7616)*0,0526)/(1,9692*1,9694) \\
 & = 0,0000
 \end{aligned}$$

$$\begin{aligned}
& 2,7616) * 0,0263) / (1,9692 * 1,9694) & + & ((2 - 2,7619) * (2 - \\
& 2,7616) * 0,0526) / (1,9692 * 1,9694) & + & ((2 - 2,7619) * (3 - \\
& 2,7616) * 0,0526) / (1,9692 * 1,9694) & + & ((2 - 2,7619) * (5 - \\
& 2,7616) * 0,0526) / (1,9692 * 1,9694) & + & ((3 - 2,7619) * (1 - \\
& 2,7616) * 0,0789) / (1,9692 * 1,9694) & + & ((3 - 2,7619) * (2 - \\
& 2,7616) * 0,0526) / (1,9692 * 1,9694) & + & ((3 - 2,7619) * (4 - \\
& 2,7616) * 0,0526) / (1,9692 * 1,9694) & + & ((3 - 2,7619) * (5 - \\
& 2,7616) * 0,0526) / (1,9692 * 1,9694) & + & ((4 - 2,7619) * (1 - \\
& 2,7616) * 0,1053) / (1,9692 * 1,9694) & + & ((4 - 2,7619) * (3 - \\
& 2,7616) * 0,0526) / (1,9692 * 1,9694) & + & ((5 - 2,7619) * (1 - \\
& 2,7616) * 0,0526) / (1,9692 * 1,9694) & + & ((5 - 2,7619) * (2 - \\
& 2,7616) * 0,0526) / (1,9692 * 1,9694) & + & ((5 - 2,7619) * (3 - \\
& 2,7616) * 0,0526) / (1,9692 * 1,9694) & & \\
& = -0,251
\end{aligned}$$

Berdasarkan hasil hitungan diatas, maka hasil akhir ekstraksi ciri matriks A pada gambar 21 adalah

Kontras = 5,8395

Homogenitas/Inverse Difference Moment = 0,37173

Energy/Angular Second Moment = 0,06648

Entropi = 1,203419

Korelasi = -0,2451

Fitur gambar yang digunakan pada penelitian ini adalah kontras, korelasi, homogenitas, energi, angular second moment, dan dissimilarity. Masing-masing

metric fitur diukur berdasarkan sudut 0, 45, 90, 135. Sehingga akan didapatkan 24 fitur yang digunakan, seperti yang dapat dilihat pada tabel 4.4.

Tabel 4.4. Fitur GLCM

No	Fitur	No	Fitur
1	Kontras_0	13	Energi_0
2	Kontras_45	14	Energi_45
3	Kontras_90	15	Energi_90
4	Kontras_135	16	Energi_135
5	Korelasi_0	17	Asm_0
6	Korelasi_45	18	Asm_45
7	Korelasi_90	19	Asm_90
8	Korelasi_135	20	Asm_135
9	Homogenitas_0	21	Disimilarity_0
10	Homogenitas_45	22	Disimilarity_45
11	Homogenitas_90	23	Disimilarity_90
12	Homogenitas_135	24	Disimilarity_135

Proses ekstraksi fitur pada penelitian ini dilakukan menggunakan bantuan library dari scikit-image dengan fungsi *greycoprops*. Sedangkan untuk matriks ko-okurensi dilakukan menggunakan fungsi *greycomatrix*. Hasil akhir dari proses ekstraksi ciri ordo kedua dapat dilihat pada tabel 4.5.

Tabel 4.5. Sebagian Hasil Akhir Ekstraksi Ciri Ordo Kedua

No	Nama_Gambar	Kontras_0	Korelasi_0	Homogeneity_0	Energi_0	Dissimilarity_0	ASM_0	Kelas
1	20051019_38557_0100_PP.tif	49,56763	0,984287	0,409287	0,194789	3,041649	0,03794282	severe
2	20051020_43808_0100_PP.tif	15,87857	0,987008	0,485068	0,186686	1,916097	0,03485156	noRD
3	20051020_43832_0100_PP.tif	19,08309	0,985407	0,446316	0,202646	2,258654	0,04106527	mild
4	20051020_43882_0100_PP.tif	43,15062	0,989482	0,418701	0,189464	2,815362	0,03589665	moderate
5	20051020_43906_0100_PP.tif	44,10417	0,989276	0,415543	0,190628	2,969473	0,03633904	severe
...
598	20060530_55390_0100_PP.tif	15,46556	0,986394	0,47681	0,195082	2,064745	0,03805718	mild
599	20060530_55451_0100_PP.tif	49,56763	0,984287	0,409287	0,194789	3,041649	0,03794282	mild
600	20060530_55468_0100_PP.tif	15,87857	0,987008	0,485068	0,186686	1,916097	0,03485156	mild

Untuk hasil lengkap dari proses ekstraksi ciri dapat dilihat pada tautan berikut <https://bit.ly/3faq73y>.

4.3.2 Fitur Selection

Untuk menentukan metrik fitur yang relevan terhadap data, digunakan fitur selection. Terdapat beberapa alasan dalam penambahan metode fitur selection, yaitu : Proses training algoritma menjadi lebih cepat, mengurangi kompleksitas model dan membuat lebih mudah untuk ditafsirkan, membangun model dengan kemampuan prediksi lebih baik, dan memilih set fitur yang tepat.

Metode fitur selection yang digunakan adalah wrapper subset evaluator. Metode wrapper bekerja dengan melakukan evaluasi pada algoritma machine learning untuk menentukan fitur-fitur yang optimal.

Maka fitur citra yang terpilih adalah kontras_45, kontras_90, korelasi_135, dissimilarity_135. Hasil fitur selection dapat dilihat pada gambar 4.6.

```

Search Method:
  Best first.
  Start set: no attributes
  Search direction: forward
  Stale search after 5 node expansions
  Total number of subsets evaluated: 186
  Merit of best subset found: 0.928

Attribute Subset Evaluator (supervised, Class (nominal): 25 kelas):
  Wrapper Subset Evaluator
  Learning scheme: weka.classifiers.Lazy_LBK
  Scheme options: -K 70 -W 0 -A weka.core.neighboursearch.LinearHSSearch -A "weka.c
  Subset evaluation: classification accuracy
  Number of folds for accuracy estimation: 5

Selected attributes: 2,3,8,24 : 4
  kontras_45
  kontras_90
  korelasi_135
  dissimilarity_135
  
```

Gambar 4.6 Hasil fitur selection

4.4. Klasifikasi

Proses klasifikasi pada penelitian ini dilakukan menggunakan algoritma K-Nearest Neighbors (KNN). Data yang digunakan sebagai inputan sistem adalah dataset hasil ekstraksi ciri orde kedua. Klasifikasi dilakukan dengan beberapa skenario percobaan. Skenario percobaan pertama adalah klasifikasi dataset menggunakan teknik cross validation. Skenario percobaan kedua adalah klasifikasi dataset menggunakan teknik split training and testing data. Skenario percobaan ketiga adalah klasifikasi dataset hasil fitur selection menggunakan teknik cross validation dan split. Contoh perhitungan algoritma KNN dilakukan menggunakan dataset ekstraksi ciri orde kedua pada tabel 4.6.

