

KLASIFIKASI *DIABETIC RETINOPATHY* MENGGUNAKAN SELEKSI FITUR DAN SUPPORT VECTOR MACHINE

Muhammad Imron Rosadi¹, Cahya Bagus Sanjaya², Lukman Hakim³

^{1,2,3}Teknik Informatika, Universitas Yudharta Pasuruan
 Pasuruan, Indonesia

e-mail: imron.rosadi@yudharta.ac.id¹, cahya@yudharta.ac.id², lukman@yudharta.ac.id³

Received : Month, Year	Accepted : Month, Year	Published : Oktober, 2018
------------------------	------------------------	---------------------------

Abstract

Diabetic Retinopathy is a disease common complications of diabetes mellitus. The complications in the form of damages on the part of the retina of the eye. The high levels of glucose in the blood are the cause of small capillaries become broke and can lead to blindness. The symptoms shown by the sufferers of Diabetic Retinopathy (DR), among others, microaneurysms, hemorrhages, exudates, soft hard exudate and neovascularization. These symptoms are at a certain intensity can be an indicator of the phase (the level of severity) DR sufferers. There are four stages of the process of pattern recognition, namely preprocessing, feature extraction, feature selection and classification. On preprocessing the image do Change the RGB image into Green channel, image Adaptive Histogram Equalization, removal of blood vessels, removal of optic disks, detection of exudate. A collection from the results of preprocessing placed in the vector of characteristics by using the feature extraction of GLCM consisting of order 1 and 2, to order then conducted as input Support Vector Machine (SVM). While in SVM there are three issues that emerged, namely; How to select a kernel function, what is the optimal number of input features, and how to determine the best kernel parameters. These issues are important, because the number of features affect the required kernel parameters values and vice versa, so that the selection of the features required in building the classification system. On the research of feature extraction methods was presented GLCM, features selection, and SVM for detecting diabetic retinopathy. feature selection process using the F-Score feature to select the results of features extraction. From the results of the selection of these features is used to input the classification. The dataset used amounted to 50 data, which is divided into 2 classes, where 25 sets taken from normal retinal scans and 25 sets of the rest of the scan of the retina with diabetic retinopathy. SVM classification with feature selection to increase accuracy and computational time than lose without a selection of features with a value of 90% accuracy and computational time 0.010 seconds.

Keywords: *Diabetic Retinopathy, Preprocessing, GLCM, Features Selection, SVM*

Abstrak

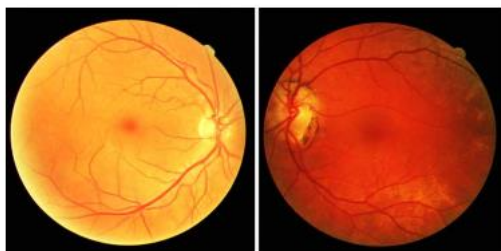
Diabetic Retinopathy merupakan sebuah penyakit komplikasi umum dari diabetes militus. Komplikasi tersebut berupa kerusakan pada bagian retina mata. Tingginya kadar glukosa dalam darah adalah penyebab pembuluh darah kapiler kecil menjadi pecah dan dapat menyebabkan kebutaan. Gejala yang ditunjukkan oleh penderita Diabetic Retinopathy (DR) antara lain microaneurysms, hemorrhages, hard exudate, soft exudates dan neovascularization. Gejala tersebut pada suatu intensitas tertentu dapat menjadi indikator fase (tingkatan keparahan) retinopati diabetes penderita. Terdapat empat tahap proses pengenalan pola, yaitu praproses, ekstraksi, seleksi dan klasifikasi. pada praproses dilakukan Perubahan citra RGB ke Grayscale citra kanal hijau, Adaptif Histogram Equalization, penghilangan

