



DETEKSI TUBERCULOSIS PARU-PARU PADA CITRA CHEST X-RAY MENGGUNAKAN FITUR GRAY LEVEL CO-OCCURRENCE MATRIX (GLCM) YANG TEROPTIMASI DENGAN METODE PRINCIPAL COMPONENT ANALYSIS

TESIS

PROGRAM MAGISTER TEKNIK ELEKTRO KEMINATAN SISTEM KOMUNIKASI DAN INFORMATIKA

**Ditujukan untuk memenuhi persyaratan
memperoleh gelar Magister Teknik**



IMAM JUNAEDI

NIM. 176060300111006

UNIVERSITAS BRAWIJAYA

FAKULTAS TEKNIK

MALANG

2019

DAFTAR ISI

	Halaman
KATA PENGANTAR	xix
DAFTAR ISI	xxi
DAFTAR TABEL	xxiii
DAFTAR GAMBAR	xxiv
DAFTAR LAMPIRAN	xxvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	4
BAB II TINJAUAN PUSTAKA	5
2.1 Penelitian Terdahulu	5
2.2 Paru-paru	7
2.3 Radiografi	8
2.4 Tuberculosis (TB)	12
2.5 GLCM	15
2.6 <i>Principal Component Analysis</i> (PCA)	17
2.7 SVM	21
2.8 <i>Confussion Matrix</i>	27
2.9 <i>Receiver Operating Characteristic</i> (ROC)	28
2.10 Matlab 2018a	30
2.11 Weka 3.8	31
BAB III KERANGKA KONSEP PENELITIAN	33
3.1 Kerangka Konsep Penelitian	33
3.2 Variabel Penelitian	34
3.3 Alat Penelitian	35
3.4 Analisis Masalah	35
3.5 Konsep Solusi	35
3.5.1 Pengumpulan Data	35
3.4.2 Perancangan Sistem	36
3.6 Hipotesis	36
BAB IV METODE PENELITIAN	37
4.1 Perancangan Sistem	37
4.1.1 <i>Preprocessing</i>	38
4.1.2 Segmentasi	39
4.1.3 Ekstraksi Fitur GLCM	39
4.1.4 Optimasi Fitur GLCM	44
4.1.5 Klasifikasi	45
4.2 Uji Validasi Sistem	46
4.3 Analisis Performansi Sistem	47
4.3.1 <i>Confusion Matrix</i>	47
4.3.1.1 Perhitungan Akurasi	47



4.3.1.2 Perhitungan Sensitivitas	48
4.3.1.3 Perhitungan Spesifisitas.....	48
4.3.1.4 Pembentukan Grafik ROC.....	48
BAB V HASIL DAN PEMBAHASAN	51
5.1 Perancangan Sistem	51
5.1.1 <i>Preprocessing</i>	51
5.1.2 Segmentasi.....	52
5.1.3 Ekstraksi fitur GLCM.....	54
5.1.4 Optimasi fitur GLCM.....	57
5.1.5 Klasifikasi	57
5.2 Hasil Penelitian dan Pembahasan	57
5.2.1 Analisis Akurasi, Sensitivitas, dan Spesifisitas	57
5.2.2 Analisis Grafik ROC.....	63
BAB VI KESIMPULAN DAN SARAN.....	69
6.1 Kesimpulan	69
6.2 Saran	70
DAFTAR PUSTAKA	71
LAMPIRAN.....	75



DAFTAR TABEL

No.	Judul	Halaman
Tabel 2.1	Penelitian terdahulu	6
Tabel 2.2	Tahapan perjalanan alamiah penyakit TB.....	13
Tabel 2.3	<i>Confusion Matrix</i>	28
Tabel 2.4	Klasifikasi nilai <i>area under ROC</i>	30
Tabel 5.1	<i>Confusion matrix</i> sistem klasifikasi SVM pada uji validasi <i>2-fold</i> hingga <i>10-fold cross-validation</i>	60
Tabel 5.2	Hasil perhitungan akurasi, sensitivitas dan spesifisitas dari uji <i>cross-validation</i> dengan masukan GLCM tanpa optimasi maupun GLCM yang telah dioptimasi.....	63
Tabel 5.3	Hasil perhitungan TPR dan FPR dari uji <i>cross-validation</i> dengan masukan GLCM tanpa optimasi maupun GLCM yang telah dioptimasi.....	64



DAFTAR GAMBAR

No.	Judul	Halaman
Gambar 2.1	Anatomi paru-paru manusia.....	7
Gambar 2.2	Pleura.....	8
Gambar 2.3	Spektrum gelombang elektromagnetik.....	8
Gambar 2.4	Pengambilan citra CXR.....	9
Gambar 2.5	Variasi kontras pada citra CXR.....	10
Gambar 2.6	Citra CXR normal.....	10
Gambar 2.7	Pola abnormalitas paru pada citra CXR.....	12
Gambar 2.8	TB Primer.....	14
Gambar 2.9	TB Sekunder.....	14
Gambar 2.10	Pembentukan GLCM.....	15
Gambar 2.11	Orientasi spasial pembentuk GLCM.....	15
Gambar 2.12	Perhitungan matriks GLCM.....	16
Gambar 2.13	Plot dari 50 observasi pada dua variabel x_1, x_2	18
Gambar 2.14	Plot dari 50 observasi dalam bentuk PC z_1, z_2	18
Gambar 2.15	Pengukuran anatomi siswa: plot 28 siswa terhadap dua PC pertama.....	19
Gambar 2.16	<i>Optimal hyperplane</i>	22
Gambar 2.17	Margin <i>hyperplane</i> lunak.....	26
Gambar 2.18	Wilayah dalam kurva ROC.....	29
Gambar 2.19	Contoh performansi sistem klasifikasi dalam ROC.....	30
Gambar 2.20	Tampilan aplikasi Matlab 2018a.....	31
Gambar 2.21	Tampilan aplikasi Weka 3.8.....	32
Gambar 3.1	Kerangka konsep penelitian.....	34
Gambar 4.1	Metodologi penelitian.....	37
Gambar 4.2	Blok diagram sistem.....	38
Gambar 4.3	Diagram alir <i>preprocessing</i>	38
Gambar 4.4	Diagram alir segmentasi.....	39
Gambar 4.5	Diagram alir ekstraksi fitur.....	39
Gambar 4.6	Ekstraksi kontras.....	40
Gambar 4.7	Ekstraksi korelasi.....	40
Gambar 4.8	Ekstraksi energi.....	41
Gambar 4.9	Ekstraksi homogenitas.....	41



Gambar 4.10	Ekstraksi <i>mean</i>	41
Gambar 4.11	Ekstraksi deviasi standar.....	42
Gambar 4.12	Ekstraksi entropi.....	42
Gambar 4.13	Ekstraksi RMS.....	43
Gambar 4.14	Ekstraksi varian.....	43
Gambar 4.15	Ekstraksi <i>smoothness</i>	43
Gambar 4.16	Ekstraksi <i>kurtosis</i>	44
Gambar 4.17	Ekstraksi <i>skewness</i>	44
Gambar 4.18	Optimasi fitur GLCM dengan metode PCA.....	45
Gambar 4.19	Diagram alir proses klasifikasi.....	46
Gambar 4.20	Diagram alir analisis performansi sistem.....	47
Gambar 4.21	Diagram alir perhitungan akurasi.....	47
Gambar 4.22	Diagram alir perhitungan sensitivitas.....	48
Gambar 4.23	Diagram alir perhitungan spesifisitas.....	48
Gambar 4.24	Diagram alir pembentukan grafik ROC.....	49
Gambar 5.1	<i>Pseudo code preprocessing</i>	52
Gambar 5.2	Hasil dari tahapan <i>preprocessing</i> citra CXR.....	52
Gambar 5.3	<i>Pseudo code</i> segmentasi.....	53
Gambar 5.4	Hasil dari tahapan segmentasi citra CXR.....	54
Gambar 5.5	Hasil dari tahapan segmentasi citra CXR yang tidak sempurna.....	54
Gambar 5.6	<i>Pseudo code</i> ekstraksi fitur GLCM bagian 1 dari 2.....	55
Gambar 5.7	<i>Pseudo code</i> ekstraksi fitur GLCM bagian 2 dari 2.....	56
Gambar 5.8	Hasil ekstraksi fitur GLCM.....	56
Gambar 5.9	Hasil uji validasi <i>2-fold cross-validation</i> pada SVM (GLCM tanpa optimasi).....	58
Gambar 5.10	Hasil uji validasi <i>2-fold cross-validation</i> pada SVM (GLCM yang telah dioptimasi).....	59
Gambar 5.11	Grafik hasil perhitungan akurasi, sensitivitas dan spesifisitas dari uji <i>cross-validation</i> sistem dengan masukan GLCM tanpa optimasi.....	62
Gambar 5.12	Grafik hasil perhitungan akurasi, sensitivitas dan spesifisitas dari uji <i>cross-validation</i> sistem dengan masukan GLCM yang telah dioptimasi.....	62
Gambar 5.13	Grafik ROC dari perhitungan manual.....	65
Gambar 5.14	Grafik ROC Weka 3.8 pada kelas normal.....	66
Gambar 5.15	Grafik ROC Weka 3.8 pada kelas PTB.....	66



DAFTAR LAMPIRAN

No.	Judul	Halaman
Lampiran 1.	<i>Source code preprocessing</i>	75
Lampiran 2.	<i>Source code segmentasi</i>	75
Lampiran 3.	<i>Source code ekstraksi fitur</i>	77
Lampiran 4.	Hasil segmentasi citra CXR	77
Lampiran 5.	Hasil ekstraksi fitur	91
Lampiran 6.	Fitur hasil optimasi	94
Lampiran 7.	Hasil klasifikasi GLCM tanpa optimasi	95
Lampiran 8.	Hasil klasifikasi GLCM yang teroptimasi	99



BAB I PENDAHULUAN

1.1 Latar Belakang

Tuberculosis (TB) adalah penyakit menular mematikan dengan *Mycobacterium Tuberculosis* (MTB) sebagai penyebabnya (Varaine & Rich, 2014). Laporan World Health Organization (WHO) pada tahun 2017 menyebutkan bahwa Indonesia masuk kedalam daftar *20 high TB burden countries* dengan jumlah kasus 446.732. WHO memperkirakan terdapat 107.000 kematian penderita TB di Indonesia pada 2017 (WHO, 2018). Faktor risiko yang menyebabkan penderita TB meninggal dunia beberapa diantaranya adalah akibat diagnosis yang terlambat, tidak adekuatnya proses pengobatan, dan buruknya kondisi kesehatan awal atau adanya penyakit lain (Kementerian Kesehatan Republik Indonesia, 2014).

Citra CXR secara historis merupakan alat utama untuk mendeteksi tuberculosis (TB) (WHO, 2016). CXR memiliki sensitifitas yang cukup tinggi terhadap pendeteksian TB, khususnya saat digunakan untuk mendeteksi abnormalitas paru terkait TB pada kasus pasien yang tidak memiliki gejala TB yang nampak secara kasat mata. Citra CXR dianalisis oleh ahli radiologi untuk mengetahui ada atau tidaknya abnormalitas pada paru-paru. Hasil analisis dari ahli radiologi dalam menganalisis citra CXR dipengaruhi oleh faktor subyektifitas dari ahli radiologi, seperti pengalaman dari ahli radiologi, kondisi pengamatan, kelelahan, dan lain-lain (Tingberg, 2000). Faktor subyektifitas dari ahli radiologi dapat diatasi dengan sistem deteksi TB pada citra CXR yang terkomputerisasi. Sistem deteksi TB pada citra CXR yang terkomputerisasi juga dapat digunakan sebagai opini kedua dalam pemberian diagnosis.

Deteksi abnormalitas paru-paru pada citra CXR telah banyak diteliti pada beberapa tahun terakhir ini. Beberapa diantaranya adalah deteksi TB pada paru-paru (Mahajan & Ganorkar, 2017), deteksi *cardiomegaly* (Mishra dkk, 2016), deteksi pneumonia (Al-Hadidi dkk, 2016), deteksi kanker paru-paru (Singh & Asuntha, 2016) dan lain-lain. Banyak metode yang digunakan untuk proses ekstraksi fitur dan klasifikasi dalam pendeteksian abnormalitas paru-paru. Metode ekstraksi fitur yang digunakan adalah *Histogram of Gradient* (HOG) (Mahajan & Ganorkar, 2017), *Gray Level Co-occurrence Matrix* (GLCM) (Pawar & Ganorkar, 2016), *Haar Wavelet Transform* (HWT) (Krithika & Alice, 2017), dan *Local Binary Pattern* (LBP) (Chand & Raj, 2016).

GLCM dikenal sebagai metode matematis untuk mendeteksi abnormalitas paru-paru dan memberikan kesempatan bagi dokter untuk melokalisasi tipe-tipe jaringan abnormal, baik tumor maupun *pulmonary edema* (penumpukan cairan pada kantong udara paru-paru) (Zayed & Elnemr, 2015). Sepuluh fitur GLCM mampu mendeteksi TB dengan akurasi sebesar 75%, spesifisitas sebesar 75% dan sensitivitas 75% (Pawar & Ganorkar, 2016).

Sistem deteksi TB Pawar dan Ganorkar dapat dikembangkan dengan mengaplikasikan optimasi fitur pada GLCM untuk meningkatkan performansi sistem.

Metode optimasi yang umumnya digunakan untuk optimasi fitur GLCM adalah *Principal Component Analysis* (PCA). PCA mampu mengoptimasi fitur GLCM dengan cara mereduksi fitur tekstur GLCM pada sistem klasifikasi komunitas vegetasi *sub-Antartic* (Murray dkk, 2010). Pada penelitian Murray dkk, tiga komponen dari hasil optimasi GLCM dengan menggunakan PCA mampu memberikan akurasi klasifikasi komunitas vegetasi *sub-Antartic* yang lebih baik dibandingkan dengan delapan komponen GLCM. PCA juga mampu mereduksi fitur GLCM pada analisis EEG spektrogram (Mustafa dkk, 2010). Dari delapan puluh komponen GLCM EEG spektrogram yang menjadi masukan PCA, hanya lima komponen dasar yang memiliki persentase *eigenvalue of covariance* tertinggi.

Sedangkan untuk metode klasifikasi yang dapat digunakan diantaranya adalah *Euclidian Distance* (Pawar & Ganorkar, 2016), *Support Vector Machine* (SVM) (Mahajan & Ganorkar, 2017), *K-nearest Neighbour* (KNN) (Antony & Banu, 2017), dan *Convolutional Neural Network* (CNN) (Liu dkk, 2017). SVM memiliki nilai akurasi yang tinggi untuk pendeteksian penyakit tuberculosis pada citra CXR (Mahajan & Ganorkar, 2017) dan untuk proses klasifikasi citra CXR untuk membedakan *frontal view* dan *lateral view* (Xue dkk, 2015).

SVM merupakan *supervised machine learning* yang dapat digunakan untuk klasifikasi. SVM memiliki tingkat akurasi yang cukup tinggi. SVM banyak diaplikasikan pada bidang bioinformatika, seperti untuk deteksi tuberculosis (Mahajan & Ganorkar, 2017), deteksi tumor otak (Kadam & Dhole, 2017), dan klasifikasi citra medis (Camlica dkk, 2015). Pada sistem deteksi tuberculosis pada CXR, SVM mampu melakukan klasifikasi dengan tingkat akurasi 88%, sensitivitas 88% dan spesifisitas 88% (Mahajan & Ganorkar, 2017). Keluaran klasifikasi dari sistem yang didesain Gayatri dapat ditingkatkan performansinya dengan menggunakan fitur GLCM yang telah dioptimasi sebagai masukan SVM.





Dalam penelitian ini didesain suatu sistem diagnosis terkomputerisasi untuk mendeteksi TB paru-paru pada citra CXR dengan bantuan software Matlab 2018a dan Weka 3.8. Citra CXR diproses melalui beberapa tahapan, yaitu *preprocessing*, segmentasi, ekstraksi fitur, optimasi fitur dan klasifikasi. Fitur GLCM yang diekstraksi sebanyak dua belas fitur. Metode PCA digunakan untuk mengoptimasi kedua belas fitur GLCM yang diperoleh. Fitur GLCM yang telah dioptimasi digunakan sebagai masukan untuk proses klasifikasi. Metode klasifikasi pada penelitian ini adalah *multiclass* SVM dengan keluaran klasifikasi sebanyak tiga kelas yaitu, normal, TB primer (PTB) dan TB sekunder (STB). Sistem ini diharapkan mampu mengurangi faktor subjektivitas dalam proses deteksi abnormalitas paru-paru pada citra CXR.

1.2 Rumusan Masalah

Rumusan masalah yang diperoleh berdasar latar belakang yang telah dipaparkan sebelumnya adalah sebagai berikut:

1. Bagaimana desain sistem deteksi tuberculosi paru-paru pada citra CXR dengan menggunakan ekstraksi fitur dengan metode GLCM yang dioptimasi dengan metode PCA dan diklasifikasi dengan menggunakan metode SVM?
2. Bagaimana performansi sistem deteksi tuberculosi paru-paru pada CXR dengan menggunakan ekstraksi fitur dengan metode GLCM yang dioptimasi dengan metode PCA dan diklasifikasi dengan menggunakan metode SVM?
3. Bagaimana performansi sistem deteksi tuberculosi paru-paru pada CXR menggunakan ekstraksi fitur dengan metode GLCM yang dioptimasi dengan metode PCA bila dibandingkan dengan metode GLCM tanpa optimasi?

1.3 Batasan Masalah

Batasan penelitian yang digunakan pada penelitian ini agar terarah sesuai dengan tujuan penelitian adalah sebagai berikut:

1. Citra CXR yang diolah diperoleh dari Shenzhen *chest X-ray set*, Guangdong Medical College, Shenzhen, Cina.
2. Sistem deteksi abnormalitas paru-paru pada citra CXR didesain dengan bantuan program Matlab 2018a dan Weka 3.8.



3. Format citra digital yang diolah adalah .png.

4. Citra CXR yang diolah hanya citra CXR orang dewasa.

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk merancang bangun sistem deteksi tuberculosis paru-paru dari citra CXR dengan menggunakan metode SVM sebagai klasifikasinya dan fitur GLCM yang dioptimasi dengan metode PCA sebagai masukannya dan mengetahui performa dari sistem deteksi tuberculosis paru-paru yang telah dibangun dan membandingkan performa sistem yang telah dibangun dengan sistem yang sama tanpa metode optimasi PCA.

1.5 Manfaat Penelitian

Harapan dari hasil penelitian ini adalah berkurangnya faktor subyektifitas dalam proses pendeteksian abnormalitas paru-paru pada citra CXR. Harapan lainnya adalah hasil penelitian ini dapat digunakan sebagai referensi tambahan untuk mahasiswa bidang medis untuk mempelajari CXR.



BAB II TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Pawar dan Ganorkar (2016) mendesain sistem *screening* TB dengan menggunakan teknik pengolahan citra. Dalam penelitiannya, metode k-mean *clustering* digunakan untuk melakukan segmentasi pada citra CXR. Langkah selanjutnya adalah ekstraksi fitur GLCM dari citra CXR yang telah disegmentasi. Fitur yang diekstraksi antara lain, energi, homogenitas, korelasi, kontras, area, *eccentricity*, *major axis length*, *minor axis length*, *perimeter* dan *solidity*. Selanjutnya citra diklasifikasi dengan *Euclidean distance*. Penelitian ini menghasilkan sistem dengan sensitivitas sebesar 75%, spesifisitas sebesar 75% dan akurasi sebesar 75%. Namun pada penelitian ini tidak terdapat proses optimasi pada fitur GLCM, sehingga performansi sistem kurang optimal.

Mahajan dan Ganorkar (2017) mendesain sistem deteksi TB pada citra *X-Ray* dengan menggunakan bantuan *software* Matlab. Pada penelitian ini, citra melalui tahapan *preprocessing*, segmentasi, ekstraksi fitur dan klasifikasi. Pada tahap *preprocessing*, citra *X-Ray* diolah dengan melakukan perubahan ukuran, serta peningkatan kualitas citra. Selanjutnya untuk tahap segmentasi, digunakan metode segmentasi biner. Metode ekstraksi fitur yang digunakan adalah *Histogram of Oriented Gradient* (HOG). Selanjutnya pada tahap klasifikasi, digunakan *Support Vector Machine* (SVM). Hasil dari penelitian ini adalah sistem dengan sensitivitas 88%, spesifisitas 88% dan akurasi sebesar 88%. Penelitian ini dapat dioptimalkan dengan menggunakan fitur GLCM yang telah dioptimasi sebagai masukan pengganti dari HOG.

Dalam Tabel 2.1, ditunjukkan penelitian lain tentang pengolahan citra CXR untuk pendeteksian penyakit paru-paru.

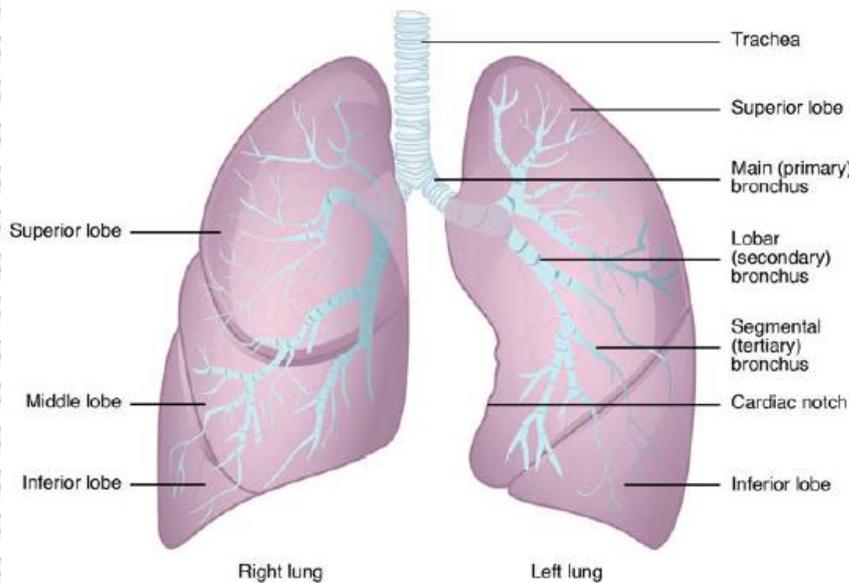
Tabel 2.1 *Penelitian terdahulu*

Tahun	2013	2016	2016	2016
Judul	<i>A Computer Based Feature Extraction of Lung Nodule in Chest X-Ray Image</i>	<i>Detection of Tuberculosis Using Chest X-Ray (CXR)</i>	<i>Automatic Detection of Major Lung Diseases Using Chest Radiographs and Classification by Feed-Forward Artificial Neural Network</i>	<i>Pneumonia Identification Using Organizing Map Algorithm</i>
Penulis	Nitin S. Lingayat, Manoj R. Tarambale	P. Sahana, T. Saravana Kumar, A. Risvin	Shubangi Khobragade, Aditya Tiwari, C. Y. Patil, Vikram Narke	Moh'd Rasoul A. Al-Hadidi, O. Dorgham, Rami Salim Razouq
Tujuan	Deteksi nodule (tumor)	Deteksi TB	Deteksi Penyakit Paru	Deteksi Pneumonia
Metode	Ekstraksi fitur GLCM	Fitur yang diekstraksi meliputi <i>skewness</i> , <i>kurtosis</i> , deviasi standar, area dan perimeter. Metode klasifikasi yang digunakan <i>Ada Boost</i>	Fitur yang diekstraksi meliputi area, perimeter, diameter, <i>irregularity index</i> , <i>mean</i> , deviasi standar dan entropi. Metode klasifikasi yang digunakan adalah <i>Feed Forward Artificial Network</i>	Fitur yang diekstraksi berupa histogram dari citra CXR yang telah disegmentasi. Metode klasifikasi yang digunakan adalah <i>Self Organizing Map (SOM)</i> . Neural network yang digunakan adalah <i>Vector Quantization Neural Network (LVQnet)</i>
Hasil	Tumor ganas memiliki nilai index regularitas yang mendekati nol, dan memiliki mean, varian, dan deviasi standar yang lebih tinggi dibandingkan dengan tumor jinak	Metode Ada Boost mampu mengklasifikasi citra CXR normal dan citra CXR TB dengan baik.	Akurasi 92%	Akurasi 97,45%
Kelebihan	Parameter yang diukur disajikan dengan jelas dalam bentuk tabel sehingga perbedaan ciri antara tumor ganas dan jinak dapat terlihat dengan jelas	Setiap tahapan proses preprocessing, segmentasi, ekstraksi fitur dan klasifikasi telah dijelaskan dengan ringkas.	Tahapan preprocessing, segmentasi, ekstraksi fitur serta klasifikasi telah dijelaskan dengan baik.	Tahapan preprocessing, segmentasi, ekstraksi fitur serta klasifikasi telah dijelaskan dengan baik.
Kelamahan	Penulis hanya meneliti ciri fitur tekstur pada tumor. Penelitian dapat dikembangkan dengan menambahkan metode klasifikasi untuk membedakan citra CXR normal dengan CXR kanker	Hasil penelitian hanya ditampilkan dalam bentuk grafik dan tidak ditampilkan dalam bentuk tabel, sehingga nilai aktual dari performansi sistem yang diajukan kurang jelas. Metode klasifikasi yang diusulkan pada penelitian ini perlu pembahasan lebih detail.	Tidak ada perhitungan sensitifitas dan spesifisitas agar performansi sistem dapat diketahui dengan lebih detail. Fitur yang diekstraksi dapat dilengkapi dengan fitur GLCM yang dioptimasi agar performansi sistem lebih optimal.	Tidak ada perhitungan sensitifitas dan spesifisitas agar performansi sistem dapat diketahui dengan lebih detail.