Tabel 4.6. Data Contoh Perhitungan KNN

Gambar	Entropi	Energi	Homogenitas	Kontras	Korelasi	Kelas
1	9.5536	0.0025	0.3517	123.997	0.8403	severe
2	9.2873	0.0032	0.3673	52.4734	0.9085	noRD
3	9.1614	0.0036	0.3331	62.4762	0.9026	Mild
4	10.192	0.0019	0.3528	136.758	0.8965	moderate
5	10.1425	0.0021	0.3537	135.165	0.9044	severe
6	9.6198	0.0026	0.3048	116.312	0.8574	noRD
7	9.2066	0.0036	0.3309	115.636	0.8869	Mild
8	10.3149	0.0016	0.3563	122.663	0.9263	Mild
9	9.3621	0.003	0.4004	45.119	0.9514	Mild
10	9.5971	0.0027	0.2971	97.2967	0.8679	??????

Berdasarkan tabel diatas, maka akan dilakukan proses identifikasi kelas pada gambar nomor 10. Langkah awal proses KNN adalah menentukan nilai k. Nilai k merupakan sebuah variabel yang menentukan banyaknya tetangga terdekat (memiliki nilai terbesar) dengan data yang diklasifikasi. Misalnya ditentukan nilai k = 5, maka berdasarkan hasil perhitungan akan dipilih nilai terkecil sebanyak 5 buah. Proses penentuan nilai jarak antar data dilakukan menggunakan *euclidean distance*. Berikut adalah proses perhitungan jarak antar data:

$$\text{Gambar 10} = \sqrt{(9,5971 - 9,5536)^2 + (0,0027 - 0,0025)^2 + (0,2971 - 0,3517)^2 + (97,2967 - 123,997)^2 + (0,8679 - 0,8403)^2} = 26,70041$$

Dengan cara yang sama maka dihasilkan nilai pada gambar lain.

$$\text{Gambar 2} = 44,82444$$

$$\text{Gambar 6} = 19,01523$$

$$\text{Gambar 3} = 34,82326$$

$$\text{Gambar 7} = 18,3435$$

$$\text{Gambar 4} = 39,46583$$

$$\text{Gambar 8} = 25,37659$$

$$\text{Gambar 5} = 37,87229$$

$$\text{Gambar 9} = 52,1784$$

Setelah mendapat nilai jarak masing-masing gambar, maka akan dipilih 5 data gambar dengan nilai terkecil, yaitu: gambar 7, gambar 6, gambar 8, gambar 1, gambar 3.

Langkah selanjutnya adalah melakukan *vote* pada data yang telah terpilih. Berdasarkan perhitungan jarak *euclidean*, maka 3 gambar dari kelas mild memiliki jarak terdekat dengan data gambar no 10. Maka dari itu, gambar no 10 diklasifikasikan masuk kelas mild.

4.4.1. Klasifikasi menggunakan 4 target kelas

Pada proses klasifikasi ini dilakukan beberapa skenario percobaan. Skenario pertama adalah klasifikasi dataset menggunakan teknik *cross validation*. Skenario kedua adalah klasifikasi dataset menggunakan teknik *split training and testing data*. Skenario ketiga adalah klasifikasi dataset hasil fitur selection menggunakan teknik *cross validation*. Sedangkan skenario keempat adalah klasifikasi dataset hasil fitur selection menggunakan teknik *split*. Kelas yang digunakan sebagai target klasifikasi adalah noRD, mild, moderate, dan severe. Percobaan dilakukan untuk mendapatkan nilai *k* yang paling optimal.

a. Skenario pertama klasifikasi menggunakan *cross validation*

Pada skenario pertama, untuk menentukan akurasi terbaik dilakukan pencarian nilai *k* optimal. Untuk mendapatkan nilai *k* optimal akan dilakukan proses klasifikasi secara berulang hingga mendapat akurasi tertinggi. Pada skenario pertama, klasifikasi dilakukan sebanyak 300 kali atau total data dibagi 2. Untuk jumlah yang digunakan pada *cross validation* adalah 10 buah. Hasil eksperimen

grafik nilai k dan hasil klasifikasi menggunakan cross validation dapat dilihat pada gambar 4.7.



Gambar 4.7 Grafik akurasi cross validation

Berdasarkan grafik diatas, nilai akurasi terbesar yaitu 29,67% dicapai ketika nilai k adalah 211. Sedangkan nilai akurasi terkecil yaitu 20,5% dicapai ketika nilai k adalah 10. Dengan rata-rata akurasi sebesar 26,565%.

Untuk persebaran data hasil klasifikasi pada masing-masing kelas dapat dilihat pada tabel 4.7.

Tabel 4.7 Hasil klasifikasi menggunakan cross validation

	Severe	NoRD	mild	Moderate
Severe	56	35	30	29
NoRD	37	42	41	30
Mild	38	28	48	36
Severe	36	24	58	32

Berdasarkan hasil klasifikasi pada tabel 4.7 maka didapatkan nilai recall sebesar 29,67% dan nilai precision sebesar 29,601%.

b. Skenario kedua klasifikasi menggunakan split percentage

Pada percobaan kedua ini akan dilakukan klasifikasi menggunakan split percentage dengan proporsi 60% data training dan 40% data testing. Percobaan dilakukan sebanyak 240 kali. Hasil percobaan dapat dilihat pada gambar 4.8.



Gambar 4.8 Grafik akurasi teknik split

Berdasarkan grafik diatas, maka akurasi tertinggi didapat ketika nilai k adalah 73 dengan nilai akurasi 31,25%. Sedangkan nilai akurasi terkecil yaitu 22,5% dicapai ketika nilai k adalah 8. Dengan rata-rata akurasi sebesar 27,382%.

Untuk sebaran data hasil klasifikasi dapat dilihat pada tabel 4.8.

Tabel 4.8 Hasil klasifikasi menggunakan teknik split

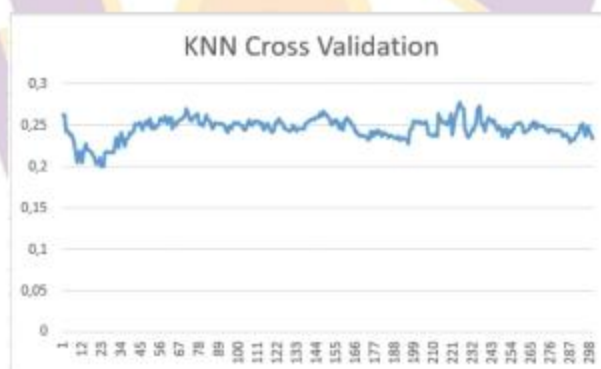
	Severe	NoRD	mild	Moderate
Severe	16	11	12	4
NoRD	10	17	17	6
Mild	7	8	20	3
Severe	12	6	25	6

Berdasarkan hasil klasifikasi pada tabel 4. maka didapatkan nilai recall sebesar 31,25% dan nilai precision sebesar 31,59%.

c. Skenario ketiga klasifikasi dataset fitur selection menggunakan cross validation

Untuk menyeleksi fitur mana yang berpengaruh, dilakukan fitur selection menggunakan wrapper subset evaluator, maka fitur yang terpilih adalah kontras_45, kontras_90, korelasi_135, disimilarity_135.