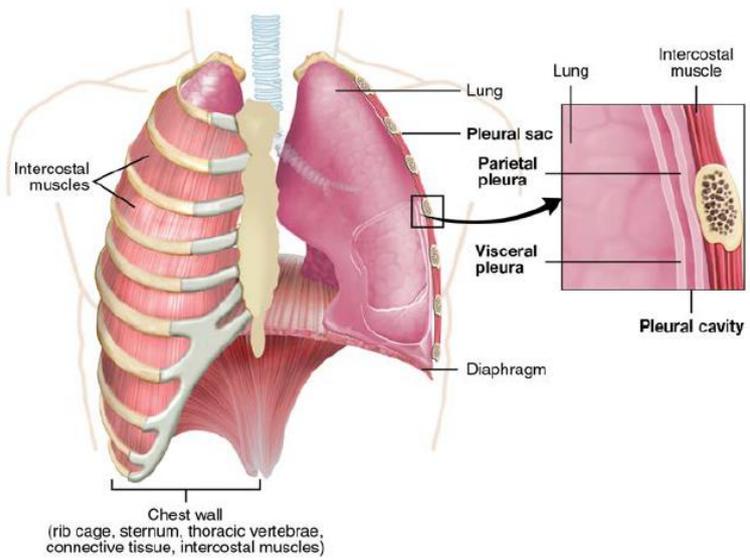
2.2 Paru-paru

Organ penting dalam sistem pernafasan yang berfungsi untuk membawa oksigen dari udara ke dalam darah dan membuang karbon dioksida dari darah adalah paru-paru (Betts dkk., 2013). Pertukaran gas oksigen dengan gas karbon dioksida pada paru-paru terjadi pada alveoli. Paru-paru memiliki bentuk seperti dua buah pyramid yang terhubung dengan trachea pada bagian bronchi kiri dan kanan (*Gambar 2.1*). Paru-paru sebelah kanan memiliki bentuk lebih pendek, lebih lebar dan memiliki volume lebih besar bila dibandingkan dengan paru-paru sebelah kiri.



Gambar 2.1 Anatomi paru-paru manusia
Sumber: Betss dkk. (2013)

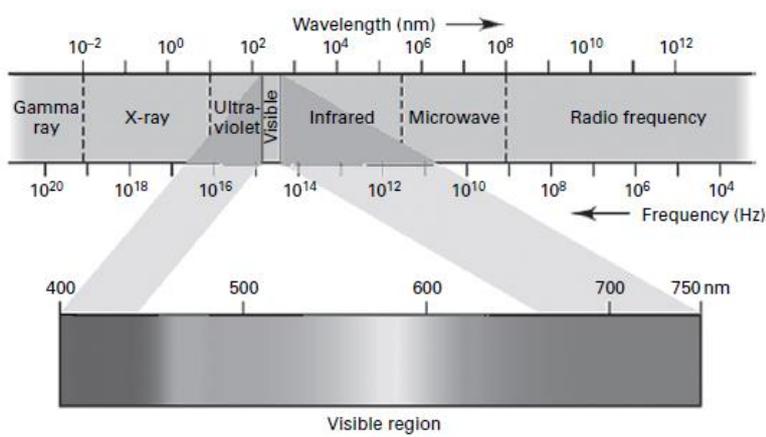
Paru-paru dibungkus oleh pleura (*Gambar 2.2*). Pleura memiliki dua fungsi utama, yaitu menghasilkan cairan pleural dan membentuk rongga yang memisahkan organ-organ utama untuk menghindari terjadinya interverensi saat organ bergerak. Cairan pleural berfungsi sebagai pelumas untuk mengurangi gesekan antar organ saat bernafas.



Gambar 2.2 Pleura
 Sumber: Betss dkk. (2013)

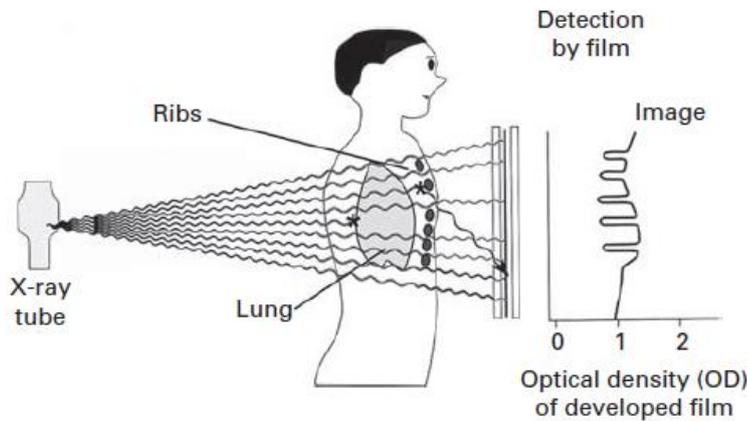
2.3 Radiografi

Radiografi merupakan metode untuk menggambarkan struktur internal dari seorang pasien dengan menggunakan X-ray (WHO, 2016). X-ray merupakan suatu bentuk radiasi elektromagnetik yang dihasilkan oleh sebuah X-ray tube. X-ray merupakan bagian dari spektrum elektromagnetik yang memiliki panjang gelombang lebih pendek dibandingkan dengan ultraviolet maupun cahaya nampak (Gambar 2.3). Karena panjang gelombangnya yang pendek, x-ray mampu menembus tubuh manusia (Dhawan, 2011).



Gambar 2.3 Spektrum gelombang elektromagnetik
 Sumber: Dougherty (2009)

X-ray menembus tubuh pasien dan ditangkap di belakang pasien dengan menggunakan sebuah film yang sensitif terhadap *X-ray* atau ditangkap dengan menggunakan sebuah detektor digital. Metode pengambilan CXR terlihat dalam *Gambar 2.4*.



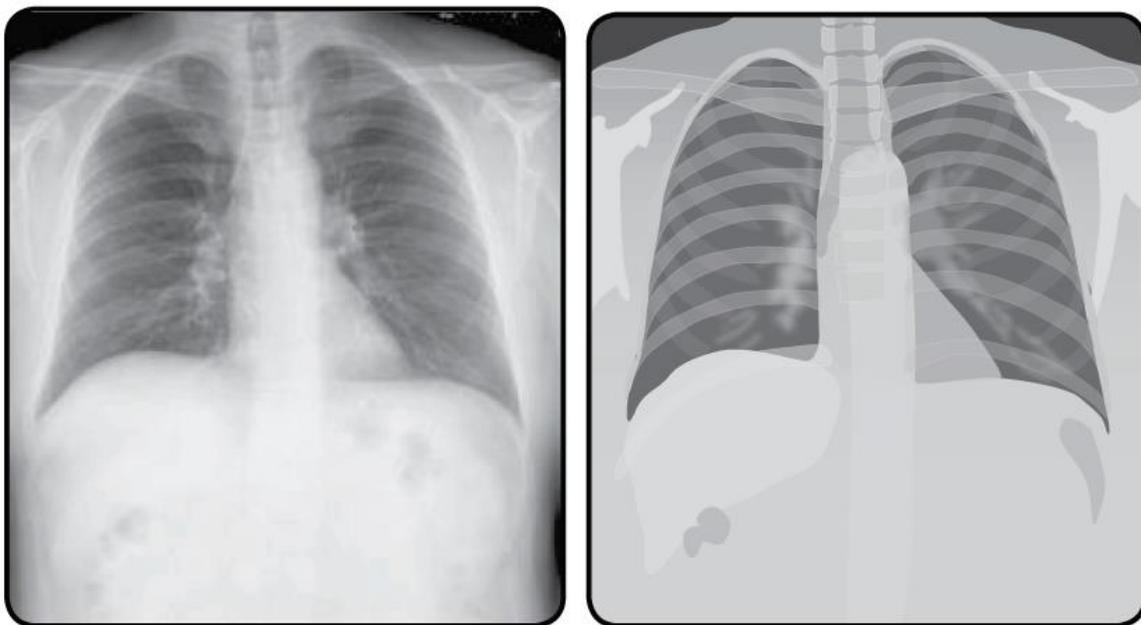
Gambar 2.4 Pengambilan citra CXR
Sumber: Dougherty (2009)

Jaringan tubuh pada tubuh manusia menyerap *X-ray* dengan level yang berbeda-beda. Tulang yang padat menyerap radiasi lebih banyak, sedangkan untuk jaringan yang lebih lunak ditembus oleh radiasi *X-Ray* (emedicinehealth.com, 2018). Tulang, besi dan benda padat lainnya menghasilkan warna putih pada citra CXR. Air, jantung, diafragma, pembuluh darah dan jaringan lunak lainnya menghasilkan warna abu-abu pada CXR. Udara menghasilkan warna hitam pada CXR. Radiasi *X-ray* yang diserap dan yang dilewatkan oleh tubuh manusia menghasilkan variasi kontras pada citra CXR seperti dalam *Gambar 2.5*. Variasi kontras dapat menunjukkan adanya logam berat (*lead*), senyawa BaSO_4 (*Barium Sulfat* yang digunakan sebagai media kontras), struktur tulang (*bone*), jaringan otot (*muscle*) dan darah (*blood*), hati (*liver*), lemak (*fat*) dan udara (*air*). Variasi kontras tersebut memberikan representasi dua dimensi dari struktur tiga dimensi (WHO, 2016). Salah satu aplikasi *X-ray* pada bidang klinis yang umum digunakan adalah pencitraan dada untuk menganalisa kondisi paru-paru.

Structure	Lead	BaSO ₄	Bone	Muscle, blood	Liver	Fat	Air
							
	↓	↓	↓	↓	↓	↓	↓
Brightness on radiograph							

Gambar 2.5 Variasi kontras pada citra X-Ray
 Sumber: Hoffer (2006)

Chest X-ray (CXR) merupakan teknik pencitraan yang mampu mengidentifikasi abnormalitas pada paru-paru. (emedicinehealth.com, 2018). CXR digunakan untuk mendiagnosis kondisi rongga dada, termasuk jalan nafas, rusuk, paru-paru, jantung dan diafragma. Contoh citra CXR dapat dilihat dalam Gambar 2.6. CXR mampu mengenali beberapa kondisi abnormal pada rongga dada seperti pneumonia, pembesaran jantung (*cardiomegaly*), gagal jantung kongestif, *mass* pada paru-paru (dapat berupa kanker atau *nodule*), hancurnya kantung udara paru-paru (*emfisema*), tuberculosis (TB), patah tulang rusuk dan tulang belakang, cairan disekitar paru (*pleural effusion*), dan adanya udara disekitar paru (*pneumothorax*).



Gambar 2.6 Citra CXR normal
 Sumber: Cleeff dkk., (2010)

Citra CXR paru-paru normal adalah citra CXR yang mampu menunjukkan bidang paru-paru dengan jelas tanpa adanya tanda-tanda abnormalitas seperti adanya citra berbentuk jaring-jaring kasar, kabut, ruang kosong, maupun adanya bagian paru-paru yang kolaps (WHO, 2011). Pada CXR paru-paru normal, rongga dada dibatasi oleh struktur tulang berwarna putih yang merepresentasikan tulang rusuk. Pada bagian atas dada terdapat leher dan tulang selangka. Pada bagian bawah, rongga dada dan bagian perut (abdomen) dibatasi oleh diafragma. Di dalam rongga dada, kolom-kolom tulang belakang dapat terlihat jelas ditengah dada. Di tiap sisi dari garis tengah terlihat bidang paru-paru yang ditandai oleh warna hitam. Bayangan putih yang menunjukkan jantung terletak ditengah bagian bawah, diatas diafragma, dan sedikit condong ke kiri.

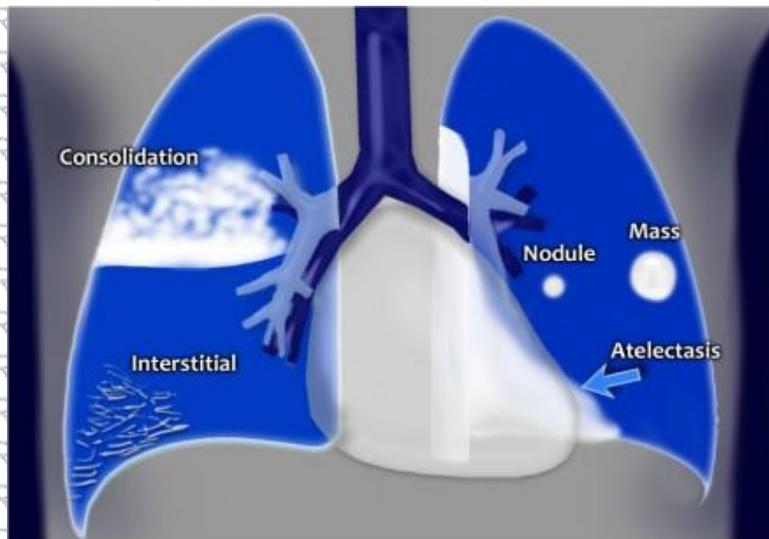
Citra CXR pria, wanita, dewasa dan anak-anak memiliki perbedaan. Perbedaan citra CXR pada pria dan wanita disebabkan karena adanya perbedaan jumlah jaringan pada payudara (Avinash M dkk., 2014). Jaringan payudara pada wanita menyerap sinar *X-ray* lebih banyak yang menyebabkan jaringan paru-paru yang berada dibelakang payudara nampak lebih putih. CXR pada anak-anak memiliki perbedaan mendasar dengan CXR pada orang dewasa, yaitu CXR pada anak-anak berubah-ubah sesuai dengan usia (Sage & van Damme, 2017). CXR pada anak-anak memiliki area pengambilan citra yang lebih kecil dibandingkan dengan sensor pada mesin *x-ray* (Kareem & Mohaideen, 2017). Citra CXR pada anak-anak memerlukan pengukuran ketebalan bagian tubuh untuk menentukan faktor paparan sinar *x-ray*. Perbedaan yang paling utama dari pengambilan citra CXR pada anak-anak dan dewasa adalah faktor keamanan dari radiasi *x-ray* (Kareem & Mohaideen, 2017). Anak-anak sangat sensitif terhadap radiasi *x-ray*, sehingga paparan *x-ray* pada pengambilan citra CXR anak-anak diatur dalam kondisi minimum dan dilakukan tanpa perlu melakukan banyak pengulangan.

Abnormalitas pada paru-paru dapat dikenali melalui citra CXR. Abnormalitas pada paru-paru umumnya ditandai dengan meningkatnya densitas (warna keputih-putihan) atau berkurangnya densitas (warna kehitam-hitaman) pada bidang paru-paru citra CXR (radiologyassistant.nl, 2018). Abnormalitas pada bidang paru citra CXR ditandai dengan adanya citra berbentuk jaring-jaring kasar, kabut, ruang kosong, maupun adanya bagian paru-paru yang kolaps.

Abnormalitas paru-paru pada citra CXR dapat diketahui dengan menggunakan metode pendekatan pola. Metode pendekatan pola membagi pola abnormalitas pada citra

CXR menjadi empat pola yaitu *consolidation*, *interstitial*, *nodule/mass*, dan *atelectasis* seperti dalam *Gambar 2.7* dan dapat dijabarkan sebagai berikut :

- *Consolidation*: terisinya alveoli dengan cairan, nanah, darah, sel (dapat berupa sel tumor) atau zat lainnya sehingga menghasilkan bayangan tidak jelas (kabut putih) pada bidang paru CXR.
- *Interstitial*: jaringan pendukung paru-paru yang menghasilkan bentuk seperti jaring/anyaman pada bidang paru CXR.
- *Nodule* atau *mass*: adanya ruang kosong yang menempati bidang paru pada citra CXR.
- *Atelectasis* : kolapsnya bagian dari paru-paru karena penurunan jumlah udara pada alveoli yang mengakibatkan bertambahnya densitas (warna keputih-putihan) pada bidang paru CXR.



Gambar 2.7 Pola abnormalitas paru pada citra CXR

Sumber: <http://radiologyassistant.nl/> (2018)

2.4 Tuberculosis (TB)

TB merupakan penyakit infeksi dengan bakteri MTB sebagai penyebabnya dan umumnya menyerang paru-paru. Gejala TB umumnya adalah batuk berdahak berkepanjangan, penurunan berat badan, hilangnya nafsu makan, pegal-pegal, sesak nafas, dada terasa sakit, demam, dan berkeringat di malam hari (Varaine & Rich, 2014).

Bakteri MTB menyebar dari penderita TB ke orang lain melalui udara. Penderita TB menyebarkan bakteri MTB dalam bentuk percikan dahak yang berukuran sangat kecil ketika

sedang batuk, berbicara atau bersin. TB akan menular bila percikan dahak MTB dihirup oleh orang lain.

Terdapat empat tahapan perjalanan alamiah penyakit TB pada manusia. Tahapan perjalanan alamiah TB dijabarkan dalam Tabel 2.2

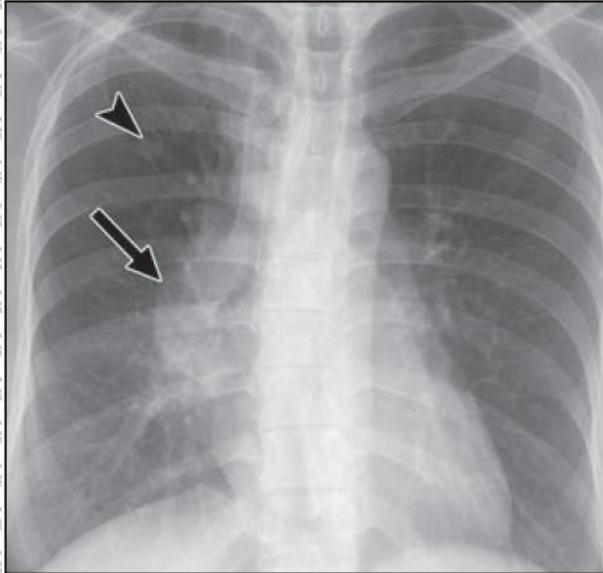
Tabel 2.2 Tahapan perjalanan alamiah penyakit TB

a. Paparan	
Peluang peningkatan paparan terkait dengan:	<ul style="list-style-type: none"> ➢ Jumlah kasus menular di masyarakat ➢ Peluang kontak dengan kasus menular ➢ Tingkat daya tular dahak sumber penularan ➢ Intensitas batuk sumber penularan ➢ Kedekatan kontak dengan sumber penularan ➢ Lamanya waktu kontak dengan sumber penularan ➢ Faktor lingkungan: konsentrasi kuman diudara (ventilasi, sinar ultra violet, penyaringan adalah faktor yang dapat menurunkan konsentrasi)
Catatan: Paparan kepada pasien TB menular merupakan syarat untuk terinfeksi. Setelah terinfeksi, ada beberapa faktor yang menentukan seseorang akan terinfeksi saja, menjadi sakit dan kemungkinan meninggal dunia karena TB.	
b. Infeksi	
Reaksi daya tahan tubuh akan terjadi setelah 6 – 14 minggu setelah infeksi	
<ul style="list-style-type: none"> ➢ Reaksi imunologi (lokal) Kuman TB memasuki alveoli dan ditangkap oleh makrofag dan kemudian berlangsung reaksi antigen – antibody. ➢ Reaksi imunologi (umum) <i>Delayed hypersensitivity</i> (hasil Tuberkulin tes menjadi positif) 	
<ul style="list-style-type: none"> ➢ Lesi umumnya sembuh total namun dapat saja kuman tetap hidup dalam lesi tersebut (<i>dormant</i>) dan suatu saat dapat aktif kembali. ➢ Penyebaran melalui aliran darah atau getah bening dapat terjadi sebelum penyembuhan lesi 	
c. Sakit TB	
Faktor risiko untuk menjadi sakit TB adalah tergantung dari :	<ul style="list-style-type: none"> ➢ Konsentrasi / jumlah kuman yang terhirup ➢ Lamanya waktu sejak terinfeksi ➢ Usia seseorang yang terinfeksi ➢ Tingkat daya tahan tubuh seseorang. Seseorang dengan daya tahan tubuh yang rendah diantaranya infeksi HIV/AIDS dan malnutrisi (gizi buruk) akan memudahkan berkembangnya TB aktif (sakit TB). Bila jumlah orang terinfeksi HIV meningkat, maka jumlah pasien TB akan meningkat, dengan demikian penularan TB di masyarakat akan meningkat pula.
Catatan: Hanya sekitar 10% yang terinfeksi TB akan menjadi sakit TB. Namun bila seorang dengan HIV positif akan meningkatkan kejadian TB melalui proses reaktivasi. TB umumnya terjadi pada paru (TB Paru). Namun, penyebaran melalui aliran darah atau getah bening dapat menyebabkan terjadinya TB diluar organ paru (TB Ekstra Paru). Apabila penyebaran secara masif melalui aliran darah dapat menyebabkan semua organ tubuh terkena (TB miliar).	
d. Meninggal dunia	
Faktor risiko kematian karena TB:	<ul style="list-style-type: none"> ➢ Akibat dari keterlambatan diagnosis ➢ Pengobatan tidak adekuat ➢ Adanya kondisi kesehatan awal yang buruk atau penyakit penyerta
Catatan: Pasien TB tanpa pengobatan, 50% akan meninggal dan risiko ini meningkat pada pasien dengan HIV positif.	

Sumber: Kementerian Kesehatan Republik Indonesia (2014,p.3-4)

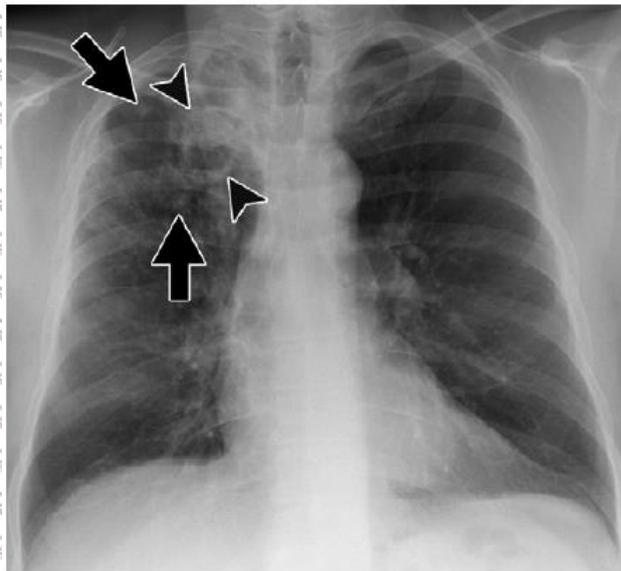
TB memiliki dua bentuk yaitu primer dan sekunder (Hashimoto dkk., 2000).

Penderita yang baru terjangkit TB disebut sebagai TB primer (PTB), sedangkan penderita yang terjangkit TB dari penyakit TB sebelumnya yang aktif kembali disebut sebagai TB sekunder atau *post primary* TB (STB) (Jeong & Lee, 2008). PTB ditandai dengan abnormalitas berupa pembesaran kelenjar pada paru-paru seperti dalam *Gambar 2.8*. STB ditandai keberadaan *consolidation* pada lobus superior atau pada lobus bawah bagian atas pada citra CXR seperti dalam *Gambar 2.9*.



Gambar 2.8 TB primer

Sumber: Jeong & Lee (2008).



Gambar 2.9 TB sekunder

Sumber: Nachiappan dkk., (2017).

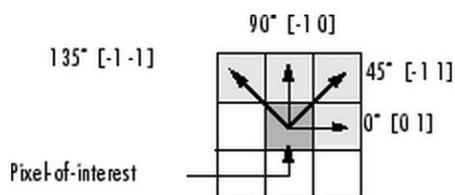
2.5 GLCM

GLCM merupakan tabulasi frekuensi kombinasi dari nilai tingkat keabuan yang terjadi dalam sebuah citra (Hall-Beyer, 2017). GLCM mencirikan tekstur suatu citra dengan menghitung seberapa sering pasangan piksel dengan nilai dan hubungan spasial tertentu terjadi dalam citra tersebut. Nilai pasangan piksel tersebut disusun ke dalam sebuah matriks, selanjutnya fitur statistik diperoleh dari perhitungan matriks tersebut. Proses pembentukan matriks pada GLCM ditunjukkan dalam *Gambar 2.10*. Matriks pada GLCM disusun oleh orientasi spasial dari sudut 0° , 45° , 90° dan 135° . Orientasi spasial pembentuk GLCM ditunjukkan dalam *Gambar 2.11*.

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

Gambar 2.10 Pembentukan GLCM

Sumber: Matlab (2018)



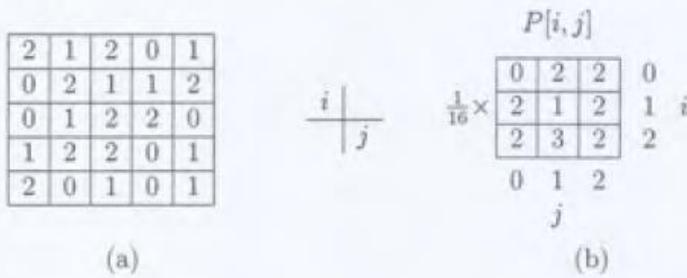
Gambar 2.11 Orientasi spasial pembentuk GLCM

Sumber: Matlab (2018)

GLCM $P[i, j]$ didefinisikan dengan terlebih dahulu menentukan vektor perpindahan $d = (dx, dy)$ dan menghitung semua pasangan piksel yang dipisahkan oleh d dan dengan tingkat keabu-abuan i dan j (Jian dkk., 1995). Sebagai contohnya, dimisalkan terdapat sebuah citra dengan dimensi 5×5 dengan tingkat keabuan 0, 1 dan 2 seperti dalam *Gambar (2.12)(a)*. Tingkat keabuan pada citra ada tiga sehingga matriks $P[i, j]$ berukuran 3×3 .