Setelah dilakukan fitur selection, klasifikasi dilakukan kembali menggunakan fitur yang telah terseleksi. Klasifikasi pertama dilakukan menggunakan teknik cross validation Menghasilkan akurasi sebesar 30,33 % pada $k = 281$. Grafik hasil klasifikasi dapat dilihat pada gambar 4.9.



Gambar 4.9 Grafik Cross Validation dan Fitur Selection
Persebaran data hasil klasifikasi dapat dilihat pada tabel 4.9

Tabel 4.9 Hasil klasifikasi cross validation setelah fitur selection

	severe	noRD	mild	moderate
severe	53	37	21	39
noRD	35	38	35	42
mild	35	30	34	51
moderate	34	21	38	57

Berdasarkan hasil klasifikasi pada tabel 4.9 maka didapatkan nilai recall sebesar 30,33% dan nilai precision sebesar 30,159 %.

d. Skenario keempat klasifikasi dataset fitur selection menggunakan split

Skenario keempat dilakukan klasifikasi dataset fitur selection menggunakan teknik split percentage. Menghasilkan akurasi sebesar 32,083% pada $k = 59$. Grafik hasil klasifikasi dapat dilihat pada gambar 4.10



Gambar 4.10 Grafik Fitur Selection dan Split

Sebaran data hasil klasifikasi dapat dilihat pada tabel 4.10

Tabel 4.10 Hasil klasifikasi teknik split setelah fitur selection

	severe	noRD	mild	moderate
severe	34	4	10	5
noRD	34	11	11	10
mild	27	7	16	10
moderate	31	8	6	16

Berdasarkan hasil klasifikasi pada tabel 4.10 maka didapatkan nilai recall sebesar 30,33% dan nilai precision sebesar 30,159%.

Hasil dari tiap skenario akan dibandingkan untuk menganalisa bagaimana hasil klasifikasi sebelum dan setelah fitur selection. Hasil perbandingan setiap skenario dapat dilihat pada tabel 4.11

Tabel 4.11 Perbandingan klasifikasi cross validation dan data testing

Teknik	Dataset Asli			Fitur Selection		
	Akurasi	Presisi	Recall	Akurasi	Presisi	Recall
Cross Validation	29,67%	29,601%	29,67%	30,33%	30,159%	30,33%
Split	31,25%	31,59%	31,25%	32,08%	32,56%	32,08%

Berdasarkan tabel tersebut, terdapat perbedaan nilai akurasi yang dihasilkan oleh sistem. Akurasi yang dihasilkan menggunakan fitur selection mengalami peningkatan sebesar 0,76% untuk CV dan 0,83% untuk teknik split. Peningkatan yang terjadi disebabkan oleh berkurangnya fitur-fitur yang kurang relevan terhadap data.

4.4.2. Klasifikasi 2 Kelas

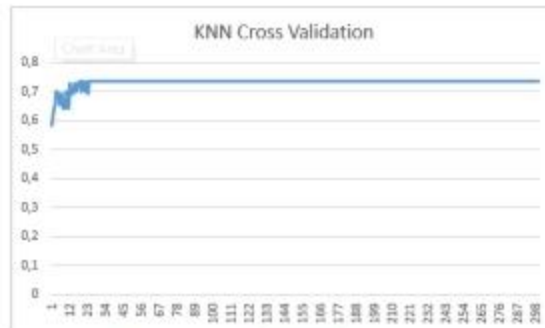
Berdasarkan hasil yang didapat pada percobaan sebelumnya, dapat diketahui bahwa akurasi yang dihasilkan oleh sistem meningkat setelah ditambahkan dengan fitur selection. Untuk memperkuat hasil ini, maka sistem akan diuji coba kembali untuk klasifikasi dengan target kelas yang lebih sedikit (2 kelas).

Tujuan lain dari eksperimen ini adalah untuk mengukur kemampuan sistem dalam mendeteksi pasien sakit berdasarkan citra retina pasien.

Data awal yang digunakan terdapat 4 kelas untuk tingkat keparahan pasien, mulai dari normal, ringan, sedang, parah. Berdasarkan data tersebut, maka data kelas data akan di gabungkan. Data dengan label normal dianggap sebagai pasien dengan mata normal (noRD), sedangkan data dengan label ringan (mild), sedang (moderate), dan parah (severe) akan dianggap sebagai mata sakit. Sama dengan percobaan sebelumnya, klasifikasi dilakukan sebanyak 4 skenario. Skenario pertama klasifikasi menggunakan Cross Validation. Skenario kedua klasifikasi menggunakan Split. Skenario ketiga Cross validation dengan fitur selection. Skenario keempat adalah Split dengan fitur selection.

a. Klasifikasi menggunakan Cross Validation

Klasifikasi dilakukan menggunakan teknik cross validation Menghasilkan akurasi sebesar 73,833% pada $k = 18$. Dengan rata-rata akurasi sebesar 73,45% Grafik hasil klasifikasi dapat dilihat pada gambar 4.11



Gambar 4.11 Klasifikasi 2 kelas Cross Validation

Berdasarkan gambar 4.11, terlihat bahwa nilai akurasi yang dihasilkan mengalami peningkatan dan stabil ketika nilai k mencapai 18. Nilai recall yang dihasilkan sistem adalah 98,2%. Sedangkan nilai precision yang dihasilkan adalah 74,78%.

b. Klasifikasi menggunakan Split Percentage

Klasifikasi dilakukan menggunakan teknik cross validation Menghasilkan akurasi sebesar 73,75% pada $k = 11$. Dengan rata-rata akurasi sebesar 71,98%. Grafik hasil klasifikasi dapat dilihat pada gambar 4.12



Gambar 4.12 Klasifikasi 2 kelas Split

Berdasarkan gambar 4.12, terlihat bahwa nilai akurasi yang dihasilkan mengalami peningkatan dan stabil ketika nilai k mencapai 11. Nilai recall yang dihasilkan sistem adalah 93,44%. Sedangkan nilai precision yang dihasilkan adalah 77,07%. Sebaran data hasil klasifikasi dapat dilihat pada tabel 4.12

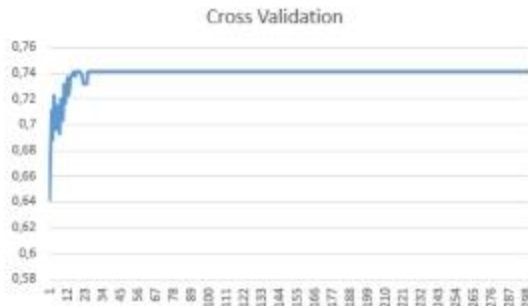
Tabel 4.12 Hasil Klasifikasi 2 Target kelas