Vektor perpindahan yang digunakan adalah $d = (1, 1)$, yang memiliki interpretasi satu piksel ke kanan dan satu piksel ke bawah. Perhitungan matriks $P[i, j]$ dilakukan dengan cara

menghitung semua pasangan piksel (piksel awal i , piksel selanjutnya dengan arah d memiliki nilai j). Hasil perhitungan matriks $P[i, j]$ ditunjukkan dalam Gambar (2.12)(b).



Gambar 2.12 Perhitungan matriks GLCM (a) Sebuah citra berdimensi 5x5 dengan tiga tingkat keabuan (0, 1, 2). (b) GLCM dengan $d = (1, 1)$
 Sumber: Jian dkk. (1995)

Matriks $P[i, j]$ terkadang asimetris. Matriks dinormalisasi dengan membagi setiap masukan dengan jumlah total pasangan piksel. Nilai $P[i, j]$ yang ternormalisasi digunakan sebagai fungsi probabilitas massa.

Fitur untuk menghitung keacakan dari distribusi tingkat keabuan adalah entropi. Entropi dapat dinotasikan seperti dalam Persamaan (2-1). Entropi akan memiliki nilai tertinggi saat semua masukan pada $P[i, j]$ bernilai sama. Entropi dengan nilai tertinggi menandakan matriks pada citra tidak memiliki pasangan keabuan yang dominan pada vektor d (Jian dkk., 1995).

$$Entropi = - \sum_i \sum_j P[i, j] \log P[i, j] \dots \dots \dots (2-1)$$

Fitur lain yang dapat dihitung adalah energi (Persamaan 2-2), kontras (Persamaan 2-3), homogenitas (Persamaan 2-4) dan korelasi (Persamaan 2-5) (Adi dkk., 2018). Energi merupakan ukuran keseragaman lokal pada tingkat keabuan. Kontras merupakan ukuran dari intensitas variasi tingkat keabuan antara piksel referensi terhadap piksel tetangganya.

Homogenitas merupakan ukuran distribusi dari elemen pada GLCM terhadap arah diagonal GLCM. Korelasi merupakan ukuran hubungan piksel referensi terhadap tetangganya dalam sebuah citra, dimana μ_x, μ_y sebagai *mean* dan σ_x, σ_y sebagai deviasi standar.

$$Energi = \sum_i \sum_j P^2[i, j] \dots \dots \dots (2-2)$$

$$Kontras = \sum_i \sum_j (i - j)^2 P[i, j] \dots \dots \dots (2-3)$$

$$Homogenitas = \sum_i \sum_j \frac{P[i, j]}{1 + |i - j|} \dots \dots \dots (2-4)$$

$$\text{Korelasi} = \sum_i \sum_j \frac{ijP_d(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y} \dots \dots \dots (2-5)$$

Vektor perpindahan d yang dipilih merupakan parameter yang penting dalam definisi GLCM. Untuk memaksimalkan nilai statistik dari $P[i, j]$, GLCM dapat dihitung dengan menggunakan beberapa nilai d .

2.6 Principal Component Analysis (PCA)

Ide utama pada PCA adalah untuk mengurangi dimensi dari kumpulan data yang terdiri dari banyak variabel yang saling berhubungan dan dengan mempertahankan sebanyak mungkin variasi yang ada dalam kumpulan data (Jolliffe, 2002). Hal ini dicapai dengan mengubah variabel yang ada menjadi kumpulan variabel baru dengan *principal components* (PC) yang tidak berhubungan dan diurutkan sedemikian sehingga diperoleh variabel pertama yang paling banyak memiliki variasi yang ada pada variabel asli. Metode PCA dapat digunakan untuk analisa data dan pengenalan pola yang sering digunakan dalam pengolahan sinyal dan citra sebagai teknik kompresi data maupun reduksi data (Mudrova, 2015).

Dimisalkan terdapat vektor x dari variabel acak p , dan yang perlu diperhatikan adalah varian dari variabel acak p serta struktur dari kovarian atau korelasi antar variabel p . Bila p berukuran kecil atau memiliki struktur yang sederhana, tidak dianjurkan untuk memperhatikan varian p dan semua korelasi atau kovarian dari $\frac{1}{2}p(p-1)$. Pendekatan alternatif yang dapat digunakan adalah dengan mencari beberapa variabel turunan ($\ll p$) yang menyimpan banyak informasi dari varian dan korelasi atau kovarian.

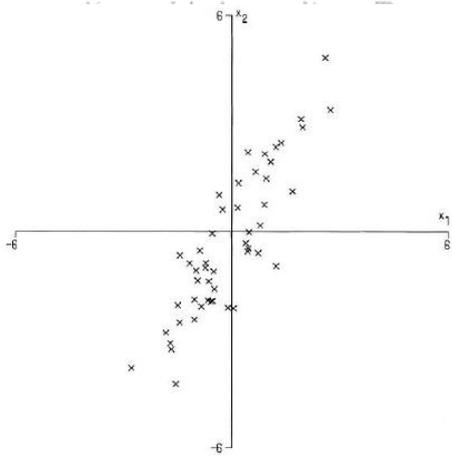
PCA berkonsentrasi pada varian dengan tidak mengabaikan kovarian dan korelasi. Langkah pertama yang dilakukan adalah dengan mencari fungsi linier $\alpha'_1 x$ dari elemen x yang memiliki nilai varian maksimum, dimana α_1 merupakan vektor dari p dengan konstanta $\alpha_{11}, \alpha_{12}, \dots, \alpha_{1p}$ dan α menotasikan transposisi, sehingga diperoleh persamaan 2-6.

$$\alpha'_1 x = \alpha_{11}x_1 + \alpha_{12}x_2 + \dots + \alpha_{1p}x_p = \sum_{j=1}^p \alpha_{1j}x_j \dots \dots \dots (2-6)$$

Langkah selanjutnya adalah mencari fungsi linier $\alpha'_2 x$ yang tidak berkorelasi dengan $\alpha'_1 x$ dan yang memiliki varian maksimum, hingga pada tahap ke- k diperoleh fungsi linier $\alpha'_k x$ yang memiliki subyek varian maksimum yang tidak berkorelasi dengan $\alpha'_1 x, \alpha'_2 x, \dots, \alpha'_{k-1} x$. Fungsi linier $\alpha'_k x$ merupakan PC ke- k pada variabel turunan ke- k . Sebagian besar

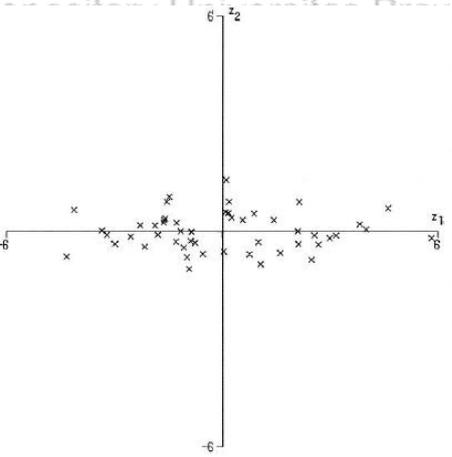
variasi dari x akan diwakili oleh PC, dimana $m \ll p$. Pengurangan kompleksitas dapat dicapai dengan mengubah variabel asli ke PC, untuk kasus sederhana dapat menggunakan $p = 2$. Kelebihan dari penggunaan $p = 2$ adalah data dapat diplot dalam bentuk dua dimensi.

Dalam *Gambar 2.13* terdapat plot dari 50 observasi pada dua variabel yang sangat berkorelasi x_1, x_2 . Terdapat variasi yang cukup besar pada kedua variabel yang cenderung ke arah x_2 dibanding ke arah x_1 . Bila ditransformasikan ke dalam PC z_1, z_2 , maka diperoleh plot seperti dalam *Gambar 2.14*.



Gambar 2.13 Plot dari 50 observasi pada dua variabel x_1, x_2 .

Sumber: Jolliffe (2002)



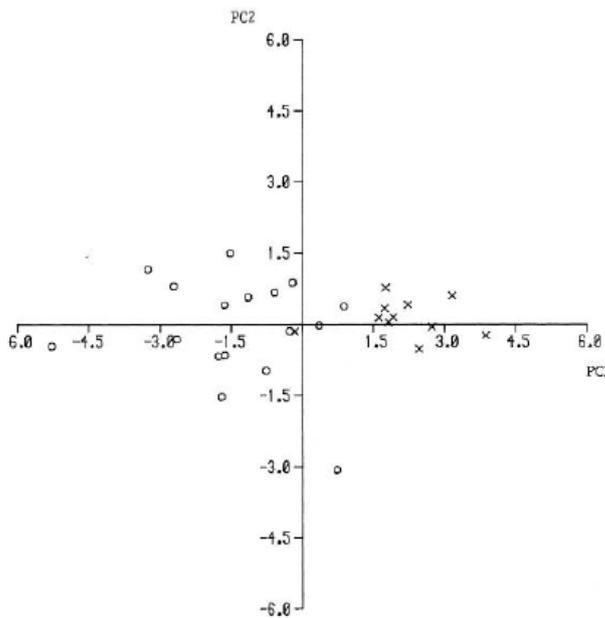
Gambar 2.14 Plot dari 50 observasi dalam bentuk PC z_1, z_2 .

Sumber: Jolliffe (2002)

Terlihat dengan jelas bahwa lebih banyak variasi pada arah z_1 dibandingkan pada kedua variabel asal, dan terdapat sedikit variasi pada arah z_2 . Jika satu set variabel $p (> 2)$

memiliki korelasi substansial diantara mereka, maka beberapa PC pertama akan mewakili sebagian besar variabel pada variabel asli. Sebaliknya, beberapa PC terakhir yang mengidentifikasi arah di mana terdapat sedikit variasi, maka PC tersebut mengidentifikasi hubungan linear yang hampir konstan di antara variabel asli.

Gambar 2.15 menunjukkan plot dari nilai dua PC pertama pada contoh kasus dengan 7 variabel. Data yang dipresentasikan terdiri dari tujuh pengukuran anatomi pada 28 siswa, dengan komposisi 11 perempuan dan 17 laki-laki. Dua PC pertama mewakili 80 persen dari variasi total pada data set, sehingga *Gambar 2.15* merupakan representasi yang sesuai untuk 28 observasi pada bidang 7 dimensi. Dalam *Gambar 2.15* terlihat jelas bahwa PC pertama mampu memisahkan sampel laki-laki dan perempuan.



Gambar 2.15 Pengukuran anatomi siswa: plot 28 siswa terhadap dua PC pertama. x menotasikan perempuan; o menotasikan laki-laki.

Sumber: Jolliffe (2002)

Setelah mendefinisikan PC, maka perlu juga untuk mengetahui cara mencarinya. Dimisalkan, terdapat vektor dari variable acak x yang diketahui memiliki matriks kovarian Σ . Matriks ini merupakan matriks dengan elemen ke (i, j) -nya diketahui sebagai kovarian antara elemen x ke- i dan ke- j ketika $i \neq j$, dan varian elemen x ke- j ketika $i = j$. Pada kasus yang lebih realistis dimana nilai Σ tidak diketahui, maka nilai Σ dapat digantikan dengan nilai sampel matriks kovarian S . Pada $k = 1, 2, \dots, p$ nilai PC ke- k mengikuti persamaan $z_k = \alpha'_k x$ dimana nilai α_k merupakan *eigenvector* dari Σ terhadap *eigenvalue* λ_k

tertinggi ke- k . Bila α_k yang dipilih memiliki panjang ($\alpha_k' \alpha_k = 1$), maka $\text{var}(z_k) = \lambda_k$, dimana $\text{var}(z_k)$ menotasikan varian dari z_k .

Untuk memperoleh PC, maka perlu diperhatikan $\alpha_1' x$; dimana vektor α_1 memaksimalkan $\text{var}[\alpha_1' x] = \alpha_1' \Sigma \alpha_1$. Nilai maksimum tidak akan diperoleh pada nilai α_1 yang terbatas maka diperlukan adanya batasan normalisasi. Batasan yang digunakan dalam penurunan adalah $\alpha_1' \alpha_1 = 1$, sehingga jumlah dari elemen α_1 kuadrat sama dengan 1. Batasan lain yang dapat digunakan untuk kondisi lain adalah $\text{Max}_j |\alpha_{ij}| = 1$. Penggunaan batasan lain selain $\alpha_1' \alpha_1 = \text{konstant}$ dalam penurunan akan mempersulit permasalahan optimasi dan menghasilkan satu set variabel turunan yang berbeda dari PC.

Untuk memaksimalkan $\alpha_1' \Sigma \alpha_1$ dengan $\alpha_1' \alpha_1 = 1$, digunakan pendekatan standar dengan menggunakan pengali Lagrange. Memaksimalkan sesuai dengan Persamaan (2-7), dimana λ adalah pengali Lagrange. Diferensiasi terhadap α_1 sehingga diperoleh Persamaan (2-8) atau (2-9). Dimana I_p merupakan matriks identitas ($p \times p$). Sehingga, λ adalah *eigenvalue* dari Σ dan α_1 sebagai *eigenvector*-nya.

$$\alpha_1' \Sigma \alpha_1 - \lambda(\alpha_1' \alpha_1 - 1), \dots \dots \dots (2-7)$$

$$\Sigma \alpha_1 - \lambda \alpha_1 = 0, \dots \dots \dots (2-8)$$

$$(\Sigma - \lambda I_p) \alpha_1 = 0, \dots \dots \dots (2-9)$$

Untuk menentukan *eigenvector* p yang menghasilkan $\alpha_1' x$ dengan varian maksimal, dengan catatan bahwa kuantitas yang dimaksimalkan sesuai dengan persamaan (2-10), Maka λ harus memiliki nilai sebesar mungkin.

$$\alpha_1' \Sigma \alpha_1 = \alpha_1' \lambda \alpha_1 = \lambda \alpha_1' \alpha_1 = \lambda \dots \dots \dots (2-10)$$

α_1 adalah *eigenvector* dari *eigenvalue* terbesar dari Σ , dan $\text{var}(\alpha_1' x) = \alpha_1' \Sigma \alpha_1 = \lambda_1$, sebagai *eigenvalue* terbesar.

Secara umum, PC ke- k dari x adalah $\alpha_k' x$ dan $\text{var}(\alpha_k' x) = \lambda_k$, dimana λ_k adalah *eigenvalue* terbesar ke- k dari Σ , dan α_k mewakili *eigenvector*. Hal ini terbukti pada $= 2$, sedangkan untuk $k \geq 3$ akan sedikit lebih rumit namun mirip.

Pada PC kedua, $\alpha_2' x$ memaksimalkan $\alpha_2' \Sigma \alpha_2$ yang tidak berkorelasi dengan $\alpha_1' x$, atau sama dengan $\text{cov}[\alpha_1', \alpha_2'] = 0$, dimana $\text{cov}(x, y)$ menotasikan kovarian antara variabel



acak x dan y . Namun sesuai pada Persamaan (2-11) maka, salah satu dari Persamaan (2-12), (2-13), (2-14) atau (2-15) dapat menentukan korelasi *zero* antara $\alpha_1'x$ dan $\alpha_2'x$.

$$Cov[\alpha_1'x, \alpha_2'x] = \alpha_1' \Sigma \alpha_2 = \alpha_2' \Sigma \alpha_1 = \alpha_2' \lambda_1 \alpha_1' = \lambda_1 \alpha_2' \alpha_1 = \lambda_1 \alpha_1' \alpha_2 \dots \dots \dots (2-11)$$

$$\alpha_1' \Sigma \alpha_2 = 0 \dots \dots \dots (2-12)$$

$$\alpha_2' \Sigma \alpha_1 = 0 \dots \dots \dots (2-13)$$

$$\alpha_1' \alpha_2 = 0 \dots \dots \dots (2-14)$$

$$\alpha_2' \alpha_1 = 0 \dots \dots \dots (2-15)$$

Dengan memilih persamaan terakhir (dengan pemilihan sembarang) dan dengan catatan bahwa normalisasi tidak lagi diperlukan, maka kuantitas yang dimaksimalkan sesuai dengan Persamaan (2-16), dimana λ, \emptyset adalah pengali Langrange. Dideferensiasi terhadap α_2 diperoleh sesuai dengan Persamaan (2-17). Perkalian dari Persamaan (2-17) terhadap α_1 diperoleh Persamaan (2-18).

$$\alpha_2' \Sigma \alpha_2 - \lambda(\alpha_2' \alpha_2 - 1) - \emptyset \alpha_2' \alpha_1 \dots \dots \dots (2-16)$$

$$\Sigma \alpha_2 - \lambda \alpha_2 - \emptyset \alpha_1 = 0 \dots \dots \dots (2-17)$$

$$\alpha_1' \Sigma \alpha_2 - \lambda \alpha_1' \alpha_2 - \emptyset \alpha_1' \alpha_1 = 0 \dots \dots \dots (2-18)$$

λ pada $\lambda = \alpha_2' \Sigma \alpha_2$ harus memiliki nilai sebesar mungkin. Dengan mengasumsikan bahwa Σ tidak mengulangi *eigenvalue* maka λ tidak dapat sama dengan λ_1 . Maka λ merupakan *eigenvalue* terbesar kedua dari Σ , dan α_2 merupakan *eigenvector*.

Turunan dari koefisien dan varian PC sebagai *eigenvector* dan *eigenvalue* dari matriks kovarian pada Persamaan (2-19) merupakan standar.

$$var[\alpha_k'x] = \lambda_k \quad \text{untuk } k = 1, 2, \dots, p \dots \dots \dots (2-19)$$

2.7 SVM

SVM menghasilkan fungsi pemetaan *input-output* dari sebuah set data *training* yang telah dilengkapi dengan label (Wang, 2005). SVM yang dilengkapi dengan sampel pelatihan mampu membuat *hyperplane* yang memiliki margin pemisah antara dua keputusan dengan maksimal (Haykin, 2008). Vektor *support* terdiri dari poin data yang diperoleh dari

sampel pelatihan. SVM dapat digunakan untuk menyelesaikan permasalahan klasifikasi pola maupun permasalahan regresi non-linear.

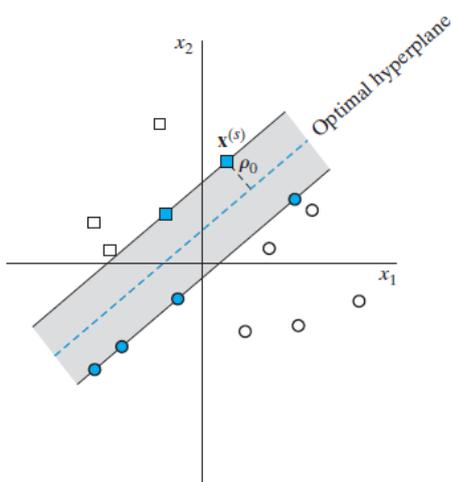
Dimisalkan terdapat sampel pelatihan $\{(x_i, d_i)\}_{i=1}^N$, dimana x_i merupakan pola masukan untuk contoh ke- i dan d_i adalah respon yang diharapkan (target keluaran). Diasumsikan terdapat dua pola (*class*) *linearly separable* direpresentasikan oleh $d_i = +1$ dan $d_i = -1$. *Hyperplane* yang digunakan untuk memisahkan kedua pola sesuai dengan Persamaan (2-20).

$$w^T x + b = 0 \quad \text{..... (2.20)}$$

Dengan x sebagai vektor masukan, w sebagai vektor bobot yang dapat disesuaikan, dan b sebagai bias. Maka Persamaan (2-20) dapat ditulis kembali ke dalam bentuk Persamaan (2-21)

$$\begin{aligned} w^T x + b &\geq 0 \text{ untuk } d_i = +1 \\ w^T x + b &\leq 0 \text{ untuk } d_i = -1 \end{aligned} \quad \text{..... (2-21)}$$

Untuk vektor bobot w dan bias b , pemisah antara *hyperplane* didefinisikan oleh persamaan (2.20) dan titik data terdekat yang disebut sebagai margin pemisah dinotasikan oleh ρ . Tujuan dari SVM adalah untuk mencari *hyperplane* dengan nilai ρ yang maksimal sehingga diperoleh *optimal hyperplane*. *Optimal hyperplane* dengan input dua dimensi ada dalam Gambar 2.16.



Gambar 2.16 *Optimal hyperplane*
Sumber: Haykin (2008)

Bila nilai optimal dari vektor bobot dinotasikan dengan w_o dan nilai optimal dari bias dinotasikan dengan b_o , maka *optimal hyperplane* dapat dinotasikan dalam Persamaan (2-22).

$$w_o^T x + b_o = 0 \dots\dots\dots (2-22)$$

Untuk memperoleh *optimal hyperplane* maka diperlukan parameter w_o dan b_o yang sesuai. Parameter w_o dan b_o harus memenuhi batasan sesuai Persamaan (2-23).

$$\begin{aligned} w_o^T x_i + b_o &\geq 1 \text{ untuk } d_i = +1 \\ w_o^T x_i + b_o &\leq -1 \text{ untuk } d_i = -1 \dots\dots\dots (2-23) \end{aligned}$$

Persamaan (2-21) berlaku bila pola *linearly separable*. Parameter w_o dan b_o pada persamaan (2-23) dapat diskalakan tanpa mempengaruhi Persamaan (2-22).

Nilai margin optimal pemisah antara dua *class* dinotasikan dengan ρ sesuai dengan Persamaan (2-24).

$$\rho = \frac{2}{\|w_o\|} \dots\dots\dots (2-24)$$

Optimal hyperplane yang didefinisikan pada Persamaan (2-22) memiliki nilai yang unik, dimana vektor bobot optimum w_o mampu memberikan pemisah yang optimal antara *positive class* dan *negative class*. Kondisi optimal ini diperoleh dengan meminimalisir *Euclidean norm* dari vektor bobot w .

Metode *quadratic optimization* dapat digunakan untuk memperoleh *optimal hyperplane* (Haykin, 2008). Formulasi dari *quadratic optimization* dilakukan dengan empat utama berikut:

1. Pencarian *optimal hyperplane* dimulai dengan pernyataan permasalahan dalam ruang optimasi yang terbatas.
2. Pembentukan fungsi Lagrangian dari permasalahan.
3. Penentuan kondisi untuk optimasi.
4. Penyelesaian masalah dalam ruang perkalian Lagrange ganda.

Sampel pelatihan dinyatakan sebagai $\mathfrak{S} = \{x_i, d_i\}_{i=1}^N$ dengan menggunakan vektor bobot w dan bias b yang memenuhi batasan pada Persamaan (2-25), dan vektor bobot w meminimalisir fungsi *cost* sesuai Persamaan (2-26).

$$d_i(w^T x_i + b) \geq 1 \text{ untuk } i = 1, 2, \dots, N \dots\dots\dots (2-25)$$

$$\phi(w) = \frac{1}{2} w^T w \quad \dots \dots \dots (2-26)$$

Fungsi Langrangian dibentuk sesuai dengan Persamaan (2-27) dengan pengali langrange dinotasikan dengan α .

$$J(w, b, \alpha) = \frac{1}{2} w^T w - \sum_{i=1}^N \alpha_i [d_i (w^T x_i + b) - 1] \quad \dots \dots \dots (2-27)$$

Solusi untuk permasalahan optimasi ditentukan oleh *saddle point* dari fungsi Langrangian $J(w, b, \alpha)$. *Saddle point* dari Langrangian merupakan titik dimana akar-akarnya bernilai real, tetapi memiliki tanda yang berlawanan. *Saddle point* harus diminimalisir terhadap w dan b , serta harus dimaksimalkan terhadap α , sehingga diperoleh kondisi optimasi seperti:

$$\text{Kondisi 1: } \frac{\partial J(w, b, \alpha)}{\partial w} = 0$$

$$\text{Kondisi 2: } \frac{\partial J(w, b, \alpha)}{\partial b} = 0$$

Aplikasi dari optimasi dengan menggunakan kondisi 1 terhadap fungsi *Langrangian* Persamaan (2-27) diperoleh hasil seperti Persamaan (2-28). Sedangkan aplikasi dari optimasi dengan menggunakan kondisi 2 terhadap fungsi Langrangian Persamaan (2-27) diperoleh hasil seperti Persamaan (2-29). Vektor solusi w didefinisikan sebagai ekspansi yang melibatkan contoh pelatihan sejumlah M . Meskipun solusi ini unik karena dalam bentuk *convex* Langrange, hal yang sama tidak berlaku pada pengali Langrange α_i .

$$w = \sum_{i=1}^w \alpha_i d_i x_i \quad \dots \dots \dots (2-28)$$

$$\sum_{i=1}^N \alpha_i d_i = 0 \quad \dots \dots \dots (2-29)$$

Sebagai catatan penting, semua batasan yang tidak sesuai dengan persamaan harus memiliki pengali α_i yang bernilai nol. Dengan kata lain, pengali tersebut harus sesuai dengan syarat pada Persamaan (2-30).

$$\alpha_i [d_i (w^T x_i + b) - 1] = 0 \quad \dots \dots \dots (2-30)$$

Permasalahan primal menggunakan fungsi *convex cost* dan batasan linier. Dengan permasalahan optimasi tersebut, permasalahan ganda dapat dibentuk. Permasalahan ganda ini memiliki nilai optimal yang sama dengan permasalahan primal, namun dengan pengali Langrange dapat diperoleh solusi optimal. Untuk menjabarkan permasalahan ganda, maka Persamaan (2-27) dikembangkan hingga diperoleh Persamaan (2-31).

$$J(w, b, \alpha) = \frac{1}{2} w^T w - \sum_{i=1}^N \alpha_i d_i w^T x_i - b \sum_{i=1}^N \alpha_i d_i + \sum_{i=1}^N \alpha_i \dots (2-31)$$

Bentuk ketiga pada sisi kanan pada Persamaan (2-31) bernilai nol sesuai dengan syarat optimal pada Persamaan (2-29). Sehingga dari Persamaan (2-27) dapat diperoleh hasil seperti pada Persamaan (2-32).

$$w^T w = \sum_{i=1}^N \alpha_i d_i w^T x_i = \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j d_i d_j w_i^T w_j \dots (2.32)$$

Sesuai dengan fungsi $J(w, b, \alpha) = Q(\alpha)$, Persamaan (2-31) ditulis kembali seperti pada Persamaan (2-33), dimana α_i tidak memiliki nilai negatif.

$$Q(\alpha) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j d_i d_j w_i^T w_j \dots (2-33)$$

Vektor bobot optimal w_o dapat dihitung menggunakan Persamaan (2-34), dengan pengali optimal *Langrange* dinotasikan oleh $\alpha_{o,i}$ dan jumlah vektor *support* dinotasikan dengan N_s .

$$w_o = \sum_{i=1}^{N_s} \alpha_{o,i} d_i x_i \dots (2-34)$$

Bias optimal b_o dihitung dengan menggunakan w_o sehingga diperoleh Persamaan (2-35). Vektor *support* $x^{(s)}$ mewakili titik manapun (x_i, d_i) dari sampel pelatihan dimana pengali *Langrange* $\alpha_{o,i}$ tidak bernilai nol.

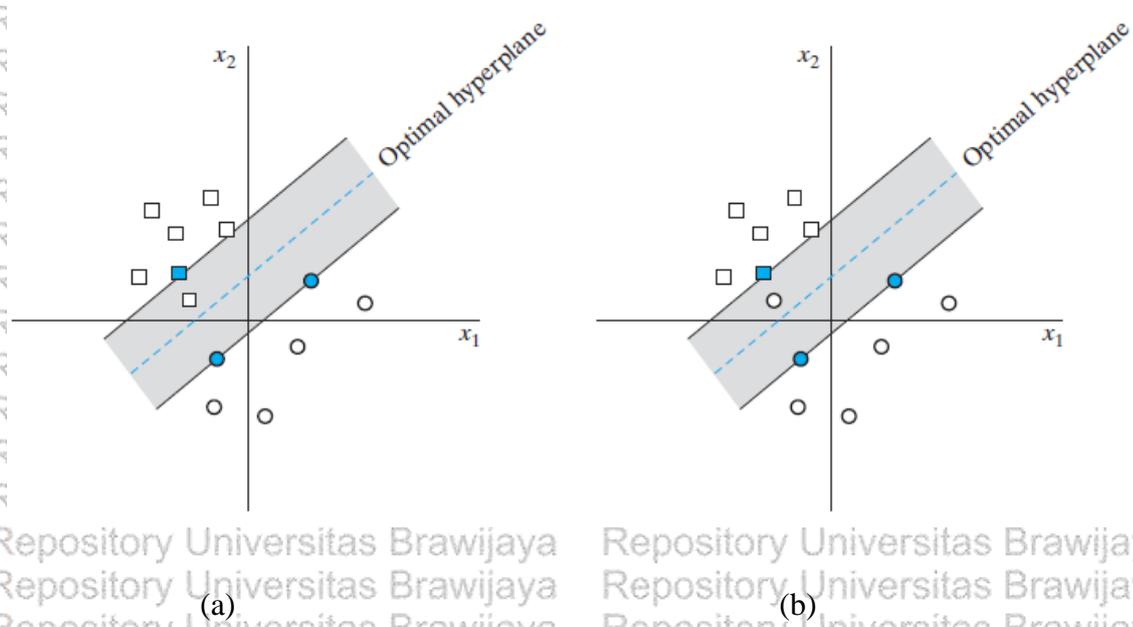
$$b_o = 1 - w_o^T x^{(s)} \quad \text{untuk } d^{(s)} = 1$$

$$b_o = 1 - \sum_{i=1}^{N_s} \alpha_{o,i} d_i x_i^T x^{(s)} \dots (2-35)$$

Pada kasus pola *nonseparable*, *hyperplane* tidak dapat dibentuk tanpa adanya *classification error*. *Optimal hyperplane* pada kasus *pola nonseparable* merupakan *hyperplane* dengan probabilitas *classification error* paling kecil diantara rata-rata sampel pelatihan.

Margin pemisah antar *class* disebut lunak bila titik data (x_i, d_i) tidak sesuai dengan syarat berikut:

- Titik data (x_i, d_i) terletak pada wilayah pemisah, namun pada sisi *decision surface* yang benar seperti pada *Gambar 2.17(a)*
- Titik data (x_i, d_i) terletak pada sisi *decision surface* yang salah seperti pada *Gambar 2.17(b)*.



Gambar 2.17 Margin *hyperplane* lunak (a) Titik data x_i (milik *class* ℓ_1 , yang disimbolkan kotak kecil) terletak pada wilayah pemisah, namun pada sisi *decision surface* yang benar. (b) Titik data x_i (milik *class* ℓ_2 , disimbolkan lingkaran kecil) terletak pada sisi *decision surface* yang salah. Sumber: Haykin (2008)

Kasus pola *unseparable* dapat diselesaikan dengan memperkenalkan serangkaian variabel skalar nonnegatif baru $\{\xi_i\}_{i=1}^N$ ke dalam definisi *hyperplane* pemisah seperti pada Persamaan (2-36).

$$d_i(w^T x_i + b) \geq 1 - \xi_i, \quad i = 1, 2, \dots, N \quad (2-36)$$

ξ_i disebut sebagai variabel *slack*, variabel ini menghitung turunan titik data dari kondisi ideal pada pemisahan pola. Pada $0 < \xi_i \leq 1$, titik data terletak pada wilayah pemisah, namun berada pada sisi *decision surface* yang benar, seperti dalam Gambar 2.15(a). Pada $\xi_i > 1$, titik data terletak pada *hyperplane* pemisah yang salah, seperti pada Gambar 2.15(b). Vektor *support* merupakan titik data yang memenuhi syarat $\xi_i = 0$.

Meminimalisir *misclassification error* pada *hyperplane* pemisah dapat diperoleh dengan menggunakan Persamaan (2-37).

$$\phi(w, \xi) = \frac{1}{2} w^T w + C \sum_{i=1}^N \xi_i \quad (2-37)$$

Parameter C merupakan parameter regulasi yang berfungsi untuk mengatur keseimbangan antara kompleksitas dari mesin dan jumlah titik *nonseparable*. Parameter C dengan nilai yang besar menandakan kualitas sampel pelatihan yang sangat bagus.

Parameter C dengan nilai yang kecil menandakan sampel pelatihan penuh dengan *noise* dan kurang tegas. Parameter C ditentukan sendiri oleh *user* dan bisa ditentukan secara eksperimen dengan menggunakan *cross-validation* untuk memperoleh hasil optimal.

Permasalahan primal untuk kasus *nonseparable* dapat diselesaikan dengan menentukan sampel pelatihan $\{x_i, d_i\}_{i=1}^N$, dan mencari nilai optimal dari vektor bobot w dan bias b yang sesuai dengan batasan Persamaan (2-38) dan Persamaan (2-39). Selanjutnya, fungsi *cost* diminimalisir dengan menggunakan Persamaan (2-40).

$$d_i(w^T x_i + b) \geq 1 - \xi_i \text{ untuk } i = 1, 2, \dots, N \quad (2-38)$$

$$\xi_i \geq 0 \text{ untuk semua } i \quad (2-39)$$

$$\phi(w, \xi) = \frac{1}{2} w^T w + C \sum_{i=1}^N \xi_i \quad (2-40)$$

Dengan menggunakan pengali Langrange dapat diformulasikan permasalahan ganda untuk pola *nonseparable*. Dengan sampel pelatihan $\{(x_i, d_i)\}_{i=1}^N$, dicari pengali Langrange $\{\alpha_i\}_{i=1}^N$ yang memaksimalkan fungsi obyektif sesuai Persamaan (2-41) dengan batasan Persamaan (2-42) dan Persamaan (2-43).

$$Q(\alpha) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j d_i d_j x_i^T x_j \quad (2-41)$$

$$\sum_{i=1}^N \alpha_i d_i = 0 \quad (2-42)$$

$$0 \leq \alpha_i \leq C \text{ untuk } i = 1, 2, \dots, N \quad (2-43)$$

2.8 Confusion Matrix

Confusion matrix adalah penggambaran performansi dari suatu algoritma klasifikasi dengan menggunakan data berupa matriks. Matriks ini membandingkan klasifikasi yang diprediksi dengan klasifikasi aktual dalam bentuk informasi *false positive* (FP), *true positive* (TP), *false negative* (FN), dan *true negative* (TN) (Kohavi dan Provost, 1998). *Confusion matrix* memberikan gambaran akurasi klasifikasi aktual terhadap prediksi (Awad and Khannda, 2015). Sebuah *confusion matrix* untuk sistem klasifikasi dua kelas ditunjukkan oleh Tabel 2.3.

Tabel 2.3 *Confusion Matrix*

Confusion Matrix		Predicted	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

TP = Keluaran diidentifikasi dengan tepat sebagai positif

TN = Keluaran diidentifikasi dengan tepat sebagai negatif

FP = Keluaran diidentifikasi dengan tidak tepat sebagai positif

FN = Keluaran diidentifikasi dengan tidak tepat sebagai negatif

Sumber: Kohavi dan Provost (1998)

Sejumlah variabel yang menunjukkan performa suatu sistem klasifikasi dapat diperoleh dari perhitungan *confusion matrix*. Variabel tersebut adalah akurasi, sensitivitas, dan spesifisitas (Kohavi dan Provost, 1998).

Akurasi merupakan jumlah data yang diklasifikasi dengan tepat. Akurasi diperoleh dengan Persamaan (2-44).

$$\text{Akurasi} = \frac{TP+TN}{TP+TN+FN+FP} \dots\dots\dots(2-44)$$

Sensitivitas merupakan ukuran sebagaimana baik suatu algoritma klasifikasi dalam melakukan klasifikasi data pada *positive class*. Sensitivitas dapat diperoleh dengan menggunakan Persamaan (2-45).

$$\text{Sensitivitas} = \frac{TP}{TP+FN} \dots\dots\dots(2-45)$$

Spesifisitas merupakan ukuran sebagaimana baik suatu algoritma klasifikasi dalam melakukan klasifikasi data pada *negative class*. Spesifisitas dapat diperoleh dengan menggunakan Persamaan (2-46).

$$\text{Spesifisitas} = \frac{TN}{TN+FP} \dots\dots\dots(2-46)$$

2.9 Receiver Operating Characteristic (ROC)

Grafik ROC merupakan penggambaran performansi suatu model klasifikasi dalam bentuk grafik (Kiranmai & Damodaram, 2014). Grafik ROC terbentuk dari penggambaran

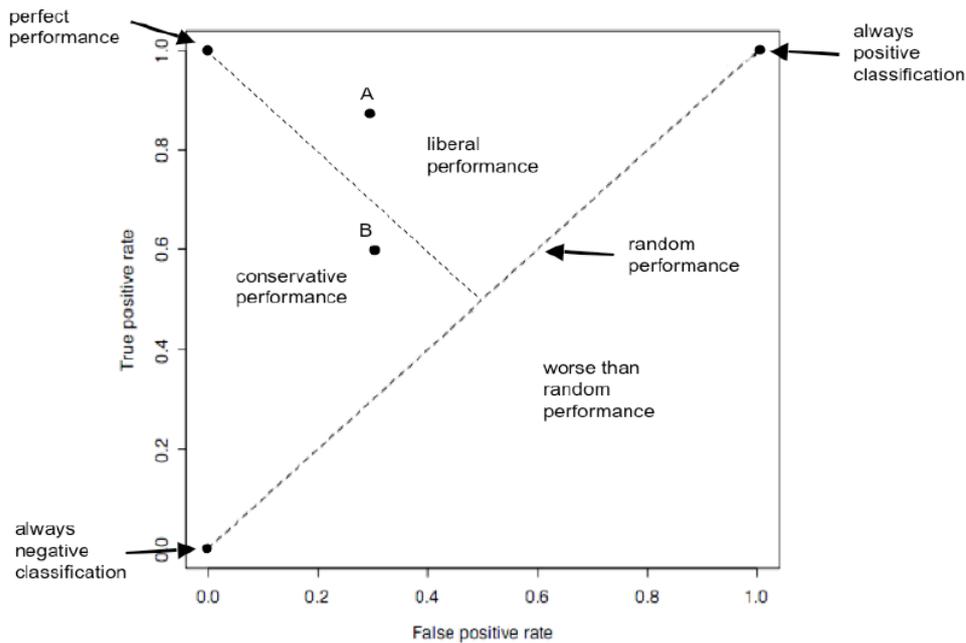
true positive rate (TPR) terhadap *false positive rate* (FPR). FPR dan TPR diperoleh dari nilai yang ada pada *confusion matrix* dengan Persamaan (2-47) dan Persamaan (2-48).

$$\text{True Positive Rate} = \frac{TP}{TP+FN} \dots\dots\dots (2-47)$$

$$\text{False Positive Rate} = \frac{FP}{FP+TN} \dots\dots\dots (2-48)$$

Grafik ROC dapat dibagi menjadi beberapa wilayah. Garis diagonal yang melintang merupakan garis yang menotasikan *random classifier*. *Random classifier* menunjukkan sistem klasifikasi yang menghasilkan respon *false positive* sebanyak respon *true positive*.

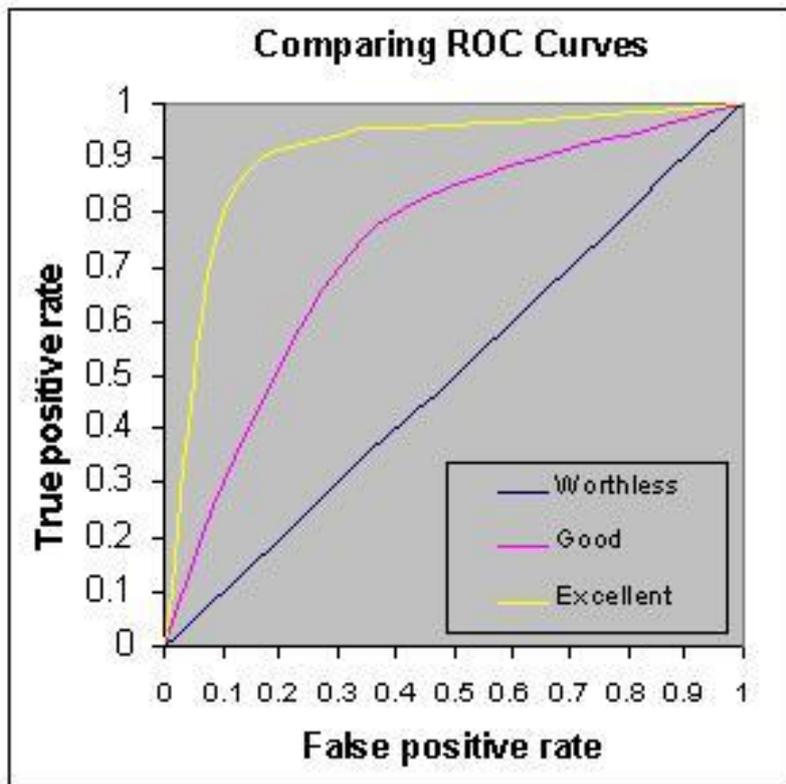
Pada bagian kiri bawah dari garis *random classifier* terdapat wilayah *conservative classification*. Wilayah *conservative classification* menunjukkan sistem klasifikasi yang memiliki respon *false positive error* yang banyak. Pada bagian kanan atas dari garis *random classifier* terdapat wilayah *liberal performance*, pada wilayah ini memiliki respon *true positive rate* yang banyak. Wilayah kurva pada ROC ada dalam *Gambar 2.18*.



Gambar 2.18 Wilayah dalam kurva ROC

Sumber: Bhardwaj (2011)

Jenis-jenis performansi dalam bentuk grafik ROC ada dalam *Gambar 2.19*.



Gambar 2.19 Contoh performansi sistem klasifikasi dalam ROC

Sumber: Tape (1999)

Akurasi pada ROC diukur menggunakan *area under ROC* (Tape, 1999). *Area under ROC* yang bernilai 1 memiliki artian bahwa sistem memiliki performansi sempurna, sedangkan nilai 0,5 memiliki artian bahwa sistem memiliki performansi yang sangat buruk (gagal). Tabel 2.4 menunjukkan panduan untuk mengklasifikasikan keakuratan pada ROC.

Tabel 2.4 Klasifikasi nilai *area under ROC*

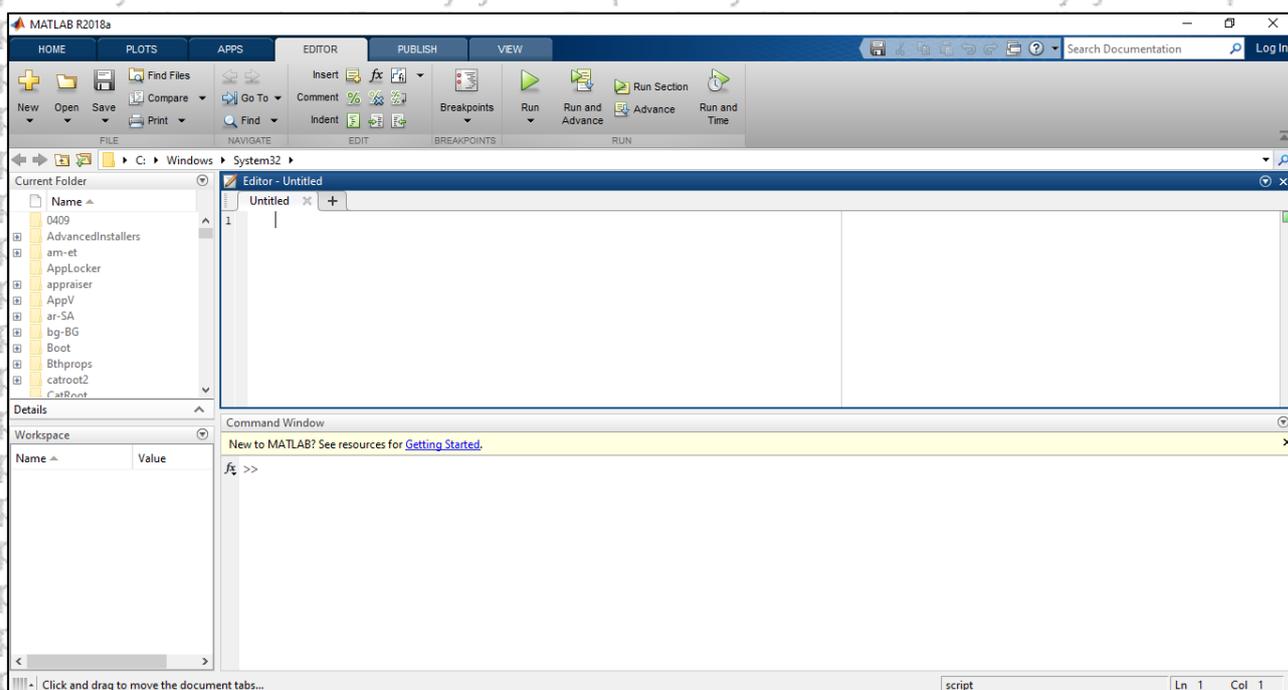
0.9 s/d 1	sangat baik
0.8 s/d 0.9	baik
0.7 s/d 0.8	sedang
0.6 s/d 0.7	buruk
0.5 s/d 0.6	sangat buruk

Sumber: Tape (1999)

2.10 Matlab 2018a

Matlab 2018a adalah platform pemrograman yang dirancang khusus untuk insinyur dan ilmuwan (mathworks.com, 2019). Inti dari Matlab adalah bahasa Matlab yang

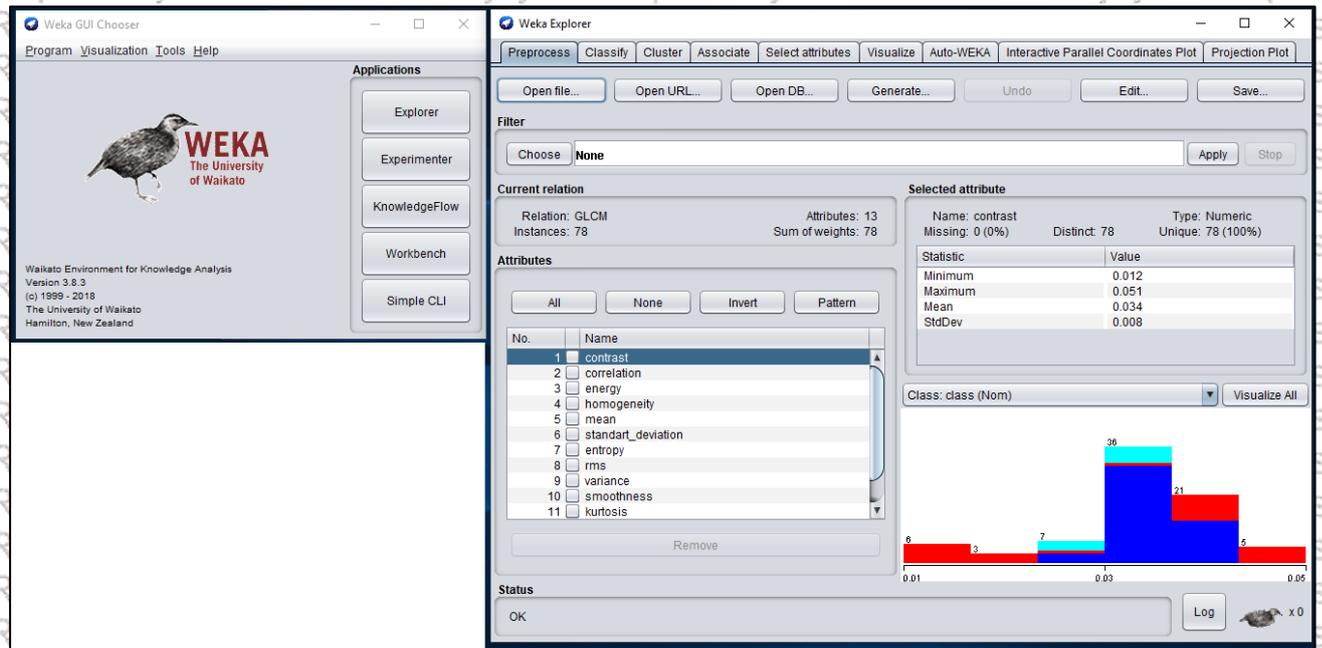
merupakan bahasa berbasis matriks yang memungkinkan ekspresi matematika komputasi yang paling alami. Matlab mampu melakukan analisa data, mengembangkan algoritma, dan membuat model serta aplikasi. Matlab dapat dipalikasiikan pada *deep learning* dan *machine learning*, pengolahan sinyal dan komunikasi, pengolahan citra dan video, sistem control, sistem uji dan pengukuran, perhitungan finansial, dan komputasi biologi. Tampilan aplikasi Matlab 2018a ada dalam *Gambar 2.20*.



Gambar 2.20 Tampilan aplikasi Matlab2018a

2.11 Weka 3.8

Weka 3.8 merupakan perangkat lunak yang berisi sekumpulan algoritma mesin pembelajaran (*machine learning*) untuk keperluan *data mining* (waikato.ac.nz, 2019). Weka 3.8 merupakan perangkat lunak *open source* yang dilengkapi dengan *tool* untuk berbagai keperluan pengolahan data. Weka 3.8 pada penelitian ini digunakan untuk proses klasifikasi dengan menggunakan metode SVM dan optimasi fitur dengan menggunakan metode PCA. Tampilan aplikasi Weka 3.8 ada dalam *Gambar 2.21*.