	Cross Validation		Split Training	
	RD (sakit)	NoRD (normal)	RD (sakit)	NoRD (normal)
RD (sakit)	442	149	171	51
NoRD (normal)	8	1	12	6

Hasil klasifikasi menggunakan teknik cross validation menunjukkan bahwa dari 600 data sistem berhasil memprediksi pasien yang benar sakit sebanyak 442 orang. Sedangkan hasil klasifikasi menggunakan teknik split menunjukkan bahwa dari 240 data, sistem berhasil memprediksi pasien sakit yang benar sakit sebesar 171 orang.

c. Klasifikasi menggunakan Cross Validation dan Feature Selection

Klasifikasi dilakukan menggunakan teknik cross validation Menghasilkan akurasi sebesar 74,167% pada $k = 18$. Dengan rata-rata akurasi sebesar 74,01%. Grafik hasil klasifikasi dapat dilihat pada gambar 4.13



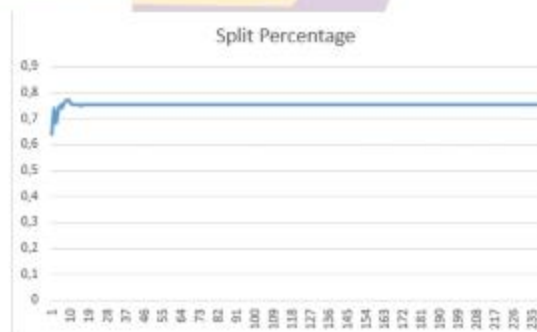
Gambar 4.13 Klasifikasi 2 kelas cross validation dan feature selection

Berdasarkan 4.13, terlihat bahwa nilai akurasi yang dihasilkan mengalami peningkatan dan stabil ketika nilai k mencapai 18. Nilai recall yang dihasilkan sistem adalah 98,24%. Sedangkan nilai precision yang dihasilkan adalah 74,80%.

Terjadi peningkatan nilai rata-rata akurasi. Rata- akurasi sebelum fitur selection adalah 73,45% sedangkan akurasi setelah fitur selection adalah 74,01%.

d. Klasifikasi menggunakan Split dan Feature Selection

Klasifikasi dilakukan menggunakan teknik cross validation Menghasilkan akurasi sebesar 77,083% pada $k = 8$. Dengan rata-rata akurasi sebesar 75,342%. Grafik hasil klasifikasi dapat dilihat pada gambar 4.14



Gambar 4.14 Klasifikasi 2 kelas split dan feature selection

Berdasarkan gambar 4.14, terlihat bahwa nilai akurasi yang dihasilkan mengalami peningkatan dan stabil ketika nilai k mencapai 8. Nilai recall yang dihasilkan sistem adalah 100%. Sedangkan nilai precision yang dihasilkan adalah 77,6%.

Terjadi peningkatan nilai akurasi tertinggi yang didapat setelah fitur selection dari akurasi awal sebesar 73,75% menjadi 77,083%. Nilai rata-rata akurasi meningkat dari 71,99% menjadi 75,342% setelah fitur selection.

Hasil sebaran data hasil klasifikasi dengan fitur selection dapat dilihat pada tabel 4.13

Tabel 4.13 Hasil klasifikasi 2 kelas setelah fitur selection

	Cross Validation		Split Training	
	RD (sakit)	NoRD (normal)	RD (sakit)	NoRD (normal)
RD (sakit)	445	150	183	55
NoRD (normal)	5	0	0	2

Keterangan :

Warna Merah = hasil prediksi

Warna Kuning = adalah label data asli

Hasil klasifikasi menggunakan teknik cross validation menunjukkan bahwa dari 600 data sistem berhasil memprediksi pasien yang benar sakit sebanyak 445 orang. Sedangkan hasil klasifikasi menggunakan teknik split menunjukkan bahwa dari 240 data, sistem berhasil memprediksi pasien sakit yang benar sakit sebesar 183 orang.

4.5. Pengujian Sistem

Pengujian sistem dilakukan menggunakan confusion matriks. Metrik yang akan diukur adalah Akurasi, Presisi, Recall. Perhitungan dilakukan menggunakan data hasil klasifikasi pada tabel 4.13. Berikut adalah proses perhitungan precision dan recall untuk kelas RD (pasien sakit).

Perhitungan klasifikasi menggunakan cross validation

$$\text{Recall} = \frac{445}{445+5} \times 100\% = 98,9 \%$$

$$\text{Precision} = \frac{448}{448+150} \times 100\% = 74,80 \%$$

Perhitungan klasifikasi menggunakan teknik split

$$\text{Recall} = \frac{183}{183+0} \times 100\% = 100 \%$$

$$\text{Precision} = \frac{183}{183+55} \times 100\% = 77,6 \%$$

Untuk perbandingan nilai precision dan recall dapat dilihat pada tabel 4.14

Tabel 4.14 Metrik precision dan recall

No	Cross Validation		Split Training Data	
	Metrik	Nilai	Metrik	Nilai
1	Accuracy	74,167%	Accuracy	77,083%
2	Precision	74,80%	Precision	77,6%
3	Recall	98,9%	Recall	100%

Untuk penjelasan masing-masing nilai metrik adalah sebagai berikut :

a. Recall

Recall disebut juga kemampuan model untuk menemukan kasus yang relevan dalam dataset. Dalam penelitian ini yang termasuk True Positive adalah data pasien sakit (RD) yang teridentifikasi sakit (RD). Sedangkan False Negative adalah data pasien normal (noRD) yang teridentifikasi normal (noRD). Nilai ini berarti adalah kemampuan model sistem untuk memprediksi data berlabel sakit (RD) ternyata teridentifikasi benar sakit (RD) dari keseluruhan data yang terprediksi sakit (RD).

Berdasarkan hasil akhir percobaan didapatkan nilai recall yaitu 98,9% untuk klasifikasi menggunakan cross validation dan 100% untuk klasifikasi menggunakan split training data. Nilai recall pada percobaan menggunakan cross validation berarti terdapat 450 data yang memiliki label asli sakit (RD) dan 445 data diantaranya diklasifikasikan oleh sistem sebagai data pasien sakit (RD). Sedangkan ketika menggunakan teknik split, terdapat 183 data yang memiliki label asli sakit (RD) dan 183 data diantaranya diklasifikasikan oleh sistem kedalam kelas sakit (RD).

Nilai recall yang semakin tinggi berarti semakin banyak data yang terprediksi benar sakit (RD) oleh sistem ternyata memiliki label asli sakit (RD). Nilai Recall yang tinggi berarti nilai False Negative kecil. Hal ini berarti bahwa tingkat kesalahan sistem dalam mendeteksi data pasien sakit semakin kecil.

b. Precision

Precision mengungkapkan proporsi data point yang menurut model relevan dengan data yang sebenarnya relevan. False positive dalam penelitian ini adalah

data yang teridentifikasi sakit (RD) oleh sistem, ternyata memiliki label asli normal (noRD). Nilai precisi menunjukkan kemampuan sistem untuk memprediksi data berlabel sakit (RD) ternyata teridentifikasi benar sakit dari keseluruhan data yang memiliki label asli sakit (RD).

Berdasarkan hasil percobaan didapatkan nilai precision yaitu 74,80% untuk klasifikasi menggunakan cross validation dan 77,6% untuk klasifikasi menggunakan teknik split. Nilai precision pada klasifikasi menggunakan cross validation memiliki arti bahwa dari 595 data yang terprediksi sakit (RD) oleh sistem, ternyata terdapat 445 data yang memiliki label asli sakit (RD). Sedangkan nilai precision pada klasifikasi menggunakan teknik split berarti bahwa dari 238 data yang diprediksi sakit oleh sistem (RD), 183 data diantaranya memiliki label asli sakit.

Nilai precision yang semakin tinggi berarti bahwa hasil yang dikeluarkan oleh sistem akan semakin tepat dengan informasi yang diminta oleh pengguna. Dalam kasus ini, informasi yang diminta adalah data pasien yang diprediksi sakit ternyata benar-benar terkena penyakit atau True Positif. Nilai True Positif yang tinggi berarti tingkat kepekaan sistem terhadap data pasien sakit semakin tinggi.

False Positif yang semakin kecil, berarti tingkat kesalahan sistem dalam mengidentifikasi pasien normal ternyata sakit semakin kecil.

4.6. Perbandingan Penelitian Lain

Berikut adalah beberapa penelitian lain yang menggunakan database Messidor sebagai sumber data penelitian:

Penelitian oleh Muhammad Usman Akram, dkk tahun 2015 berjudul “An Automated System for the Grading of Diabetic Maculopathy in Fundus Images”. Dataset yang digunakan pada penelitian ini berasal dari database Messidor. Perbedaan penelitian terletak pada jenis penyakit yang dideteksi, yaitu diabetes makulopati. Algoritma klasifikasi yang digunakan adalah SVM. Hasil yang diperoleh adalah sensitifitas sebesar 92,6%, spesifitas 97,8%, dan akurasi 97,3%.

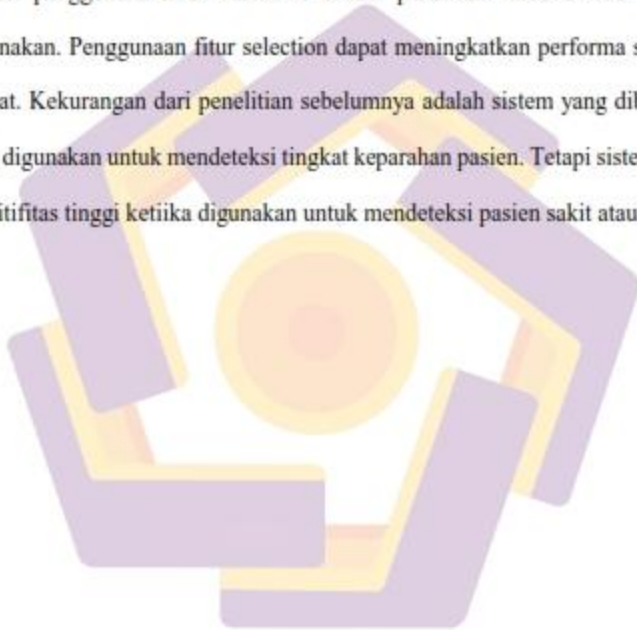
Penelitian oleh Purabi Sharma, dkk tahun 2014 berjudul “A System for Grading Diabetic Maculopathy Severity Level”. Penelitian ini menggunakan dataset dari Messidor dan DIARETDB1. Perbedaan penelitian terletak pada jenis penyakit yang dideteksi, yaitu diabetes makulopati. Deteksi dilakukan dengan menganalisa pembesaran makula pada gambar retina. Hasil yang diperoleh memiliki sensitifitas sebesar 96,2%, spesifitas sebesar 100%, dan akurasi 96,5%.

Penelitian oleh Wahyudi Setiawan dan Fitri Damayanti tahun 2016 berjudul “Klasifikasi Citra Retina Menggunakan K-Nearest Neighbor untuk Mendeteksi Makulopati Diabetik”. Dataset yang digunakan berjumlah 75 data berasal dari database Messidor. Ekstraksi ciri yang digunakan adalah Dimensional Linear Discriminant Analysis (*2DLDA*). Hasil yang didapat adalah sistem memiliki akurasi sebesar 93,33%.

Penelitian Anam Tariq, dkk tahun 2013 berjudul “Automated Detection and Grading of Diabetic Maculopathy in Digital Retinal Images. Dataset yang digunakan berasal dari database Messidor dan STARE dengan total berjumlah 1281 data. Metode preprocessing yang digunakan adalah *Gaussian Model Filter*. Hasil

yang diperoleh adalah akurasi sebesar 97,83% untuk deteksi pembengkakan makula.

Berdasarkan beberapa penelitian tersebut dapat disimpulkan bahwa data dalam dabase Messidor dapat digunakan untuk beberapa kasus selain deteksi Retinopati Diabetes. Adapun hasil penelitian yang membedakan dari penelitian lain adalah penggunaan fitur selection dalam pemilihan metrik fitur citra yang digunakan. Penggunaan fitur selection dapat meningkatkan performa sistem yang dibuat. Kekurangan dari penelitian sebelumnya adalah sistem yang dibuat kurang baik digunakan untuk mendeteksi tingkat keparahan pasien. Tetapi sistem memiliki sensitifitas tinggi ketika digunakan untuk mendeteksi pasien sakit atau tidak.



BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan penelitian yang telah dilakukan, maka terdapat beberapa kesimpulan yang dapat diambil, yaitu

1. Percobaan pertama dilakukan klasifikasi sebanyak 2 kali. Data terklasifikasi ke dalam 4 kelas yaitu, NoRD, Mild, Moderate, dan Severe. Klasifikasi dilakukan dengan tujuan mendeteksi tingkat keparahan pasien, yaitu normal, ringan, sedang, dan parah. Skenario pertama klasifikasi data asli menggunakan cross validation menghasilkan akurasi sebesar 29,67% presisi sebesar 29,601%, dan recall 29,67%. Skenario kedua klasifikasi data asli menggunakan split menghasilkan akurasi 31,25%, presisi sebesar 31,59%, dan recall 31,25%.
2. Percobaan kedua dilakukan klasifikasi sebanyak 2 kali menggunakan dataset hasil fitur selection. Skenario pertama klasifikasi dataset fitur selection menggunakan cross validation menghasilkan akurasi 30,33%, presisi 30,159%, dan recall 30,33%. Skenario kedua klasifikasi dataset fitur selection menggunakan split menghasilkan akurasi 32,083%, presisi 32,56%, dan recall 32,08%.
3. Percobaan ketiga dilakukan dengan menguji sistem untuk klasifikasi dengan target kelas yang lebih sedikit (2 kelas). Data ciri orde dua dikelompokkan kembali menjadi 2 kelas yaitu NoRD (tidak sakit) dan RD (sakit). Kelas RD terdiri dari Mild, Moderate, dan Severe. Tujuan

dilakukan klasifikasi adalah untuk mendeteksi data pasien sakit atau tidak. Untuk nilai akurasi menggunakan teknik cross validation menghasilkan akurasi sebesar 73,833%, nilai precision 74,78% dan nilai recall 98,2%. Sedangkan untuk nilai akurasi yang didapat menggunakan teknik split adalah 73,75%, nilai precision sebesar 93,44% dan nilai recall sebesar 77,07%.