Gambar 2.21 Tampilan aplikasi Weka 3.8

BAB III

KERANGKA KONSEP PENELITIAN

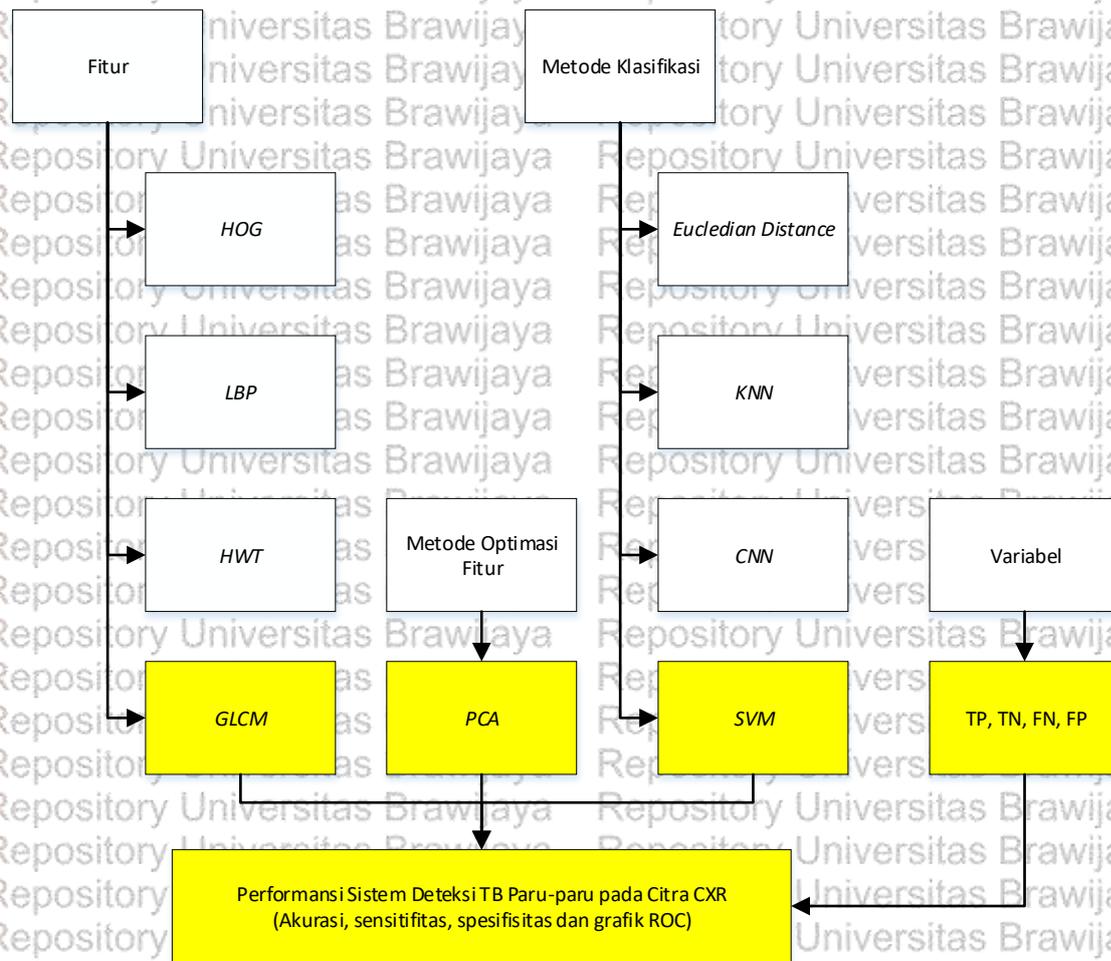
3.1 Kerangka Konsep Penelitian

Sistem deteksi TB paru-paru pada citra CXR performansinya ditunjukkan dalam bentuk nilai akurasi, sensitifitas, spesifisitas, dan grafik ROC. Fitur yang diekstraksi dari citra CXR dapat berupa HOG, GLCM, HWT, dan LBP. Fitur GLCM yang dipakai pada penelitian ini, karena GLCM mampu merepresentasikan kondisi normal dan abnormal paru-paru pada citra CXR. Agar hasil ekstraksi fitur lebih optimal maka fitur GLCM dioptimasi dengan PCA. PCA dipilih karena metode ini mampu mengoptimasi fitur GLCM dengan mereduksi fitur tekstur GLCM.

Metode klasifikasi yang dapat digunakan untuk klasifikasi citra normal dan abnormal beberapa diantaranya adalah *Euclidian Distance*, SVM, KNN, dan CNN. Metode klasifikasi yang digunakan pada penelitian ialah SVM. SVM banyak diaplikasikan dalam bidang biomedis dan memiliki keluaran dengan akurasi yang cukup bagus.

Variabel yang diteliti pada penelitian ini adalah TP, FP, FN dan TN. Variabel ini merupakan representasi dari performansi baik atau buruknya suatu sistem klasifikasi. Variabel TP, FP, TN, dan FN digunakan untuk perhitungan parameter performansi sistem klasifikasi seperti, akurasi, sensitivitas, spesifisitas serta pembentukan grafik ROC seperti dalam formulasi yang telah dijabarkan dalam bab tinjauan pustaka.

Kerangka konsep pada penelitian ada dalam *Gambar 3.1*.



: Ruang lingkup penelitian

Gambar 3.1 Kerangka konsep penelitian

3.2 Variabel Penelitian

Confusion matrix merupakan variabel yang diteliti. Variabel *confusion matrix* yang diteliti meliputi TP, TN, FN dan FP. Variabel *confusion matrix* diperoleh untuk masing-masing kelas klasifikasi (normal, PTB dan STB) baik untuk sistem dengan masukan GLCM yang tidak dioptimasi maupun sistem dengan masukan GLCM yang telah dioptimasi. Ukuran *confusion matrix* pada penelitian ini adalah 3x3.

Parameter performansi yang dihitung dari *confusion matrix* pada penelitian ini adalah akurasi, sensitivitas, spesifisitas dan grafik ROC. Parameter performansi pada sistem deteksi TB menunjukkan performansi sistem untuk masing-masing kelas (normal, PTB dan STB). Akurasi, sensitivitas dan spesifisitas ditunjukkan dalam bentuk prosentase. Grafik ROC

pada penelitian digunakan untuk menggambarkan performansi sistem deteksi TB pada citra CXR dengan lebih detail.

3.3 Alat Penelitian

Alat penelitian yang digunakan ialah laptop ASUS dengan spesifikasi Processor Intel(R) Core(TM) i7-4720HQ 2,60 GHz, 4 GB dengan OS Win 10 Pro dan aplikasi Matlab 2018a serta Weka 3.8. Aplikasi Matlab 2018a digunakan untuk melakukan proses *preprocessing*, segmentasi, dan ekstraksi fitur, sedangkan aplikasi Weka 3.8 digunakan untuk melakukan proses optimasi fitur menggunakan PCA dan klasifikasi SVM.

3.4 Analisis Masalah

TB merupakan penyakit mematikan yang disebabkan oleh bakteri MTB. Citra CXR merupakan alat utama untuk mendeteksi penyakit TB pada paru-paru. Citra CXR dianalisis oleh ahli radiologi untuk menentukan ada atau tidaknya penyakit TB pada paru-paru. Hasil analisis citra CXR oleh ahli radiologi dipengaruhi oleh faktor subyektifitas dari ahli radiologi. Faktor subyektifitas ahli radiologi dapat diatasi dengan sistem deteksi TB pada citra CXR yang terkomputerisasi.

3.5 Konsep Solusi

Konsep solusi pada penelitian meliputi pengumpulan data serta perancangan sistem.

3.5.1 Pengumpulan Data

Data sekunder yang digunakan berupa citra CXR yang bersumber dari *Shenzhen chest X-ray set*, Guangdong Medical College, Shenzhen, Cina (Jaeger dkk., 2014). Citra CXR yang dipakai adalah citra abu-abu (*grayscale*) CXR orang dewasa dengan format Portable Network Graphic (PNG) dan memiliki ukuran kurang lebih 3000x3000 piksel. Citra CXR yang diolah sebanyak 78 citra yang terdiri atas 46 citra dengan label normal, 24 citra dengan label PTB, dan 8 citra dengan label STB.

3.5.2 Perancangan Sistem

Perancangan sistem pada penelitian ini meliputi penentuan proses *preprocessing*, *segmentasi*, ekstraksi fitur, optimasi fitur dan klasifikasi. Pada penelitian dipakai PCA untuk optimasi fitur dan SVM sebagai proses klasifikasi. Proses *preprocessing*, *segmentasi*,

dan ekstraksi fitur dilakukan dengan menggunakan aplikasi Matlab 2018a. Aplikasi Weka

3.8 digunakan untuk melakukan proses optimasi fitur dan klasifikasi.

3.6 Hipotesis

Sistem deteksi TB paru-paru pada citra CXR dengan ekstraksi ciri GLCM yang dioptimasi dengan metode PCA dan klasifikasi SVM dapat dibangun pada penelitian ini.

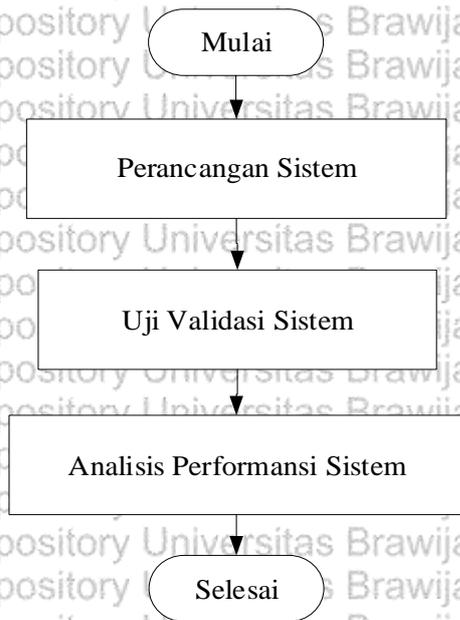
Performansi sistem deteksi TB paru-paru pada citra CXR dengan ekstraksi ciri GLCM yang dioptimasi dengan metode PCA dan klasifikasi SVM diharapkan memiliki nilai akurasi, sensitivitas dan spesifisitas yang cukup tinggi. Performansi sistem deteksi TB paru-paru pada citra CXR dengan metode optimasi PCA diharapkan memiliki performansi lebih baik bila dibandingkan dengan sistem deteksi TB paru-paru tanpa optimasi.



BAB IV

METODE PENELITIAN

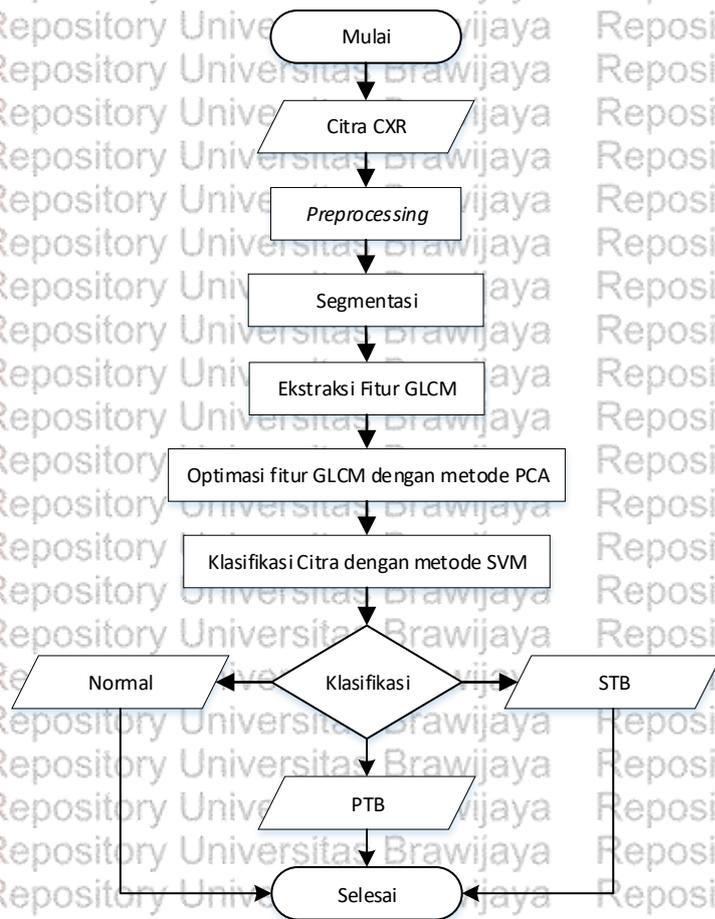
Bab IV menjelaskan perihal pengaplikasian metode. Pada bab ini akan dijabarkan perancangan sistem, uji validasi sistem, dan analisis performansi sistem. Metodologi pada penelitian ada dalam diagram alir *Gambar 4.1*.



Gambar 4.1 Metodologi penelitian

4.1 Perancangan Sistem

Langkah awal pada penelitian ini adalah perancangan sistem deteksi TB pada citra CXR. Citra CXR diolah melalui beberapa tahapan yaitu, *preprocessing*, segmentasi, ekstraksi fitur GLCM, optimasi fitur GLCM menggunakan metode PCA dan klasifikasi citra dengan menggunakan metode *multiclass* SVM. Citra CXR diklasifikasi ke dalam kelas normal, PTB atau STB. *Preprocessing*, segmentasi dan ekstraksi fitur dilakukan dengan menggunakan perangkat lunak Matlab 2018a sedangkan proses optimasi fitur dan klasifikasi dilakukan dengan menggunakan perangkat lunak Weka 3.8. Pada *Gambar 4.2* ditunjukkan blok diagram sistem yang digunakan pada penelitian ini.



Gambar 4.2 Blok diagram sistem

4.1.1 Preprocessing

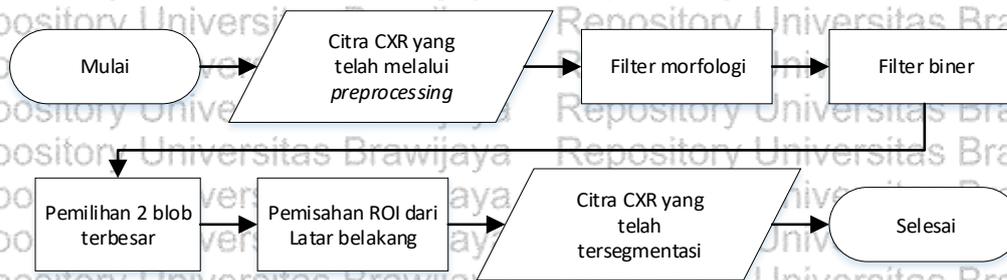
Citra CXR diperkecil ukurannya (*resize*) menjadi 512x512 untuk meringankan proses kalkulasi. Citra CXR dipertajam dengan menggunakan *adapt histogram*. Selanjutnya citra difilter dengan menggunakan *gaussian filter* untuk mengurangi *noise* pada citra. Gaussian filter dipilih karena mampu mengurangi *noise* pada citra secara signifikan (Russ, 2016). Diagram alur *preprocessing* ada dalam Gambar 4.3.



Gambar 4.3 Diagram alir preprocessing

4.1.2 Segmentasi

Segmentasi merupakan proses untuk memisahkan *region of interest* (ROI) citra CXR dari latar belakang. ROI pada citra CXR pada penelitian ini adalah citra paru-paru. Segmentasi dilakukan seperti pada *Gambar 4.4*. Proses segmentasi yang dilakukan meliputi penggunaan filter morfologi pada citra CXR hasil *preprocessing*. Dilanjutkan dengan penggunaan filter biner dan pemilihan dua *binary large object* (blob) terbesar dari hasil filter biner. Dua blob terbesar ini merupakan *mask* yang digunakan untuk memisahkan ROI dari latar belakang.

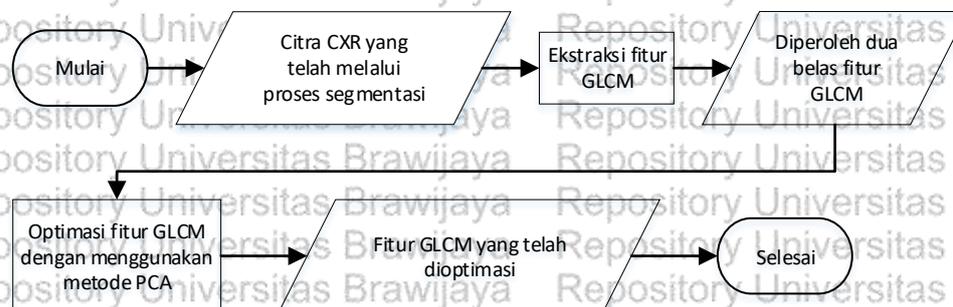


Gambar 4.4 Diagram alir segmentasi

4.1.3 Ekstraksi Fitur GLCM

Ekstraksi fitur merupakan proses pengambilan fitur GLCM dari citra yang telah melalui proses segmentasi. Setelah fitur GLCM diperoleh, maka fitur GLCM tersebut akan dioptimasi menggunakan dengan menggunakan metode PCA.

Diagram alir proses ekstraksi fitur ditunjukkan oleh *Gambar 4.5*.



Gambar 4.5 Diagram alir ekstraksi fitur

Dua belas fitur GLCM yang diekstraksi adalah:

1. Kontras

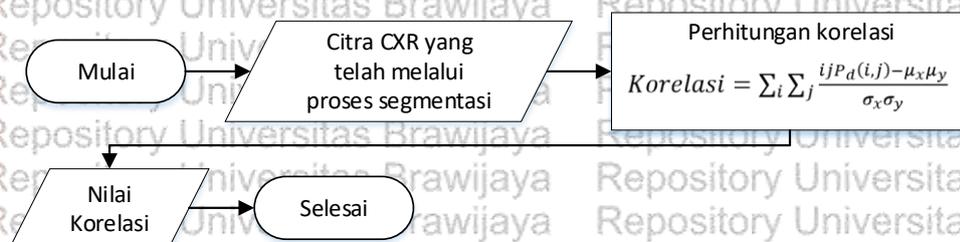
Kontras ialah variasi tingkat keabuan antar piksel referensi. Kontras akan bernilai nol untuk citra polos. Diagram alir ekstraksi kontras ditunjukkan dalam *Gambar 4.6* dimana matriks $P[i,j]$ merupakan matriks GLCM dari citra dengan nilai keabuan i dan j .



Gambar 4.6 Ekstraksi kontras

2. Korelasi

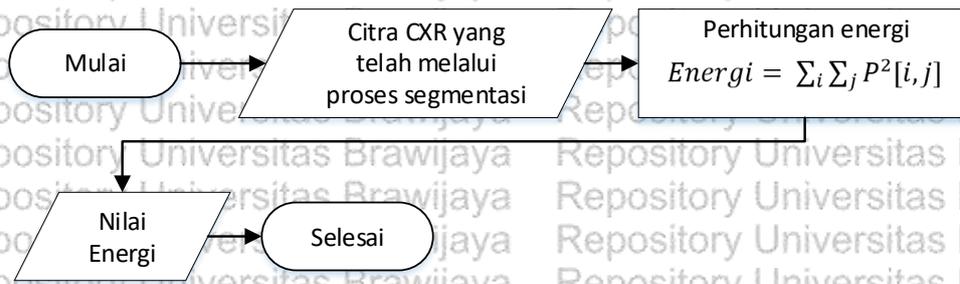
Korelasi merupakan ukuran hubungan piksel referensi terhadap tetangganya dalam sebuah citra. *Mean* dinotasikan sebagai μ_x, μ_y dan deviasi standar dinotasikan sebagai σ_x, σ_y . Diagram alir ekstraksi korelasi ditunjukkan dalam *Gambar 4.7*.



Gambar 4.7 Ekstraksi korelasi

3. Energi

Energi merupakan ukuran keseragaman (*uniformity*) lokal pada tingkat keabuan. Energi akan bernilai satu pada citra polos. Diagram alir ekstraksi energi ditunjukkan dalam *Gambar 4.8*.

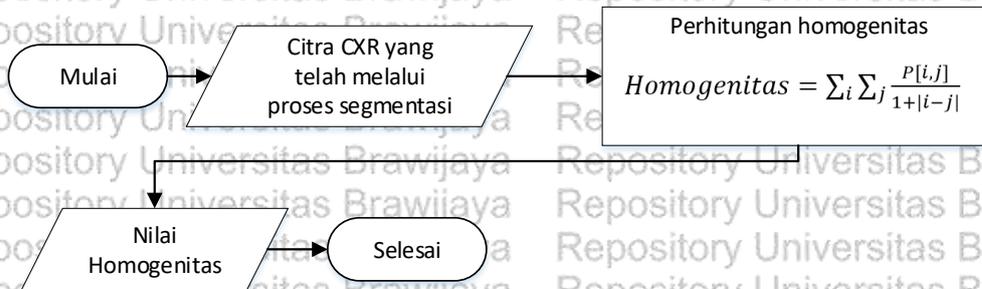


Gambar 4.8 Ekstraksi energi

4. Homogenitas

Homogenitas merupakan ukuran distribusi elemen pada GLCM terhadap diagonal GLCM. Diagram alir ekstraksi homogenitas ditunjukkan dalam

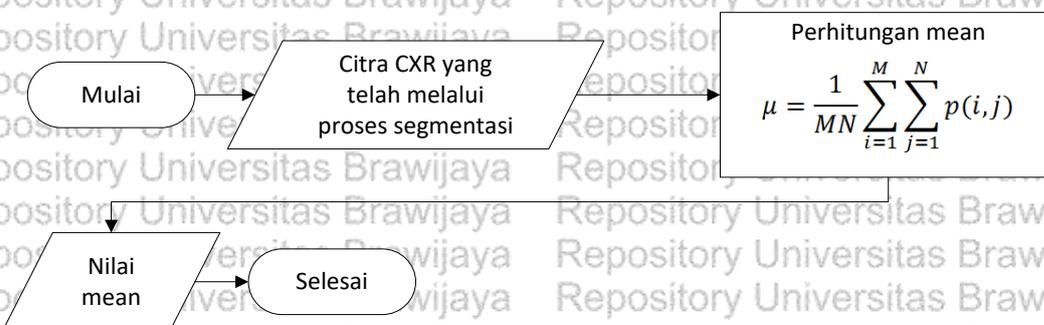
Gambar 4.9.



Gambar 4.9 Ekstraksi homogenitas

5. Mean

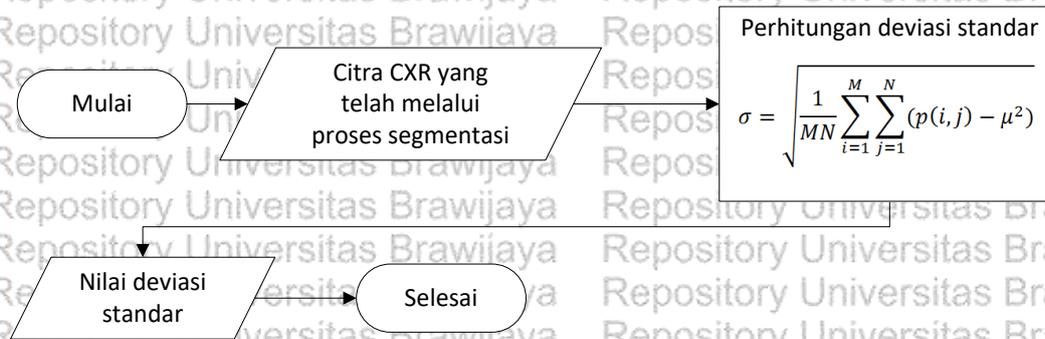
Mean (μ) merupakan nilai rata-rata piksel dalam sebuah citra. Diagram alir ekstraksi *mean* ditunjukkan dalam Gambar 4.10. $p(i, j)$ merupakan piksel titik (i, j) dari sebuah citra dengan ukuran $M \times N$.



Gambar 4.10 Ekstraksi mean

6. Deviasi standar

Deviasi standar (σ) merupakan estimasi deviasi kuadrat rata-rata piksel abu-abu dari nilai *mean*. Deviasi standar menggambarkan disperse di dalam wilayah lokal. Diagram alir ekstraksi deviasi standar ditunjukkan dalam *Gambar 4.11*.

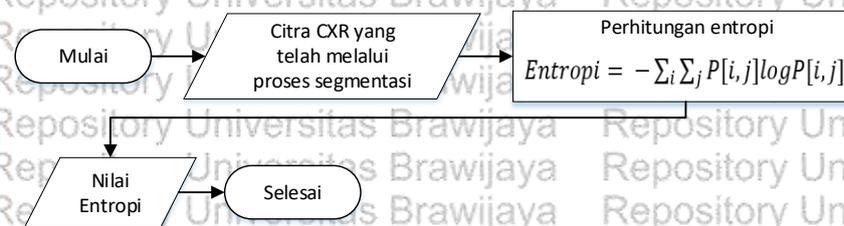


Gambar 4.11 Ekstraksi deviasi standar

7. Entropi

Entropi merupakan ukuran keacakan dari distribusi tingkat keabuan. Entropi digunakan sebagai ukuran distribusi variasi dalam suatu wilayah.

Diagram alir ekstraksi entropi ditunjukkan dalam *Gambar 4.12*.



Gambar 4.12 Ekstraksi entropi

8. Root Mean Square (RMS)

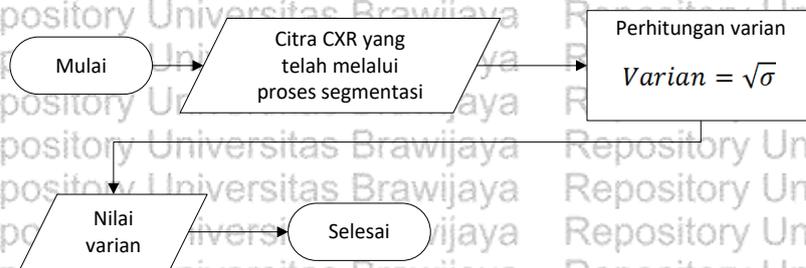
RMS merupakan perhitungan RMS dari tiap baris atau kolom masukan di sepanjang vektor dari dimensi input. Diagram alir ekstraksi RMS ditunjukkan dalam *Gambar 4.13*. I_{ij} merupakan intensitas piksel dari elemen ke- i dan ke- j pada citra dua dimensi dengan ukuran $M \times N$. Nilai \bar{I} merupakan intensitas rata-rata semua nilai piksel.



Gambar 4.13 Ekstraksi RMS

9. Varian

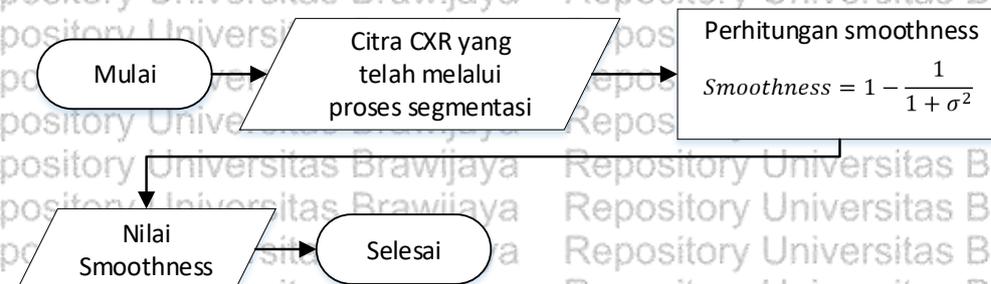
Varian merupakan akar kuadrat dari deviasi standar (σ). Diagram alir ekstraksi varian ditunjukkan dalam Gambar 4.14.



Gambar 4.14 Ekstraksi varian

10. Smoothness

Smoothness mendeskripsikan *relative smoothness* dari intensitas citra. *Smoothness* merupakan metode statistik untuk memilih blok citra yang memiliki *noise* lebih kecil. Diagram alir ekstraksi *smoothness* ditunjukkan dalam Gambar 4.15.



Gambar 4.15 Ekstraksi *smoothness*

11. Kurtosis

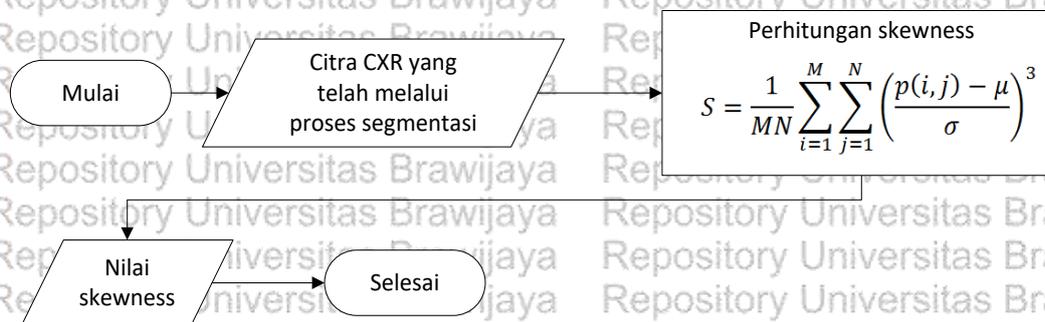
Kurtosis (K) merupakan ukuran keruncingan atau kedataran sebuah distribusi terhadap distribusi normal. Diagram alir ekstraksi *kurtosis* ditunjukkan dalam Gambar 4.16.



Gambar 4.16 Ekstraksi *kurtosis*

12. *Skewness*

Skewness (S) merupakan ukuran derajat distribusi asimetris suatu piksel disekitar *mean*. Diagram alir ekstraksi *skewness* ditunjukkan oleh Gambar 4.17.

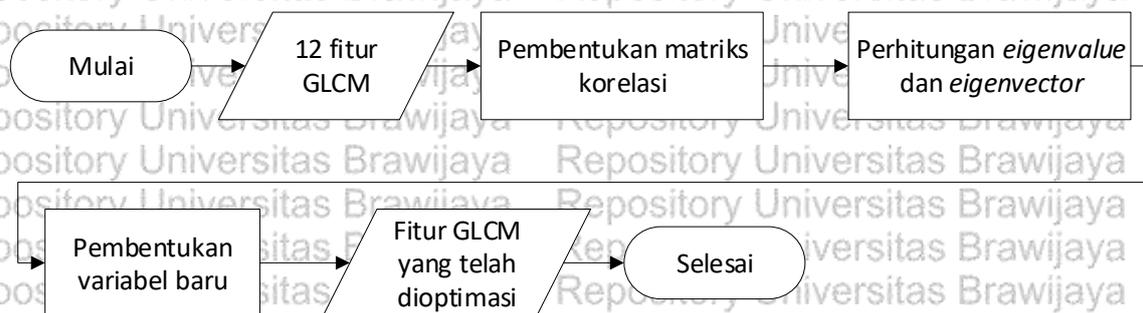


Gambar 4.17 Ekstraksi *skewness*

4.1.4 Optimasi Fitur GLCM

Dua belas fitur GLCM dari hasil proses ekstraksi fitur pada citra dioptimasi dengan menggunakan metode PCA. Metode PCA berfungsi untuk mengurangi dimensi fitur GLCM. Proses awal dari metode PCA adalah pembentukan matriks korelasi. Dua belas fitur GLCM diubah ke dalam bentuk matriks korelasi. Langkah selanjutnya adalah perhitungan *eigenvalue* dan *eigenvector* yang dilanjutkan dengan pembentukan fitur GLCM baru. Fitur GLCM baru tersebut digunakan sebagai

masuk ke dalam sistem klasifikasi SVM. Diagram alir proses optimasi fitur GLCM dengan metode PCA ditunjukkan pada Gambar 4.18.



Gambar 4.18 Optimasi fitur GLCM dengan metode PCA

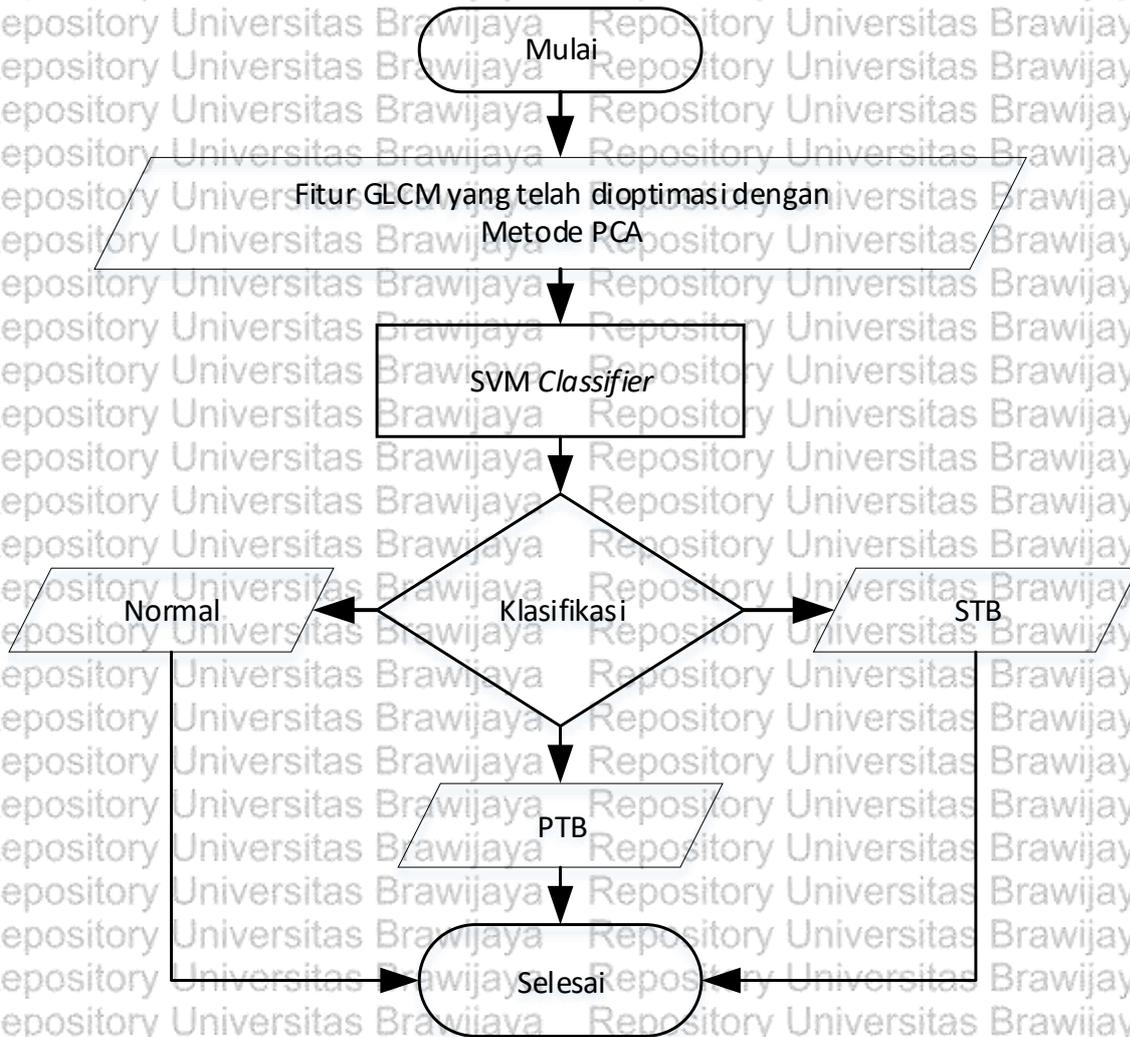
4.1.5 Klasifikasi

Proses klasifikasi merupakan proses penentuan citra CXR tergolong ke dalam kategori normal, PTB atau STB. Proses klasifikasi dilakukan dengan mengolah hasil ekstraksi fitur dengan menggunakan *multiclass SVM*. *Radial basis function* (RBF) merupakan tipe kernel SVM yang dipakai. Persamaan kernel RBF ditunjukkan dalam Persamaan 3-1, dimana $u-v$ merupakan jarak euclidean antar dua vektor fitur dan γ merupakan parameter yang menunjukkan seberapa jauh pengaruh suatu titik data latih.

$$\text{Kernel} = \exp(-\gamma|u - v|^2) \dots\dots\dots (3-1)$$

Kernel RBF dipilih pada penelitian ini karena kemampuannya untuk memetakan hampir semua fungsi nonlinier (Mueller dan Massaron, 2016). RBF bekerja dengan cara menciptakan margin disekitar setiap *support vector* dengan cara menggambarkan bentuk seperti gelembung. Margin dan *hyperplane* yang dihasilkan akan berbentuk sangat melengkung dan bersifat sangat fleksibel. Sifat fleksibilitas dari kernel RBF ini menyebabkan kernel mampu menemukan aturan klasifikasi yang rumit yang tidak mampu ditemukan oleh algoritma lain.

Fitur citra CXR yang telah dilengkapi dengan label normal, PTB dan STB digunakan sebagai data pelatihan. Proses klasifikasi ada dalam Gambar 4.19.



Gambar 4.19 Diagram alir proses klasifikasi

4.2 Uji Validasi Sistem

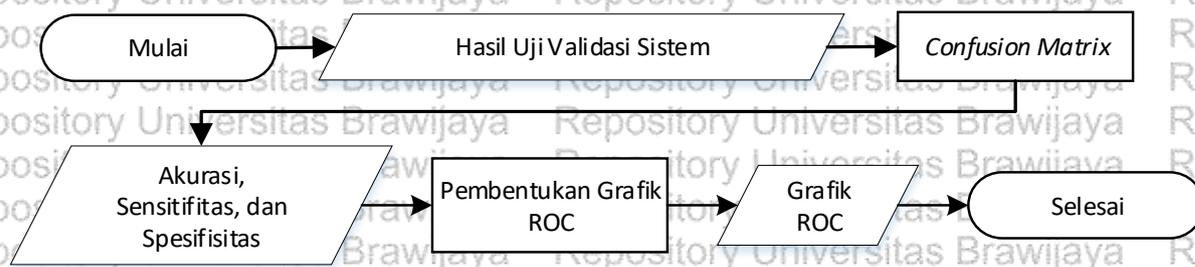
Sistem klasifikasi SVM memiliki dua masukan yaitu fitur GLCM tanpa optimasi dan fitur GLCM yang dioptimasi dengan menggunakan metode PCA. Sistem klasifikasi SVM pada penelitian ini divalidasi dengan menggunakan metode *k-fold cross-validation*. *Cross-validation* ialah metode statistik untuk mengevaluasi dan membandingkan algoritma dengan membagi data menjadi dua bagian (latih dan uji) (Refaeilzadeh dkk., 2009). *K-fold cross-validation* membagi data sebanyak k lipatan dengan merata. Proses pelatihan dan uji validasi data dilakukan k -kali secara berulang dengan setiap perulangan data yang diuji menggunakan lipatan data yang berbeda dengan syarat menyisakan lipatan sebanyak $k-1$ untuk data latih. Penelitian ini menggunakan *2-fold* hingga *10-fold cross-validation* sebagai uji validasinya.

4.3 Analisis Performansi Sistem

Penelitian ini membandingkan performansi sistem dengan masukan GLCM tanpa optimasi dengan performansi sistem dengan masukan GLCM yang telah dioptimasi.

Variabel yang dianalisis adalah *confusion matrix*. Parameter untuk mengukur performansi dari sistem yang diperoleh dari *confusion matrix* adalah akurasi, sensitivitas, spesifisitas dan grafik ROC. Grafik ROC dianalisis untuk mengetahui performansi sistem lebih detail.

Diagram alir analisis performansi sistem ditunjukkan dalam *Gambar 4.20*.



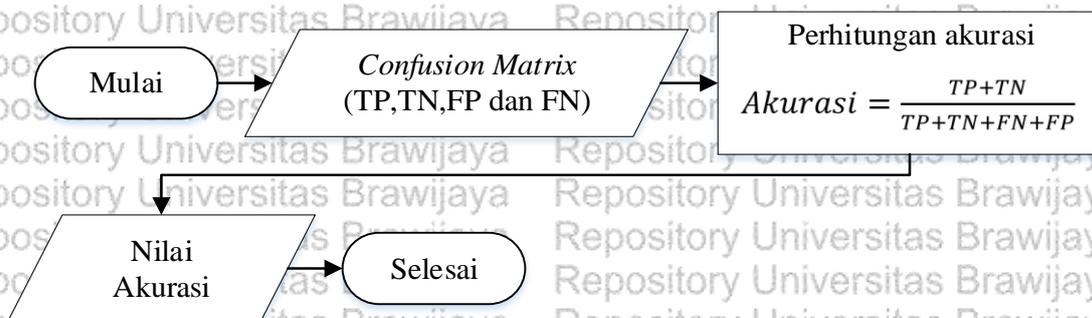
Gambar 4.20 Diagram alir analisis performansi sistem

4.3.1 *Confusion Matrix*

Confusion matrix diperoleh dari hasil uji validasi *2-fold* hingga *10-fold cross-validation*. Perhitungan akurasi, sensitivitas, spesifisitas dan grafik ROC dilakukan pada masing-masing kelas (normal, PTB dan STB) di setiap uji *cross-validation*.

4.3.1.1 Perhitungan Akurasi

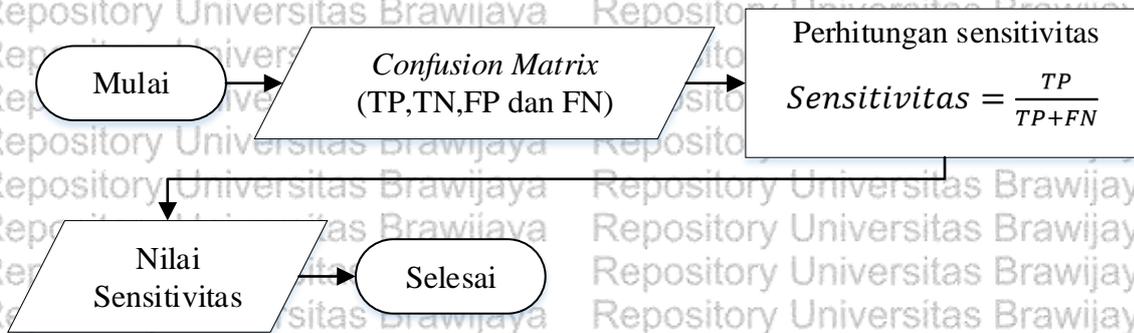
Akurasi adalah banyaknya data yang dapat diklasifikasi dengan tepat oleh sistem. *Gambar 4.21* merupakan diagram alir langkah perhitungan akurasi dari sistem yang didesain dalam penelitian ini.



Gambar 4.21 Diagram alir perhitungan akurasi

4.3.1.2 Perhitungan Sensitivitas

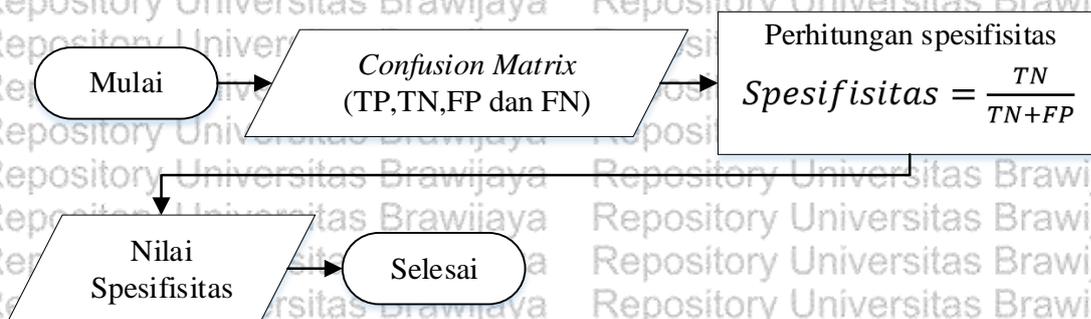
Sensitivitas adalah ukuran seberapa baiknya algoritma klasifikasi dalam melakukan klasifikasi data pada *positive class*. Gambar 4.22 merupakan diagram alir langkah perhitungan sensitivitas dari sistem yang didesain dalam penelitian ini.



Gambar 4.22 Diagram alir perhitungan sensitivitas

4.3.1.3 Perhitungan Spesifisitas

Spesifisitas adalah ukuran seberapa baiknya algoritma klasifikasi dalam melakukan klasifikasi data pada *negative class*. Gambar 4.23 merupakan diagram alir langkah perhitungan spesifisitas dari sistem yang didesain dalam penelitian ini.

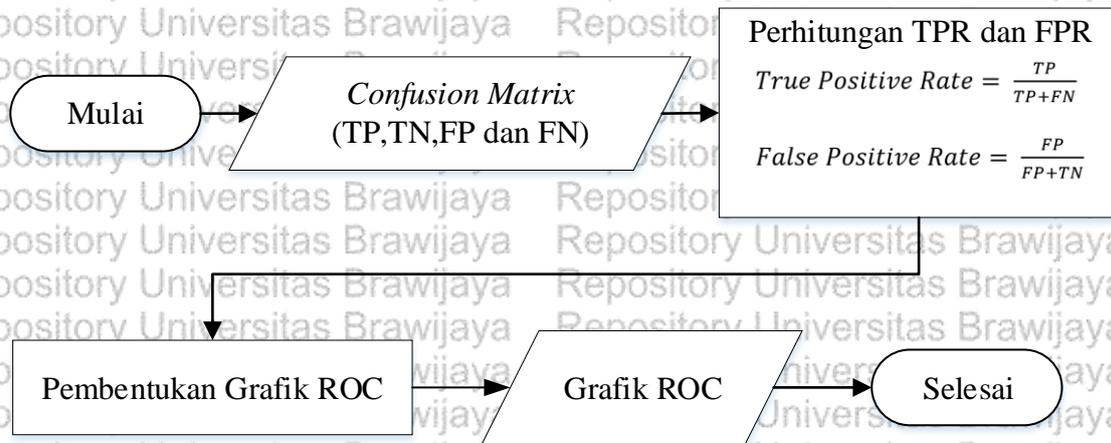


Gambar 4.23 Diagram alir perhitungan spesifisitas

4.3.1.4 Pembentukan Grafik ROC

Grafik ROC menggambarkan performa model klasifikasi dalam bentuk dua dimensi. Grafik ROC pada penelitian ini hanya menggambarkan performansi sistem dengan detail pada uji validasi dengan nilai akurasi

tertinggi. Grafik ROC terbentuk dari TPR dan FPR. *Gambar 4.24* merupakan diagram alir langkah pembentukan grafik ROC.



Gambar 4.24 Diagram alir pembentukan grafik ROC

BAB V

HASIL DAN PEMBAHASAN

Bab V menjelaskan prosedur pengolahan citra CXR, hasil penelitian serta pembahasan deteksi TB pada citra CXR dengan metode ekstraksi fitur GLCM yang dioptimasi dengan menggunakan metode PCA dan metode SVM untuk klasifikasinya. Pelulisan bab V meliputi prosedur pengolahan citra, hasil penelitian dan pembahasan.

5.1 Perancangan Sistem

Sistem deteksi TB pada citra CXR terdiri dari *preprocessing*, segmentasi, ekstraksi fitur, optimasi fitur serta klasifikasi. Komputasi berlangsung selama kurang lebih 30 menit.

5.1.1 *Preprocessing*

Preprocessing meliputi proses pengurangan ukuran citra (*resize*), penajaman fitur pada citra dan pengurangan *noise*. Proses *preprocessing* dilakukan dengan menggunakan bantuan perangkat lunak Matlab 2018a.

Tahap awal dari *preprocessing* adalah pengurangan dan penyamaan ukuran citra (*resizing*). Citra CXR dikurangi ukurannya menjadi 512x512 piksel untuk mempercepat proses pengolahan citra CXR. Setelah citra CXR dikurangi ukurannya, langkah selanjutnya adalah penajaman fitur pada citra CXR.

Fitur pada CXR dipertajam adalah fitur kontras. Fitur kontras pada citra CXR dipertajam dengan menggunakan menggunakan metode CLAHE (*Contrast Limited Adaptive Histogram Equalization*). CLAHE merupakan metode yang mampu meningkatkan kualitas kontras pada suatu citra dengan mempertimbangkan keberadaan *noise*. CLAHE diformulasikan dalam bentuk fungsi *adapthisteq* pada Matlab 2018a.

Langkah setelah penajaman fitur adalah pengurangan *noise* pada citra CXR. *Noise* pada citra CXR dikurangi dengan menggunakan filter gaussian. Filter gaussian mampu mengurangi *noise* dan menghaluskan (*smoothing*) citra CXR untuk mempermudah deteksi tepi (*edge detection*) pada proses segmentasi. *Gambar 5.1* menunjukkan *pseudo code* proses *preprocessing*. *Gambar 5.2* merupakan hasil dari tahapan *preprocessing* pada citra CXR.

```

Algoritma: Preprocessing
Masukan: Citra CXR (CXR)
Keluaran: Citra CXR preprocess (CXR_p)
//resize
1: CXR_r ← ubah ukuran menjadi 512x512 pada citra (CXR)
//mempertajam kontras
2: CXR_c ← CLAHE dengan cliplimit 0,003 dan distribusi Rayleigh pada citra (CXR_r)
//mengurangi noise
3: CXR_p ← mengaplikasikan filter Gaussian dengan sigma sebesar 1 pada citra (CXR_c)

```

Gambar 5.1 Pseudo code preprocessing



(a)

(b)

(c)

Gambar 5.2 Hasil dari tahapan *preprocessing* citra CXR; (a) hasil *resize*, (b) hasil penajaman fitur (CLAHE), (c) hasil pengurangan noise (filter gaussian).

5.1.2 Segmentasi

Proses segmentasi merupakan proses pemisahan area *region of interest* (ROI) dari citra latar belakang. Proses segmentasi memisahkan bidang paru-paru pada citra CXR. Proses segmentasi pada penelitian ini menggunakan filter morfologi, metode segmentasi biner dan pemilihan dua *blob* (*binary large object*) terbesar untuk memperoleh *mask*. *Mask* digunakan untuk memisahkan citra latar belakang dengan area ROI pada citra CXR. *Pseudo code* untuk proses segmentasi ditunjukkan dalam Gambar 5.3. Hasil dari tahapan proses segmentasi ditunjukkan dalam Gambar 5.4 dan Gambar 5.5.

```

Algoritma: Segmentasi
Masukan: Citra CXR preprocess (CXR_p)
Keluaran: Citra CXR segmentasi (CXR_s)
//filter morfologi
1: Strel ← membentuk filter berbentuk berlian dengan jari-jari 20 piksel
2: Ie ← mengikis (erode) citra (CXR_p) menggunakan filter (Strel)
3: Ie ← mengikis (erode) citra (Ie) menggunakan filter (Strel)
4: Iobr ← merekonstruksi citra (Ie) berdasar citra (CXR_p)
5: Iobrd ← melebarkan citra (Iobr) dengan menggunakan filter (Strel)
6: Iobrcbr ← merekonstruksi citra (Iobrd) negatif berdasar citra (Iobr) negatif
7: Iobrcbr ← citra Iobrcbr negatif
8: se ← membentuk filter garis dengan ketebalan 10 piksel dengan arah 0 dan 90 derajat
9: Iobrcbr ← melebarkan citra (Iobrcbr) dengan menggunakan filter garis (se)
//segmentasi biner
1: bw ← membentuk citra biner dari citra (Iobrcbr) dengan menggunakan metode Otsu
2: bw1 ← mengisi/menambal lubang pada citra (bw)
//pemilihan dua blob terbesar
1: Labelblob ← melakukan pelabelan pada semua blob pada citra (bw1)
2: Area ← menghitung luasan masing-masing blob yang telah diberi label (Labelblob)
3: bw2 ← mengambil dua blob terbesar (Labelblob, Area)
//Merapikan bentuk mask
1: D ← mendeteksi tepi citra (bw2) dengan metode sobel
2: se2 ← membentuk filter garis dengan ketebalan 10 piksel dengan arah 0 dan 90 derajat
3: E ← melebarkan tepi citra (D) dengan menggunakan filter (se2)
4: F ← mengisi lubang pada citra (E)
5: seds ← membentuk filter berbentuk berlian dengan jari-jari 3 piksel
6: G ← mengikis citra (F) menggunakan filter (seds)
7: G2 ← mengikis citra (G) menggunakan filter (seds)
8: se3 ← membentuk filter garis dengan ketebalan 7 piksel dengan arah 0 dan 90 derajat
9: bw3 ← melebarkan citra (G2) dengan menggunakan filter (se3)
//Menggabungkan citra hasil preprocessing dengan mask
1: CXR_s ← menggabungkan citra hasil preprocessing (CXR_p) dengan citra (bw3)

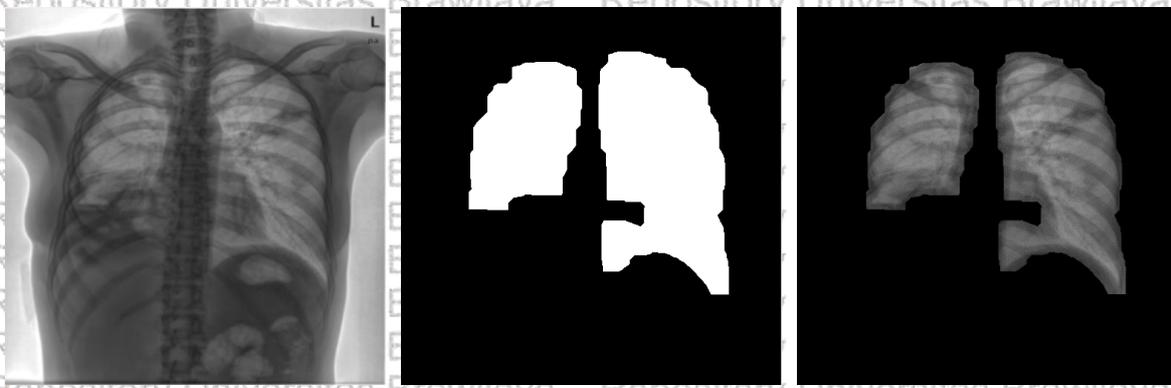
```

Gambar 5.3 Pseudo code segmentasi



(a) (b) (c)

Gambar 5.4 Hasil dari tahapan segmentasi citra CXR; (a) hasil *preprocessing* citra CXR, (b) *mask*, (c) hasil dari proses segmentasi.