4. Percobaan keempat dilakukan klasifikasi dataset hasil fitur selection menggunakan cross validation menghasilkan akurasi sebesar 74,167%, presisi sebesar 74,80% dan recall sebesar 98,9%. Klasifikasi dataset fitur selection menggunakan split menghasilkan akurasi sebesar 77,083%, presisi sebesar 77,6% dan recall sebesar 100%.
5. Terjadi peningkatan hasil akurasi yang didapatkan pada klasifikasi sebelum dan sesudah penggunaan fitur selection. Berdasarkan 4 percobaan yang dilakukan maka dapat dinyatakan bahwa penggunaan fitur selection dalam pemilihan atribut fitur dapat meningkatkan performa sistem.
6. Berdasarkan hasil percobaan, maka kemampuan sistem untuk memprediksi pasien sakit dari seluruh data pasien benar sakit (Recall) adalah 98,9% untuk Cross Validation dan 100% untuk Split Percentage. Kemampuan sistem untuk klasifikasi data pasien benar sakit dari keseluruhan data yang terprediksi sakit (Precision) adalah 74,80% untuk Cross Validation dan 77,6% untuk Split Percentage.
7. Berdasarkan beberapa skenario percobaan yang telah dilakukan, maka dapat dinyatakan bahwa sistem yang dibangun menghasilkan performa

yang cukup baik ketika mendeteksi pasien sakit atau tidak (klasifikasi menggunakan 2 kelas target noRD dan RD) berdasarkan gambar retina. Akan tetapi sistem menghasilkan performa yang kurang baik ketika digunakan untuk mendeteksi tingkat keparahan pasien (klasifikasi menggunakan multi-class target noRD, mild, moderate, severe).

5.2. Saran

Berdasarkan analisis yang dilakukan, terdapat beberapa faktor yang dapat menyebabkan nilai akurasi pada klasifikasi 4 kelas rendah, antara lain:

1. Jumlah data: jumlah data berpengaruh pada tingkat performansi sistem.
2. Perbedaan tekstur gambar pada tiap kelas dalam dataset tidak memiliki perbedaan yang jelas. Sehingga sistem sulit membedakan gambar antara kelas satu dan lainnya.
3. Penggunaan metode GLCM kurang cocok digunakan untuk gambar yang tidak memiliki tekstur.

Adapun saran untuk penelitian selanjutnya adalah:

1. Perlu dilakukan penyortiran data gambar. Jika memungkinkan, gunakan gambar dengan kualitas terbaik. Terbaik dalam artian, terlihat jelas perbedaan yang dimiliki gambar pada tiap-tiap kelas atau kelompoknya.
2. Melakukan penambahan metode preprocessing lain pada gambar yang digunakan, seperti segmentasi ataupun augmentasi pada gambar, sehingga dapat menampilkan tekstur gambar dengan lebih jelas.

DAFTAR PUSTAKA

PUSTAKA BUKU

- Alpaydin, E. (2010). *Introduction to Machine Learning*. MIT press.
- Anonim. (2014). *Waspada Diabetes Eat Well Live Well Situasi Dan Analisis Diabetes*. Jakarta Selatan: Pusat Data Dan Informasi Kementerian Kesehatan RI.
- Basuki, A. (2005). *Metode Numerik dan Algoritma Komputasi*. Yogyakarta: ANDI.
- Gonzalez, R.C. & Woods, R.E. (2010). *Digital Image Processing, Third edition*. New Jersey: Prentice Hall.
- Kingham, K. (2009). *Makan Oke Hidup Oke dengan Diabetes*. Jakarta: Penerbit Erlangga.
- Lukmanto, D. H. (1995). *Diagnosis Fisik Edisi 17*. Jakarta: Penerbit Buku Kedokteran.
- Prasetyo, E. (2011). *Pengolahan Citra Digital dan Aplikasinya menggunakan Matlab*. Yogyakarta : Andi.
- Purnomo, M.H. dan Muntasa, A. (2010). *Konsep Pengolahan Citra Digital dan Ekstraksi Fitur*. Surabaya: GRAHA ILMU.
- Sutoyo, T. (2009). *Teori Pengolahan Citra Digital*. Yogyakarta: Penerbit Andi.

PUSTAKA MAJALAH, JURNAL ILMIAH ATAU PROSIDING

- Coltelli, Primo and Gualtieri, Paulo. (1990). A procedure for the extraction of object features in microscope images. *International Journal of Bio-Medical Computing*, Volume 25, Issues 2-3, Pages 169-176.
- Dillak, R.Y., Harjoko, A. (2011). Klasifikasi Fase Retinopati Diabetes Menggunakan Backpropagation Neural Network. *IJEIS*, 89-100.
- Eleyan, A. and Demirel, H. (2011). Co-Occurrence Matrix and Its Statistical Features as a New Approach for Face Recognition. *Turkish Journal of Electrical Engineering and Computer Science*.
- Fadzil, A. I. (2011). Analysis of Retinal Fundus Images for Grading of Diabetic Retinopathy Severity. *Med. Biol. Eng. Comput*.
- Gitasari, R. A. (2015). **KLASIFIKASI PENYAKIT DIABETES RETINOPATI BERDASARKAN CITRA DIGITAL DENGAN MENGGUNAKAN**

METODE WAVELET DAN SUPPORT VECTOR MACHINE. *e- Proceeding of Engineering Vol 2 No 1 Telkom University*. 2015.