Gambar 5.5 Hasil dari tahapan segmentasi citra CXR yang tidak sempurna; (a) hasil *preprocessing* citra CXR, (b) *mask*, (c) hasil dari proses segmentasi.

Gambar 5.4 menunjukkan proses segmentasi yang sempurna, dimana sistem mampu memisahkan bidang paru-paru dari gambar latar belakang dengan sempurna.

Gambar 5.5 menunjukkan proses segmentasi yang tidak sempurna, dimana bidang paru-paru pada citra CXR terpotong sebagian. Ketidak sempurnaan segmentasi pada citra CXR ini dikarenakan adanya warna putih yang pekat pada bidang paru-paru.

Warna putih pada bidang paru-paru ini diakibatkan karena adanya benda asing pada paru-paru, sehingga sistem tidak mampu mengenali bagian tersebut sebagai bagian dari paru-paru. Dari 78 citra CXR yang disegmentasi, terdapat 10 citra yang tidak dapat disegmentasi secara sempurna. 10 citra CXR tersebut terdiri atas 9 citra berlabel PTB dan 1 citra berlabel STB.

5.1.3 Ekstraksi Fitur GLCM

GLCM merupakan fitur yang diekstraksi. Fitur GLCM pada penelitian ini sejumlah dua belas fitur, yaitu kontras, korelasi, energi, homogenitas, mean, deviasi

standar, entropi, RMS, varian, *smoothness*, *kurtosis*, dan *skewness*. *Pseudo code* proses ekstraksi fitur ditunjukkan dalam *Gambar 5.6* dan *Gambar 5.7*. Hasil dari proses ekstraksi fitur ditunjukkan dalam *Gambar 5.8*.

Algoritma: Ekstraksi fitur GLCM

Masukan: Citra CXR hasil segmentasi (CXR_s)

Keluaran: Kontras, Korelasi, Energi, Homogenitas, Mean, Deviasi_Standar, Entropi, RMS, Varian, Smoothness, Kurtosis, Skewness

//Kontras

1: $P[i, j] \leftarrow$ membentuk matrix GLCM dari citra (CXR_s)

2: Kontras \leftarrow menghitung kontras dengan menggunakan formula:

$$\text{Kontras} = \sum_i \sum_j (i - j)^2 = P[i, j]$$

//Korelasi

1: $\mu_x, \mu_y \leftarrow$ menghitung mean dari matrix GLCM ($P[i, j]$)

2: $\sigma_x, \sigma_y \leftarrow$ menghitung deviasi standar dari matrix GLCM ($P[i, j]$)

3: Korelasi \leftarrow menghitung korelasi dengan menggunakan formula:

$$\text{Korelasi} = \sum_i \sum_j \frac{ijP_d(i, j) - \mu_x\mu_y}{\sigma_x\sigma_y}$$

//Energi

1: Energi \leftarrow menghitung energy dengan menggunakan formula:

$$\text{Energi} = \sum_i \sum_j P^2 [i, j]$$

//Homogenitas

1: Homogenitas \leftarrow menghitung homogenitas dengan menggunakan formula:

$$\text{Homogenitas} = \sum_i \sum_j \frac{P[i, j]}{1 + |i - j|}$$

//Mean

1: $M, N \leftarrow$ menghitung ukuran panjang (M) dan lebar (N) dari citra (CXR_s)

2: $p(i, j) \leftarrow$ menghitung nilai piksel pada citra (CXR_s)

3: Mean \leftarrow menghitung mean dari keseluruhan citra dengan formula:

$$\mu = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N p(i, j)$$

Gambar 5.6 Pseudo code ekstraksi fitur GLCM bagian 1 dari 2.

```
//Deviasi_Standar
1: Deviasi_Standar ← menghitung deviasi standar dari keseluruhan citra dengan formula:

$$\sigma = \sqrt{\frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (p(i,j) - \mu)^2}$$

//Entropi
1: Entropi ← menghitung entropi dengan formula:

$$Entropi = - \sum_i \sum_j P[i,j] \log P[i,j]$$

//RMS
1: Iij ← menghitung intensitas piksel tiap titik pada citra (CXR_s)
2:  $\bar{I}$  ← menghitung rata-rata intensitas piksel (Iij) secara keseluruhan
3: RMS ← menghitung RMS dengan formula:

$$RMS = \sqrt{\frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (I_{ij} - \bar{I})^2}$$

//Varian
1: Varian ← menghitung varian dengan menggunakan formula:

$$Varian = \sqrt{\sigma}$$

//Smoothness
1: Smoothness ← menghitung smoothness dengan menggunakan formula:

$$Smoothness = 1 - \frac{1}{1 + \sigma^2}$$

//Kurtosis
1: Kurtosis ← menghitung kurtosis dengan menggunakan formula:

$$Kurtosis = \left\{ \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N \left[ \frac{p(i,j) - \mu}{\sigma} \right]^4 \right\} - 3$$

//Skewness
1: Skewness ← menghitung skewness dengan formula:

$$Skewness = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N \left( \frac{p(i,j) - \mu}{\sigma} \right)^3$$

```

Gambar 5.7 Pseudo code ekstraksi fitur GLCM bagian 2 dari 2.

	1	2	3	4	5	6	7	8	9	10	11	12
	Kontras	Korelasi	Energi	Homogenitas	Mean	Dev_Standar	Entropi	RMS	Varian	Smoothness	Kurtosis	Skewness
1	0.0402	0.9907	0.4626	0.9838	37.1860	53.6253	3.1763	7.6964	2.0149e+03	1.0000	1.9854	0.8740
2	0.0360	0.9910	0.4861	0.9850	33.6736	51.6879	3.0648	7.5226	1.9970e+03	1.0000	2.4009	1.0475
3	0.0366	0.9913	0.4688	0.9848	36.3881	53.1367	3.1438	7.3450	1.8348e+03	1.0000	2.0378	0.9034
4	0.0389	0.9910	0.4558	0.9842	37.4986	53.5809	3.2297	7.3368	1.7361e+03	1.0000	1.9817	0.8638
5	0.0367	0.9909	0.4745	0.9844	35.2191	52.1491	3.1052	7.1643	1.7033e+03	1.0000	2.1219	0.9406
6	0.0321	0.9918	0.5065	0.9861	32.3158	51.3137	2.9273	7.1921	1.9400e+03	1.0000	2.4975	1.1078
7	0.0315	0.9922	0.4927	0.9871	34.1611	52.1236	3.0011	7.3324	1.9671e+03	1.0000	2.2356	0.9996
8	0.0397	0.9903	0.4937	0.9846	34.0289	52.3092	2.9819	6.8489	1.7240e+03	1.0000	2.2717	1.0177
9	0.0335	0.9913	0.5353	0.9871	30.6116	50.7299	2.7259	6.9259	1.8887e+03	1.0000	2.6481	1.1823
10	0.0389	0.9910	0.4609	0.9848	37.3522	53.5583	3.1808	8.0758	2.1888e+03	1.0000	1.9497	0.8580
11	0.0394	0.9903	0.4739	0.9843	35.3882	52.2327	3.0982	7.2613	1.8149e+03	1.0000	2.0920	0.9277
12	0.0344	0.9915	0.5178	0.9867	32.7969	52.0115	2.8106	7.0109	1.9324e+03	1.0000	2.3288	1.0622
13	0.0335	0.9916	0.4968	0.9873	33.7684	51.8291	2.9796	7.3429	1.9783e+03	1.0000	2.3023	1.0185
14	0.0290	0.9923	0.5350	0.9871	30.0133	50.2380	2.7447	7.0195	1.9561e+03	1.0000	2.7952	1.2278
15	0.0337	0.9915	0.4922	0.9858	33.5756	51.7274	3.0084	7.5325	2.0095e+03	1.0000	2.3383	1.0371

Gambar 5.8 Hasil ekstraksi fitur GLCM.

5.1.4 Optimasi Fitur GLCM

Fitur GLCM yang dioptimasi dengan menggunakan metode PCA dioptimasi dengan cara mengubah fitur GLCM yang ada menjadi sekumpulan fitur baru. Dua belas fitur GLCM sebelum dioptimasi dengan menggunakan metode PCA adalah kontras, korelasi, energy, homogenitas, mean, deviasi standar, entropi, RMS, varian, *smoothness*, *kurtosis*, dan *skewness*. Metode optimasi PCA menghasilkan dua fitur baru dari dua belas fitur GLCM yang ada sebelumnya. Dua fitur baru ditunjukkan dalam bentuk formulasi sebagai berikut:

$$\text{Fitur 1} = -0,303\text{energi} - 0,303\text{skewness} + 0,301\text{entropi} + 0,3\text{RMS} + 0,298\text{mean}$$

$$\text{Fitur 2} = -0,861\text{korelasi} - 0,373\text{varian} + 0,192\text{kontras} - 0,186\text{deviasi standar} \\ - 0,147\text{homogenitas}$$

5.1.5 Klasifikasi

Penelitian ini mengklasifikasikan citra CXR menjadi tiga kelas, yaitu normal, TB primer (PTB) dan TB sekunder (STB). Metode klasifikasi yang digunakan adalah *multiclass SVM*. Jenis *multiclass SVM* yang digunakan pada penelitian adalah C-SVC (*support vector class*). C-SVC merupakan tipe SVM yang memaksimalkan jarak antara titik sampel terhadap *hyperplane* untuk meminimalisir *misclassification error*. Fungsi pada C-SVC untuk meminimalisir *misclassification error* ditunjukkan dalam Persamaan (2-37). Parameter C yang digunakan pada penelitian ini bernilai 1.

5.2 Hasil Penelitian dan Pembahasan

Sub bab ini menjelaskan performansi sistem klasifikasi citra CXR dengan metode SVM dengan masukan fitur GLCM tanpa optimasi dan GLCM yang telah dioptimasi dengan menggunakan metode PCA. Variabel yang diamati pada penelitian ini adalah *confusion matrix*. Parameter performansi yang dapat diperoleh dari *confusion matrix* meliputi akurasi, sensitivitas, spesifisitas dan grafik ROC.

5.2.1 Analisis Akurasi, Sensitivitas dan Spesifisitas

Metode klasifikasi SVM pada penelitian ini memiliki dua jenis masukan yaitu GLCM tanpa optimasi dan GLCM yang telah dioptimasi dengan PCA. Hasil klasifikasi SVM ditunjukkan dalam *Gambar 5.12* untuk masukan GLCM tanpa optimasi, dan *Gambar 5.13* untuk masukan GLCM yang telah dioptimasi.

Sistem klasifikasi SVM divalidasi dengan *cross-validation* dari 2-fold hingga 10-fold. Hasil validasi dari sistem dengan masukan GLCM tanpa optimasi akan dibandingkan dengan sistem dengan masukan GLCM yang telah dioptimasi dengan menggunakan PCA. Hasil uji validasi 2-fold *cross-validation* ditunjukkan pada Gambar 5.9 (GLCM tanpa optimasi) dan Gambar 5.10 (GLCM yang telah dioptimasi dengan PCA).

```

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      41           52.5641 %
Incorrectly Classified Instances    37           47.4359 %
Kappa statistic                    0.1164
Mean absolute error                 0.3595
Root mean squared error             0.4183
Relative absolute error             97.0579 %
Root relative squared error         97.9093 %
Total Number of Instances          78

=== Detailed Accuracy By Class ===

                TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                0.609   0.469   0.651     0.609   0.629     0.138   0.647    0.767    normal
                0.542   0.407   0.371     0.542   0.441     0.125   0.626    0.449    PTB
                0.000   0.000   ?         0.000   ?         ?       0.589    0.150    STB
Weighted Avg.   0.526   0.402   ?         0.526   ?         ?       0.634    0.606

=== Confusion Matrix ===

  a  b  c  <-- classified as
28 18  0 | a = normal
11 13  0 | b = PTB
 4  4  0 | c = STB

```

Gambar 5.9 Hasil uji validasi 2-fold *cross-validation* pada SVM (GLCM tanpa optimasi).

```

=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances      74          94.8718 %
Incorrectly Classified Instances    4           5.1282 %
Kappa statistic                    0.9031
Mean absolute error                 0.0994
Root mean squared error             0.1593
Relative absolute error             26.8431 %
Root relative squared error         37.2745 %
Total Number of Instances          78

=== Detailed Accuracy By Class ===

                TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                1.000  0.063  0.958     1.000  0.979     0.948   1.000    1.000    normal
                1.000  0.037  0.923     1.000  0.960     0.943   0.999    0.998    PTB
                0.500  0.000  1.000     0.500  0.667     0.688   0.996    0.975    STB
Weighted Avg.   0.949  0.048  0.952     0.949  0.941     0.920   0.999    0.997

=== Confusion Matrix ===

 a  b  c  <-- classified as
46  0  0 | a = normal
 0 24  0 | b = PTB
 2  2  4 | c = STB

```

Gambar 5.10 Hasil uji validasi 2-fold cross-validation pada SVM (GLCM yang telah dioptimasi).

Confusion matrix dalam Gambar 5.12 menunjukkan bahwa sistem mampu mengklasifikasi citra CXR normal dengan tepat sebanyak 28 citra dari 46 citra, sedangkan citra CXR dengan label PTB mampu diklasifikasi dengan tepat sebanyak 13 citra dari 24 citra, dan citra CXR dengan label STB tidak mampu diklasifikasi sama sekali. Confusion matrix dalam Gambar 5.13 menunjukkan bahwa sistem mampu mengklasifikasi citra CXR normal dan PTB dengan tepat, sedangkan citra CXR dengan label STB hanya mampu diklasifikasi dengan tepat sebanyak 4 dari 8 citra.

Confusion matrix sistem klasifikasi SVM pada uji validasi 2-fold hingga 10-fold cross validation ditunjukkan Tabel 5.1.

Tabel 5.1 *Confusion matrix* sistem klasifikasi SVM pada uji validasi *2-fold* hingga *10-fold cross-validation*

GLCM tanpa optimasi				GLCM yang telah dioptimasi dengan PCA			
2-fold	Normal	PTB	STB	2-fold	Normal	PTB	STB
Normal	28	18	0	Normal	46	0	0
PTB	11	13	0	PTB	0	24	0
STB	4	4	0	STB	2	2	4
3-fold	Normal	PTB	STB	3-fold	Normal	PTB	STB
Normal	46	0	0	Normal	45	0	1
PTB	23	1	0	PTB	0	24	0
STB	8	0	0	STB	2	1	5
4-fold	Normal	PTB	STB	4-fold	Normal	PTB	STB
Normal	45	1	0	Normal	46	0	0
PTB	23	1	0	PTB	0	24	0
STB	8	0	0	STB	0	1	7
5-fold	Normal	PTB	STB	5-fold	Normal	PTB	STB
Normal	42	4	0	Normal	46	0	0
PTB	19	5	0	PTB	0	24	0
STB	5	3	0	STB	1	1	6
6-fold	Normal	PTB	STB	6-fold	Normal	PTB	STB
Normal	40	6	0	Normal	45	0	1
PTB	17	7	0	PTB	0	24	0
STB	6	2	0	STB	2	1	5
7-fold	Normal	PTB	STB	7-fold	Normal	PTB	STB
Normal	45	1	0	Normal	46	0	0
PTB	24	0	0	PTB	0	24	0
STB	8	0	0	STB	0	1	7
8-fold	Normal	PTB	STB	8-fold	Normal	PTB	STB
Normal	43	3	0	Normal	46	0	0
PTB	20	4	0	PTB	0	24	0
STB	7	1	0	STB	0	1	7
9-fold	Normal	PTB	STB	9-fold	Normal	PTB	STB
Normal	43	3	0	Normal	45	0	1
PTB	19	5	0	PTB	0	23	1
STB	7	1	0	STB	1	1	6
10-fold	Normal	PTB	STB	10-fold	Normal	PTB	STB
Normal	42	4	0	Normal	45	0	1
PTB	20	4	0	PTB	0	24	0
STB	7	1	0	STB	0	1	7

Contoh perhitungan nilai akurasi, sensitivitas dan spesifisitas dari uji validasi *2-fold cross-validation* pada sistem klasifikasi SVM dengan masukan GLCM tanpa optimasi pada kelas normal adalah sebagai berikut:

$$TP = 28$$

$$TN = 13 + 4 = 17$$

$$FN = 18$$

$$FP = 11 + 4 = 15$$

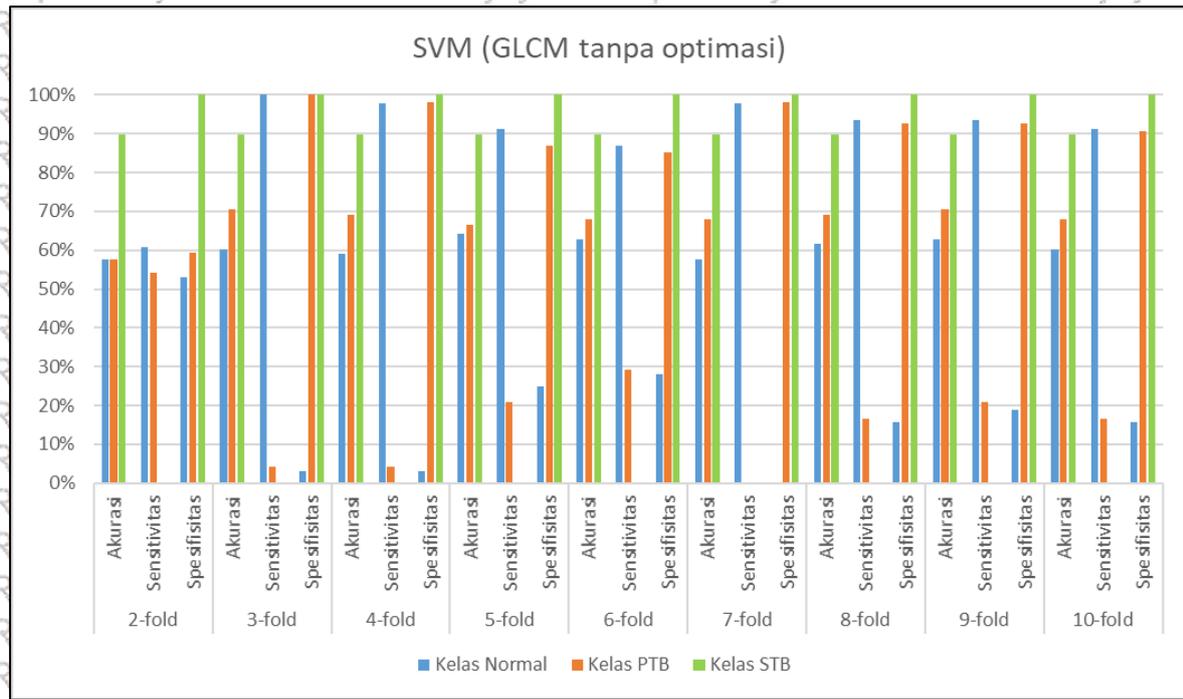
$$Akurasi = \frac{TP + TN}{TP + TN + FN + FP} = \frac{28 + 0}{28 + 17 + 18 + 15} = \frac{28}{78} = 57,69\%$$

$$Sensitivitas = \frac{TP}{TP + FN} = \frac{28}{28 + 18} = \frac{28}{46} = 60,87\%$$

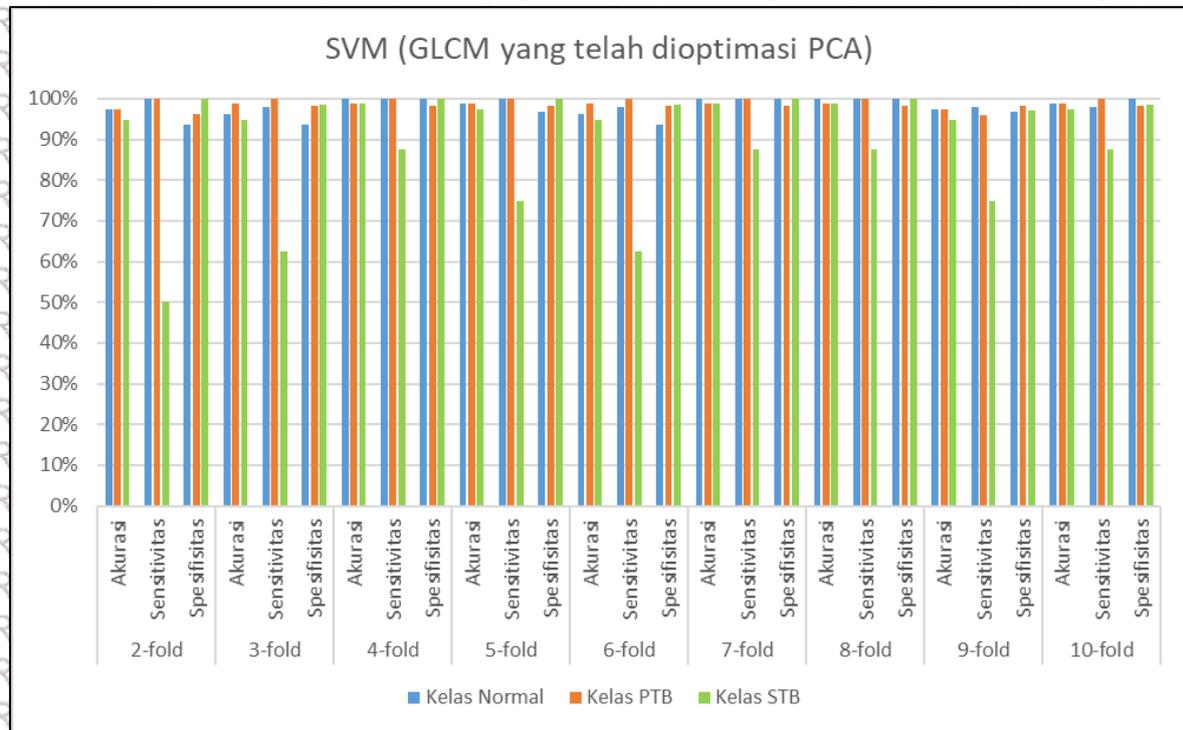
$$Spesifisitas = \frac{TN}{TN + FP} = \frac{17}{17 + 15} = \frac{17}{32} = 53,13\%$$

Grafik hasil perhitungan akurasi, sensitivitas dan spesifisitas dari uji validasi *2-fold* hingga *10-fold cross validation* sistem klasifikasi SVM dengan masukan GLCM tanpa optimasi ditunjukkan dalam *Gambar 5.11*. Grafik hasil perhitungan akurasi, sensitivitas dan spesifisitas dari uji validasi *2-fold* hingga *10-fold cross validation* sistem klasifikasi SVM dengan masukan GLCM yang telah dioptimasi dengan PCA ditunjukkan dalam *Gambar 5.12*. Hasil perhitungan akurasi, sensitivitas dan spesifisitas dari uji validasi *2-fold* hingga *10-fold cross validation* sistem klasifikasi SVM dengan masukan GLCM tanpa optimasi maupun GLCM yang telah dioptimasi dengan metode PCA ditunjukkan dalam Tabel 5.2.