- Mohanaiah., P. S. (2013). Image Texture Feature Extraction Using GLCM Approach. *Journal of Scientific and Research Publications*.
- Namrata and Aurora, S. (2015). Exudates Detection from Digital Fundus Image of Diabetic Retinopathy. *International Journal of Advances in Biology*.
- Namrata dan Arora, S. (2015). EXUDATES DETECTION FROM DIGITAL FUNDUS IMAGE OF DIABETIC RETINOPATHY. *International Journal of Advances in Biology*.
- Neneng, A. K. (2016). Support Vector Machine Untuk Klasifikasi Citra Jenis Daging Berdasarkan Tekstur Menggunakan Ekstraksi Ciri Gray Level Co-Occurrence Matrices(GLCM). *Jurnal Sistem Informasi Bisnis*.
- P. P. Pullaperuma, A. T. Dharmaratne. (2013). Taxonomy of File Fragments Using Gray-Level Co-Occurrence Matrices. *International Conference on Digital Image Computing: Techniques and Applications (DICTA)*.
- R.M. Haralick, K. Shanmugam, dan I. Dinstein. (1973). Textural Features For Image Classification. *IEEE Transaction On System Man and Cybernatics*, Vol 3, No. 6 Hal 25-31.
- Setiawan, W. D. (2016). KLASIFIKASI CITRA RETINA MENGGUNAKAN K-NEARESTNEIGHBOR UNTUK MENDETEKSI MAKULOPATI DIABETIK. *Seminar Nasional Sains dan Teknologi Fakultas Teknik Universitas Muhammadiyah Jakarta*. Jakarta.
- Stehling, Renato O., Nascimento, Mario A., Falcão, Alexandre X. (2002). A compact and efficient image retrieval approach based on border/interior pixel classification. *Proceedings of the eleventh international conference on Information and knowledge management*, 102-109.
- Surya, R. A., Fadhil, A., dan Yudhana, A. (2017). Ekstraksi Ciri Metode Gray Level Co-Occurrence Matrix (GLCM) dan Filter Gabor untuk Klasifikasi Citra Batik Pekalongan. *Jurnal Pengembangan IT*.
- Usman, K. (2008). Perhitungan Sel Darah Merah Bertumpuk Berbasis Pengolahan Citra Digital dengan Operasi Morfologi. *paper presented to the Seminar Nasional Informatika*. Yogyakarta.
- Widodo, R. W. (2018). Pemanfaatan Ciri Gray Level Co-Occurrence Matrix (GLCM) Citra Buah Jeruk Keprok (*Citrus reticulata* Blanco) untuk Klasifikasi Mutu. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*.

Wu, Xindong., Kumar, Vipin., et all. (2006). Top 10 Algorithms in Data Mining. *International Conference on Data Mining (ICDM)*.

PUSTAKA LAPORAN PENELITIAN

Gorasiya, D. B. (2019). Early Detection of Curable Diabetic Retinopathy from Retinal Images.

Guojun Lu, Atul Sajjanhar. (1999). Region-based shape representation and similarity measure suitable for content-based image retrieval. *Multimedia Systems*, 165-174.

Nalband, Saif., Prince, Amalin., Agrawal, Anita. (2017). An entropy based feature extraction and classification of vibroarthographic signal using improved complete ensemble empirical mode decomposition with adaptive noise. *IET Science Measurement*.

Rahman, K., Kartasmita, A.S., Heksan. (2013). Pedoman Penanganan Retinopati Diabetika. PERDAMI Seminat Vitreoretina.

Riliandi, G. P. (2013). Deteksi Diabetes Melalui Citra Retina Menggunakan Transformasi Wavelet dan Learning Vector Quantization.

Setiawan, W. D. (2016). Klasifikasi Citra Retina Menggunakan K-Nearest Neighbor untuk Mendeteksi Makuslopati Diabetik.

Siquera, F. R. (2013). Multi-scale Gray Level Co-Occurence Matrices for Texture Description. *Neurocomputing*, 336-345.

LAMPIRAN



Lampiran 1 Ekstraksi Ciri dan Preprocessing Data

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Aug 19 17:09:01 2020
4
5 @author: FADIL
6 """
7 import pandas as pd
8 import cv2
9 import glcm_fitur
10 import time
11
12 start_time = time.time()
13
14 data = pd.read_csv("../S2/tesis dataset/image_dataset.csv", sep=",", encoding='latin-1')
15
16 nama_gambar = list(data["nama_gambar"])
17 RG = list(data["rd_level"])
18
19 print(nama_gambar)
20 fitur = []
21 for x in range(len(nama_gambar)):
22
23
24     #print("../S2/tesis dataset/dataset_resize/resized_image_"+nama_gambar[x]+".")
25
26     image = cv2.imread("../S2/tesis dataset/dataset_resize/resized-image_"+nama_gambar[x]+".")
27
28     #img = cv2.resize(image, (650, 550))
29
30     #_, binary = cv2.threshold(gray, 225, 255, cv2.THRESH_BINARY_INV)
31
32     rgb = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
33
34     gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
35
36     _, binary = cv2.threshold(gray, 225, 255, cv2.THRESH_BINARY_INV)
37
38     contours, hierarchy = cv2.findContours(binary, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)
39
40
41     image_contour = cv2.drawContours(gray, contours, -1, (0, 255, 0), 25)
42
43     glcm = glcm_fitur.getGLCM(image_contour)
44
45     entropi = str(round(glcm_fitur.getEntropy(glcm),4))
46     energi = str(round(glcm_fitur.getASM(glcm),4))
47     homogenitas = str(round(glcm_fitur.getIdM(glcm),4))
48     kontras = str(round(glcm_fitur.getContrast(glcm),4))
49     korelasi = str(round(glcm_fitur.getCorrelation(glcm),4))
50
51     tbl = nama_gambar[x], entropi, energi, homogenitas, kontras, korelasi, RG[x]
52
53     fitur.append(tbl)
54
55
56     print("gambar",nama_gambar[x],"=", entropi, energi, homogenitas, kontras, korelasi, RG[x])
57
58     #print("../data retina/new folder/Basis1/"+nama_gambar[x]+".")
59
60 df = pd.DataFrame(fitur)
61 csv = df.to_csv("fitur_messidor_ordo2.csv", index= False)
62
63 print(time.time() - start_time)

```

Lampran 3 Source Code GLCM

```

7
8 import numpy as np
9
10 L = 256
11
12 def getAngledGLCM(img, alpha):
13     row, col = img.shape
14     row, col, layer = img.shape
15     glcm = np.zeros((L,L), np.uint8)
16     xy = [0,0]
17     rowStart = 0
18     colStart = 0
19     rowEnd = row
20     colEnd = col - 1
21
22     if (alpha == 0):
23         xy = [0,1]
24     elif (alpha == 45):
25         xy = [1,1]
26         rowStart = 1
27         colStart = 0
28     elif (alpha == 90):
29         xy = [1,0]
30         rowStart = 1
31         colEnd = col
32     else:
33         xy = [1,-1]
34         rowStart = 1
35         colStart = 1
36         colEnd = col
37
38     for x in range(rowStart,rowEnd):
39         for y in range(colStart,colEnd):
40             pixVal = img[x,y]
41             pixValNeighbor = img[x + xy[0],y + xy[1]]
42             glcm[pixVal,pixValNeighbor] += 1
43
44     return glcm / glcm.sum()
45
46 def getGLCM(img):
47     glcm = np.zeros((L,L), np.float64)
48     alpha = 0
49
50     for i in range(4):
51         glcm += getAngledGLCM(img, alpha)
52         alpha += 45
53
54     return glcm / 4
55
56 def getSumGLCM(glcm):
57     sumX = []
58     sumY = []
59
60     for i in range(L):
61         sumY.append(glcm[:,i].sum())
62         sumX.append(glcm[i,:].sum())
63
64     return [sumX, sumY]
65
66 def getMean(glcm):
67     sumX, sumY = getSumGLCM(glcm)
68     meanX = 0.0
69     meanY = 0.0
70
71     for i in range(L):
72         meanX += i * sumX[i]
73         meanY += i * sumY[i]
74
75     return [meanX, meanY]

```