Performansi sistem klasifikasi SVM dengan masukan GLCM yang telah dioptimasi dengan menggunakan metode PCA lebih baik dibandingkan dengan klasifikasi SVM dengan masukan GLCM tanpa optimasi. Hasil uji *cross-validation* sistem klasifikasi SVM dengan masukan GLCM tanpa optimasi PCA memiliki nilai tertinggi pada *5-fold* dengan nilai akurasi 64,10%, sensitivitas 91,30% dan spesifisitas 25% untuk kelas normal; nilai akurasi 66,67%, sensitivitas 20,83% dan spesifisitas 87,04% untuk kelas PTB; dan nilai akurasi 89,74%, sensitivitas 0% dan spesifisitas 100% untuk kelas STB. Hasil uji *cross-validation* pada SVM dengan masukan GLCM yang telah dioptimasi dengan metode PCA memiliki nilai tertinggi pada *7-fold* dengan nilai akurasi 100%, sensitivitas 100% dan spesifisitas 100% untuk kelas normal; nilai akurasi 98,72%, sensitifitas 100% dan spesifisitas 98,15% untuk kelas PTB; dan nilai akurasi 98,72%, sensitivitas 87,50% dan spesifisitas 100% untuk kelas STB.



Gambar 5.11 Grafik hasil perhitungan akurasi, sensitivitas dan spesifisitas dari uji *cross-validation* sistem dengan masukan GLCM tanpa optimasi



Gambar 5.12 Grafik hasil perhitungan akurasi, sensitivitas dan spesifisitas dari uji *cross-validation* sistem dengan masukan GLCM yang telah dioptimasi

Tabel 5.2 Hasil perhitungan akurasi, sensitivitas dan spesifisitas dari uji *cross-validation* dengan masukan GLCM tanpa optimasi maupun GLCM yang telah dioptimasi.

Uji Cross-validation Klasifikasi SVM									
GLCM tanpa optimasi		Kelas			GLCM yang telah dioptimasi dengan PCA		Kelas		
		Normal	PTB	STB			Normal	PTB	STB
uji_2-fold	Akurasi	57.69%	57.69%	89.74%	uji_2-fold	Akurasi	97.44%	97.44%	94.87%
	Sensitivitas	60.87%	54.17%	0.00%		Sensitivitas	100.00%	100.00%	50.00%
	Spesifisitas	53.13%	59.26%	100.00%		Spesifisitas	93.75%	96.30%	100.00%
uji_3-fold	Akurasi	60.26%	70.51%	89.74%	uji_3-fold	Akurasi	96.15%	98.72%	94.87%
	Sensitivitas	100.00%	4.17%	0.00%		Sensitivitas	97.83%	100.00%	62.50%
	Spesifisitas	3.13%	100.00%	100.00%		Spesifisitas	93.75%	98.15%	98.57%
uji_4-fold	Akurasi	58.97%	69.23%	89.74%	uji_4-fold	Akurasi	100.00%	98.72%	98.72%
	Sensitivitas	97.83%	4.17%	0.00%		Sensitivitas	100.00%	100.00%	87.50%
	Spesifisitas	3.13%	98.15%	100.00%		Spesifisitas	100.00%	98.15%	100.00%
uji_5-fold	Akurasi	64.10%	66.67%	89.74%	uji_5-fold	Akurasi	98.72%	98.72%	97.44%
	Sensitivitas	91.30%	20.83%	0.00%		Sensitivitas	100.00%	100.00%	75.00%
	Spesifisitas	25.00%	87.04%	100.00%		Spesifisitas	96.88%	98.15%	100.00%
uji_6-fold	Akurasi	62.82%	67.95%	89.74%	uji_6-fold	Akurasi	96.15%	98.72%	94.87%
	Sensitivitas	86.96%	29.17%	0.00%		Sensitivitas	97.83%	100.00%	62.50%
	Spesifisitas	28.13%	85.19%	100.00%		Spesifisitas	93.75%	98.15%	98.57%
uji_7-fold	Akurasi	57.69%	67.95%	89.74%	uji_7-fold	Akurasi	100.00%	98.72%	98.72%
	Sensitivitas	97.83%	0.00%	0.00%		Sensitivitas	100.00%	100.00%	87.50%
	Spesifisitas	0.00%	98.15%	100.00%		Spesifisitas	100.00%	98.15%	100.00%
uji_8-fold	Akurasi	61.54%	69.23%	89.74%	uji_8-fold	Akurasi	100.00%	98.72%	98.72%
	Sensitivitas	93.48%	16.67%	0.00%		Sensitivitas	100.00%	100.00%	87.50%
	Spesifisitas	15.63%	92.59%	100.00%		Spesifisitas	100.00%	98.15%	100.00%
uji_9-fold	Akurasi	62.82%	70.51%	89.74%	uji_9-fold	Akurasi	97.44%	97.44%	94.87%
	Sensitivitas	93.48%	20.83%	0.00%		Sensitivitas	97.83%	95.83%	75.00%
	Spesifisitas	18.75%	92.59%	100.00%		Spesifisitas	96.88%	98.15%	97.14%
uji_10-fold	Akurasi	60.26%	67.95%	89.74%	uji_10-fold	Akurasi	98.72%	98.72%	97.44%
	Sensitivitas	91.30%	16.67%	0.00%		Sensitivitas	97.83%	100.00%	87.50%
	Spesifisitas	15.63%	90.74%	100.00%		Spesifisitas	100.00%	98.15%	98.57%

5.2.2 Analisis Grafik ROC

Grafik ROC menggambarkan performansi sistem klasifikasi SVM pada penelitian ini. Grafik ROC yang dianalisis pada penelitian ini adalah grafik ROC dari *confusion matrix* SVM dengan nilai tertinggi yaitu pada *5-fold cross-validation* pada SVM dengan masukan GLCM tanpa optimasi dan *7-fold cross-validation* pada SVM dengan masukan GLCM yang telah dioptimasi dengan metode PCA. Contoh perhitungan TPR dan FPR pada klasifikasi SVM dengan masukan GLCM tanpa optimasi pada kelas normal dengan uji validasi *5-fold cross-validation* adalah sebagai berikut:

$$TP = 42$$

$$TN = 5 + 3 = 8$$

$$FN = 4$$

$$FP = 24$$

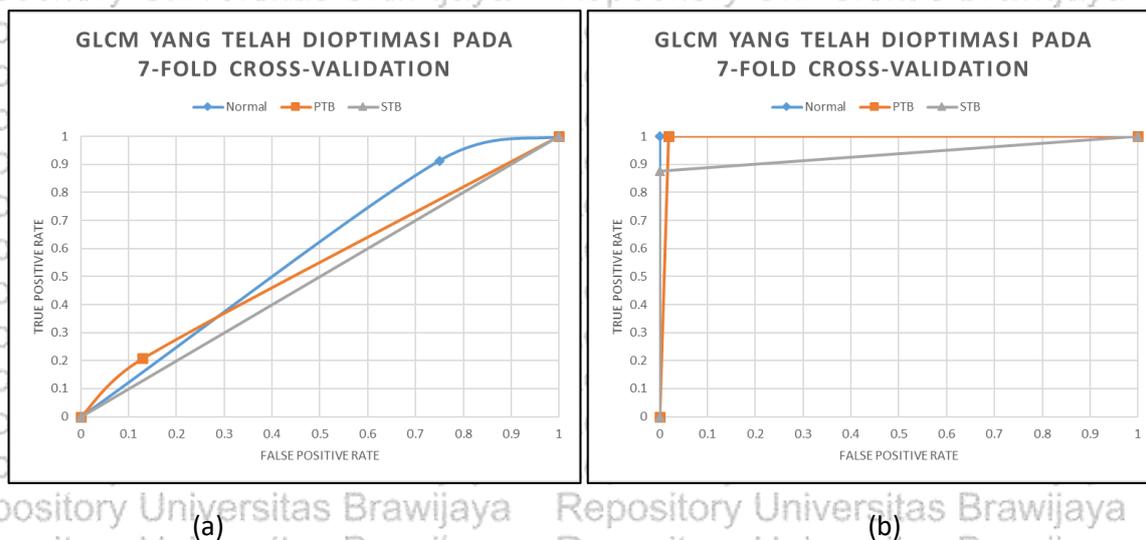
$$\text{True Positive Rate (TPR)} = \frac{TP}{TP + FN} = \frac{42}{42 + 4} = 0,913$$

$$\text{False Positive Rate (FPR)} = \frac{FP}{FP + TN} = \frac{24}{24 + 8} = 0,75$$

Hasil dari perhitungan TPR dan FPR dari uji validasi *2-fold* hingga *10-fold cross-validation* sistem klasifikasi SVM dengan masukan GLCM tanpa optimasi maupun GLCM yang telah dioptimasi dengan metode PCA ditunjukkan dalam Tabel 5.3. Nilai TPR dan FPR yang telah diperoleh digunakan untuk menggambarkan grafik ROC secara manual. Grafik ROC dari hasil perhitungan TPR dan FPR pada uji validasi *5-fold cross-validation* klasifikasi SVM dengan masukan GLCM tanpa optimasi ditunjukkan dalam Gambar 5.13.

Tabel 5.3 Hasil perhitungan TPR dan FPR dari uji *cross-validation* dengan masukan GLCM tanpa optimasi maupun GLCM yang telah dioptimasi.

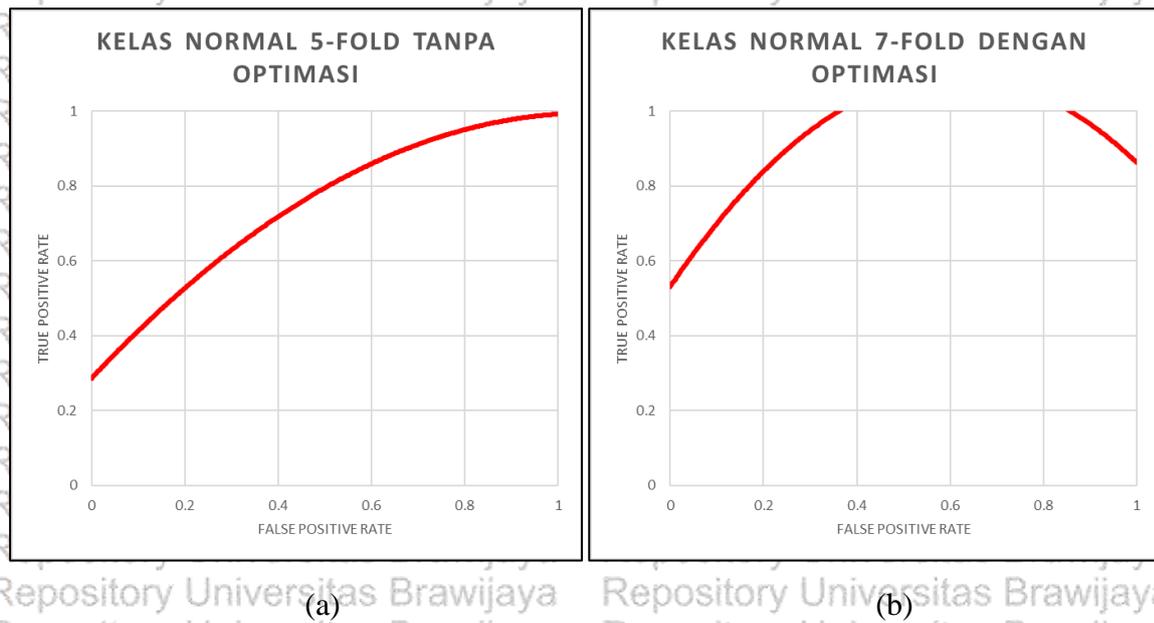
Uji Cross-validation									
GLCM tanpa optimasi		Kelas			GLCM yang telah dioptimasi dengan PCA		Kelas		
		Normal	PTB	STB			Normal	PTB	STB
uji_2-fold	TPR	0.608696	0.541667	0	uji_2-fold	TPR	1	1	0.5
	FPR	0.46875	0.407407	0		FPR	0.0625	0.037037	0
uji_3-fold	TPR	1	0.041667	0	uji_3-fold	TPR	0.978261	1	0.625
	FPR	0.96875	0	0		FPR	0.0625	0.018519	0.014286
uji_4-fold	TPR	0.978261	0.041667	0	uji_4-fold	TPR	1	1	0.875
	FPR	0.96875	0.018519	0		FPR	0	0.018519	0
uji_5-fold	TPR	0.913043	0.208333	0	uji_5-fold	TPR	1	1	0.75
	FPR	0.75	0.12963	0		FPR	0.03125	0.018519	0
uji_6-fold	TPR	0.869565	0.291667	0	uji_6-fold	TPR	0.978261	1	0.625
	FPR	0.71875	0.148148	0		FPR	0.0625	0.018519	0.014286
uji_7-fold	TPR	0.978261	0	0	uji_7-fold	TPR	1	1	0.875
	FPR	1	0.018519	0		FPR	0	0.018519	0
uji_8-fold	TPR	0.934783	0.166667	0	uji_8-fold	TPR	1	1	0.875
	FPR	0.84375	0.074074	0		FPR	0	0.018519	0
uji_9-fold	TPR	0.934783	0.208333	0	uji_9-fold	TPR	0.978261	0.958333	0.75
	FPR	0.8125	0.074074	0		FPR	0.03125	0.018519	0.028571
uji_10-fold	TPR	0.913043	0.166667	0	uji_10-fold	TPR	0.978261	1	0.875
	FPR	0.84375	0.092593	0		FPR	0	0.018519	0.014286



Gambar 5.13 Grafik ROC dari perhitungan manual; (a) 5-fold cross-validation dengan masukan GLCM tanpa optimasi, (b) 7-fold cross-validation dengan masukan GLCM yang telah dioptimasi.

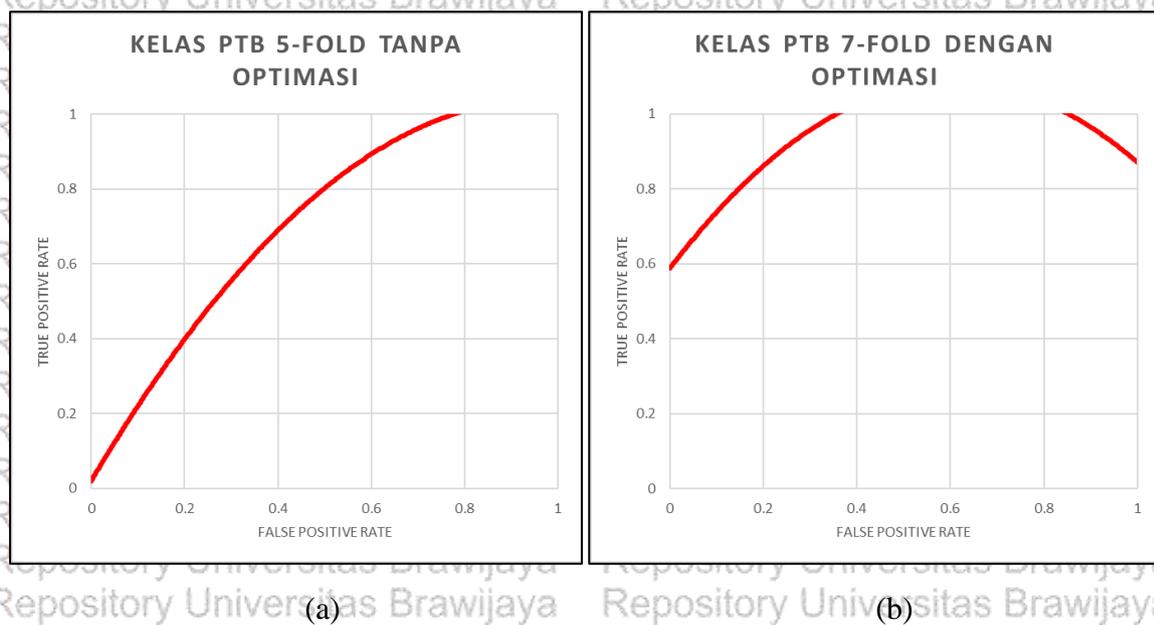
Gambar 5.13 (a) menunjukkan bahwa sistem klasifikasi SVM dengan masukan GLCM tanpa optimasi pada uji validasi 5-fold cross-validation memiliki performansi yang sangat buruk. Sistem klasifikasi pada kelas normal cenderung mengklasifikasikan hampir semua masukan ke dalam kelas normal. Sistem klasifikasi pada kelas PTB performansinya hampir menyamai sistem klasifikasi acak. Sistem klasifikasi pada kelas STB merupakan sistem klasifikasi yang gagal, dimana sistem klasifikasi tidak mampu mengklasifikasikan dengan tepat sama sekali ke dalam kelas STB. Gambar 5.13 (b) menunjukkan bahwa sistem klasifikasi SVM dengan masukan GLCM yang telah dioptimasi dengan metode PCA pada uji validasi 7-fold cross-validation memiliki performansi yang sangat bagus. Sistem klasifikasi SVM dengan masukan GLCM yang telah dioptimasi dengan metode PCA mampu mengklasifikasi kelas normal dengan sempurna, sedangkan klasifikasi pada kelas PTB dan STB terdapat kesalahan klasifikasi yang sangat kecil.

Grafik ROC keluaran dari Weka 3.8 pada uji validasi 5-fold cross-validation klasifikasi SVM dengan masukan GLCM tanpa optimasi dan 7-fold cross-validation klasifikasi SVM dengan masukan GLCM yang telah dioptimasi dengan menggunakan metode PCA ditunjukkan dalam Gambar 5.14 hingga Gambar 5.16.



Gambar 5.14 Grafik ROC Weka 3.8 pada kelas normal; (a) 5-fold cross-validation dengan masukan GLCM tanpa optimasi. (b) 7-fold cross-validation dengan masukan GLCM yang telah dioptimasi.

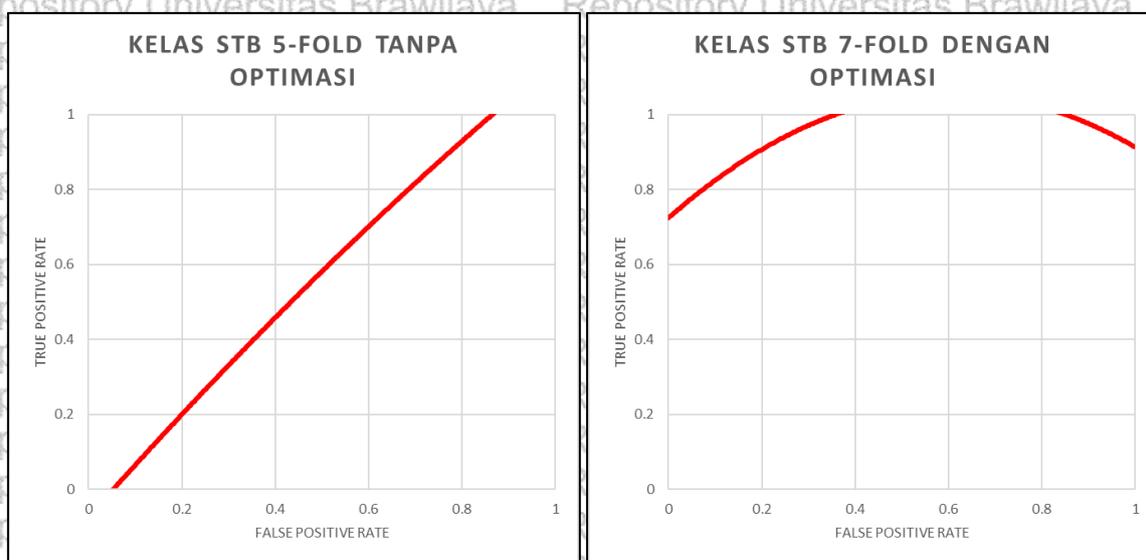
Gambar 5.14 (a) menunjukkan bahwa pada uji validasi 5-fold cross-validation klasifikasi SVM dengan masukan GLCM tanpa optimasi pada kelas normal memiliki performansi yang sedang. Gambar 5.14 (b) menunjukkan bahwa pada uji validasi 7-fold cross-validation klasifikasi SVM dengan masukan GLCM yang telah dioptimasi dengan metode PCA pada kelas normal memiliki performansi yang sangat baik.



Gambar 5.15 Grafik ROC Weka 3.8 pada kelas PTB; (a) pada uji validasi 5-fold cross-validation klasifikasi SVM dengan masukan GLCM tanpa optimasi. (b) 7-fold cross-

validation klasifikasi SVM dengan masukan GLCM yang telah dioptimasi dengan metode PCA.

Gambar 5.15 (a) menunjukkan bahwa pada uji validasi *5-fold cross-validation* klasifikasi SVM dengan masukan GLCM tanpa optimasi pada kelas PTB memiliki performansi yang sedang. *Gambar 5.15 (b)* menunjukkan bahwa pada uji validasi *7-fold cross-validation* klasifikasi SVM dengan masukan GLCM yang telah dioptimasi dengan metode PCA pada kelas PTB memiliki performansi yang sangat baik.



(a)

(b)

Gambar 5.16 Grafik ROC Weka 3.8 pada kelas STB; (a) *5-fold cross-validation* dengan masukan GLCM tanpa optimasi, (b) *7-fold cross-validation* dengan masukan GLCM yang telah dioptimasi.

Gambar 5.16 (a) menunjukkan bahwa pada uji validasi *5-fold cross-validation* klasifikasi SVM dengan masukan GLCM tanpa optimasi pada kelas STB merupakan sistem klasifikasi yang buruk. *Gambar 5.16 (b)* menunjukkan bahwa pada uji validasi *7-fold cross-validation* klasifikasi SVM dengan masukan GLCM yang telah dioptimasi dengan metode PCA pada kelas STB memiliki performansi yang sangat baik.

Dalam *Gambar 5.14*, hingga *Gambar 5.16* ditunjukkan bahwa performansi sistem dengan masukan GLCM yang telah dioptimasi dengan PCA memiliki performansi yang lebih baik dibandingkan dengan sistem dengan masukan GLCM

yang belum dioptimasi. Grafik ROC dari sistem dengan masukan GLCM yang telah dioptimasi memiliki performansi yang sangat baik pada ketiga kelas klasifikasi. Pada grafik ROC sistem dengan masukan GLCM yang tidak dioptimasi menunjukkan performansi yang sedang pada kelas normal dan PTB, sedangkan pada kelas STB menunjukkan performansi yang buruk.



BAB VI

KESIMPULAN DAN SARAN

Bab VI berisi tentang kesimpulan beserta saran untuk penelitian yang telah dilakukan pada tesis.

5.3 Kesimpulan

Sistem deteksi TB pada citra CXR telah berhasil didesain pada penelitian ini. Hasil penelitian tentang sistem deteksi TB pada citra CXR dapat disimpulkan sebagai berikut:

1. Sistem deteksi TB pada citra CXR dengan masukan fitur GLCM yang telah dioptimasi dengan PCA dan diklasifikasi dengan metode SVM berhasil dibangun pada penelitian ini. Sistem optimasi PCA mampu mengurangi fitur GLCM dari dua belas fitur menjadi dua fitur baru.
2. Sistem deteksi TB pada citra CXR dengan masukan fitur GLCM yang telah dioptimasi dengan PCA memiliki performansi lebih baik bila dibandingkan dengan sistem deteksi TB dengan masukan fitur GLCM tanpa optimasi. Sistem deteksi TB dengan fitur GLCM yang belum dioptimasi memiliki nilai tertinggi pada uji *5-fold cross-validation* dengan nilai akurasi 64,10%, sensitivitas 91,30% dan spesifisitas 25% untuk kelas normal; nilai akurasi 66,67%, sensitivitas 20,83% dan spesifisitas 87,04% untuk kelas PTB; dan nilai akurasi 89,74%, sensitivitas 0% dan spesifisitas 100% untuk kelas STB. Sistem deteksi TB dengan fitur GLCM yang telah dioptimasi memiliki nilai tertinggi pada uji *7-fold cross-validation* dengan nilai akurasi 100%, sensitivitas 100% dan spesifisitas 100% untuk kelas normal; nilai akurasi 98,72%, sensitifitas 100% dan spesifisitas 98,15% untuk kelas PTB; dan nilai akurasi 98,72%, sensitivitas 87,50% dan spesifisitas 100% untuk kelas STB.
3. Kurva ROC sistem deteksi TB dengan masukan fitur GLCM yang telah dioptimasi dengan menggunakan metode PCA meunjukkan performansi yang lebih baik dibandingkan dengan kurva ROC pada sistem deteksi TB dengan masukan fitur GLCM yang belum dioptimasi. Kurva ROC pada sistem deteksi TB dengan masukan fitur GLCM yang telah dioptimasi dengan PCA pada uji *7-fold cross-validation* menunjukkan bahwa sistem memiliki performa yang sangat bagus, hal ini

ditunjukkan dengan nilai *area under ROC* sebesar 1 pada kelas normal, 0,9992 pada kelas PTB dan 1 pada kelas STB. Kurva ROC pada sistem deteksi TB dengan masukan GLCM tanpa optimasi PCA pada uji *5-fold cross-validation* yang menunjukkan bahwa sistem memiliki performansi kurang baik, hal ini ditunjukkan dengan nilai *area under ROC* sebesar 0,7643 pada kelas normal, 0,7037 pada kelas PTB dan 0,5696 pada kelas STB.

5.4. Saran

Saran untuk pengembangan penelitian ialah sebagai berikut:

1. Sistem dapat dikembangkan dengan data yang memiliki komposisi yang seimbang untuk tiap kelasnya.
2. Penelitian dapat dikembangkan dengan menggunakan fitur klasifikasi lain seperti HOG dan LBP.
3. Optimasi pada fitur klasifikasi dapat divariasikan menggunakan metode optimasi lain.
4. Sistem klasifikasi yang digunakan pada penelitian dapat dikembangkan menggunakan sistem kecerdasan buatan (*artificial neural network*).