```

77 def getVarianceXY(glcM):
78     meanX, meanY = getMean(glcM)
79     sumX, sumY = getSumGLCM(glcM)
80     varX = 0.0
81     varY = 0.0
82
83     for i in range(L):
84         varX += ((i - meanX) ** 2) * sumX[i]
85         varY += ((i - meanY) ** 2) * sumY[i]
86
87     return [varX, varY]
88
89 def getStandardDeviation(glcM):
90     varX, varY = getVarianceXY(glcM)
91
92     return [np.sqrt(varX), np.sqrt(varY)]
93
94
95 def getASM(glcM):
96     return np.power(glcM.flatten(), 2).sum()
97
98
99 def getContrast(glcM):
100     con = 0.0
101
102     for x in range(L):
103         for y in range(L):
104             con += ((x - y) ** 2) * glcM[x,y]
105
106     return con
107
108
109 def getCorrelation(glcM):
110     cor = 0.0
111     meanX, meanY = getMean(glcM)
112     sdX, sdY = getStandardDeviation(glcM)
113
114     for x in range(L):
115         for y in range(L):
116             cor += (x * y) * glcM[x,y]
117
118     return (cor - (meanX * meanY)) / (sdX * sdY)
119
120
121 def getVariance(glcM, meanX, meanY):
122     meanX, meanY = getMean(glcM)
123     var = 0.0
124
125     for x in range(L):
126         for y in range(L):
127             var += (x - meanX) * (y - meanY) * glcM[x,y]
128
129     return var
130
131
132 def getIDM(glcM):
133     idm = 0.0
134
135     for x in range(L):
136         for y in range(L):
137             idm += glcM[x,y] / (1 + ((x - y) ** 2))
138
139     return idm
140

```

```

142 def getEntropy(glcm):
143     E = 0.0
144
145     for x in range(L):
146         for y in range(L):
147             if (glcm[x,y] <= 0.0):
148                 E += glcm[x,y] * np.log2(np.e)
149             else:
150                 E += glcm[x,y] * np.log2(glcm[x,y])
151
152     return -E

```

Lampiran 4 Weka Fitur Selection

```

Search Method:
  Best first.
  Start set: no attributes
  Search direction: forward
  Stale search after 5 node expansions
  Total number of subsets evaluated: 186
  Merit of best subset found: 0.328

Attribute Subset Evaluator (supervised, Class (nominal): 25 kelas):
  Wrapper Subset Evaluator
  Learning scheme: weka.classifiers.lazy.IBk
  Scheme options: -K 70 -W 0 -A weka.core.neighboursearch.LinearIBSearch -A "weka.c
  Subset evaluation: classification accuracy
  Number of folds for accuracy estimation: 5

Selected attributes: 2,3,8,24 : 4
  kontras_45
  kontras_90
  korelasi_135
  disimilarity_135

```

Lamplan 5 Source Code Algoritma KNN

```

11 import sklearn
12 from sklearn.utils import shuffle
13 from sklearn.neighbors import KNeighborsClassifier
14 import pandas as pd
15 import numpy as np
16 from sklearn import linear_model, preprocessing
17
18 data = pd.read_csv("F:/52/tesis_python/tesis/ordo_22.csv", sep=",")
19
20
21 #nama_gambar = list(data["nama_gambar"])
22 entropi = list(data["entropi"])
23 energi = list(data["energi"])
24 homogenitas = list(data["homogenitas"])
25 kontras = list(data["kontras"])
26 korelasi = list(data["korelasi"])
27 rd_level = list(data["rd_level"])
28
29 #print(entropi)
30
31 predict = "rd_level"
32 tbl = []
33 #p = entropi, energi, homogenitas, kontras, korelasi
34 #print(p)
35 X = list(zip(entropi, energi, homogenitas, kontras, korelasi))
36 #t = tbl.append(p)
37
38 y = list(rd_level)
39
40 x_train, x_test, y_train, y_test = sklearn.model_selection.train_test_split(X, y, test_size = 0.1)
41
42 #print(x_train)
43
44 model = KNeighborsClassifier(n_neighbors=5)
45
46 model.fit(x_train, y_train)
47 acc = model.score(x_test, y_test)
48
49
50
51 predicted = model.predict(x_test)
52
53 names = ["0", "1"]
54
55 #print(len(predicted))#
56
57 for x in range(len(predicted)):
58     print("predicted =", names[predicted[x]], "Actual =", names[y_test[x]])
59
60
61 print(acc)
62
63 test = model.predict([[1.244,0.30703,0.50456,1211.599,0.3956]])
64 print(rd_level[test[0]])
65

```


Lampiran 6 Hasil Klasifikasi KNN (Weka)

```

=== SUMMARY OF THIS MODEL ===

Time taken to test model on test split: 1.05 seconds

=== Summary ===

Correctly Classified Instances      155          75.3604 %
Incorrectly Classified Instances    49           24.6396 %
Kappa statistic                    0.0407
Mean absolute error                0.3359
Root mean squared error            0.4301
Relative absolute error            34.3376 %
Root relative squared error        100.0094 %
Total Number of Instances          204

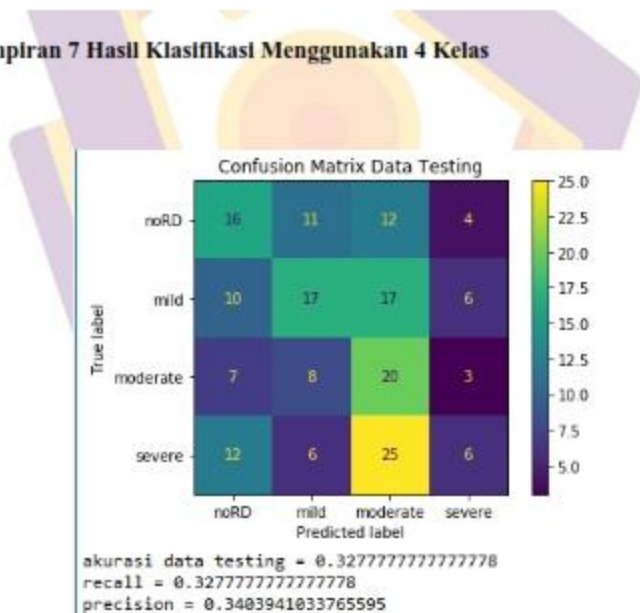
=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
-----
0.937  0.359  0.765  0.327  0.462  0.086  0.504  0.311  rd
0.041  0.313  0.800  0.041  0.075  0.086  0.504  0.311  normal
Weighted Avg.  0.740  0.732  0.701  0.700  0.473  0.086  0.504  0.483

=== Confusion Matrix ===

  a  b  <== classified as
153  2  |  a = rd
 47  2  |  b = normal
  
```

Lampiran 7 Hasil Klasifikasi Menggunakan 4 Kelas



Lampiran 8 Hasil Klasifikasi Menggunakan 2 Kelas