pembuluh darah, penghilangan optic disk, deteksi eksudat . Kumpulan dari hasil praproses dimasukan dalam vektor ciri dengan menggunakan ekstraksi fitur GLCM yang terdiri dari orde 1 dan orde 2, untuk kemudian dilakukan sebagai input Support Vector Machine (SVM). Sedangkan pada SVM terdapat tiga masalah yang muncul, yaitu; bagaimana memilih fungsi kernel, berapakah jumlah fitur input yang optimal, dan bagaimana menentukan parameter kernel terbaik. Masalah-masalah tersebut penting, karena jumlah fitur mempengaruhi nilai parameter kernel yang diperlukan dan sebaliknya, sehingga seleksi fitur diperlukan dalam membangun sistem klasifikasi. Pada penelitian ini diajukan metode ekstraksi fitur GLCM, seleksi fitur, dan SVM untuk mendeteksi diabetes retinopati. proses seleksi fitur menggunakan F-Score untuk memilih fitur hasil ekstraksi fitur. Dari hasil seleksi fitur tersebut digunakan untuk input klasifikasi. Dataset yang digunakan berjumlah 50 data, dimana terbagi menjadi 2 kelas, dimana 25 set diambil dari scan retina normal dan 25 set sisanya dari scan retina dengan diabetes retinopati. Klasifikasi SVM dengan seleksi fitur mampu meningkatkan akurasi dan menurunkan waktu komputasi daripada tanpa seleksi fitur dengan nilai akurasi 90% dan waktu komputasi 0.010 detik.

Kata Kunci: Diabetic Retinopathy, Praproses, GLCM, Seleksi Fitur, SVM

1. PENDAHULUAN

Penelitian citra retina untuk deteksi penyakit yang terdapat pada retina bertujuan untuk mendapatkan informasi penting misalnya informasi tentang tingkat resiko penyakit *diabetic retinopathy*. *diabetic retinopathy* merupakan komplikasi umum dari Diabetes Melitus. Pada tahap awal penderita Diabetes Retinopati tidak akan mengalami penurunan tajam penglihatan, tetapi akan mendadak kehilangan penglihatan jika telah terjadi kerusakan yang sangat parah pada retina [1].



Gambar 1: Retina mata sehat (kiri), retina mata kelainan (kanan).

Sumber: Amin dan Yan, 2011 [2]

Diabetic Retinopathy ditandai dengan adanya gangguan pembuluh darah di retina berupa kebocoran, sumbatan dan pada tahap selanjutnya timbul pembuluh darah tidak normal yang sangat rapuh dan mudah menimbulkan pendarahan dengan segala akibat yang merugikan [3]. Diagnosis awal Diabetes Retinopati biasanya dilakukan dengan mengamati secara langsung citra retina dari kamera fundus tetapi memerlukan waktu yang relatif lama untuk mengetahui hasilnya. dari 6,9% penderita diabetes melitus yang didapatkan, 30,4% yang telah didiagnosis

sebelumnya dan 69,6% tidak didiagnosis sebelumnya. Meskipun peningkatan proporsi penderita diabetes melitus yang terdiagnosis namun proporsi yang tidak terdiagnosis sebelumnya masih besar[4].

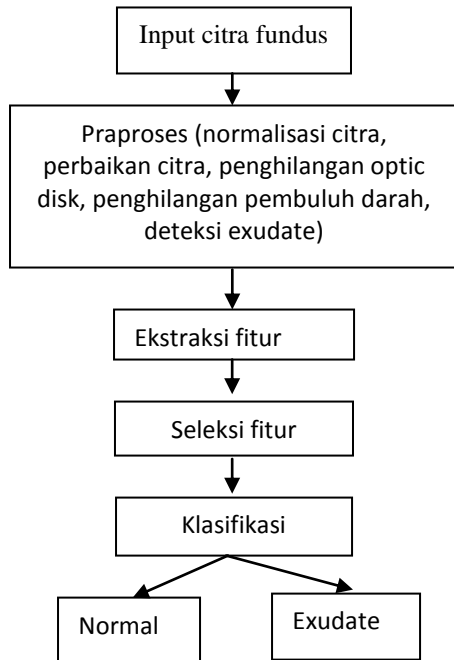
Penerapan metode data mining sangat membantu untuk menganalisis permasalahan tersebut. Salah satu metode terbaik adalah SVM. Beberapa penelitian yang menggunakan metode SVM. Gandhi dan Dhanasekaran menggunakan Teknik praproses citra, ekstraksi menggunakan GLCM orde 1 sebagai input SVM untuk klasifikasi diabetes retinopati [5]. Layadi, dkk menggunakan metode Support Vector Machine untuk menganalisis data pada jaringan sensor nirkabel. Hasil akurasi yang dicapai SVM kernel Gaussian lebih baik dari kernel Polynomial dan linear [6]. Wahyudi, dkk melakukan ekstraksi ciri 2DLDA dan klasifikasi Support Vector Machine (SVM) menghasilkan akurasi 84% [7]. Dari beberapa penelitian tersebut belum menggunakan seleksi fitur dan menemukan fitur mana yang optimal dari beberapa fitur input yang digunakan untuk klasifikasi.

Ada salah satu metode seleksi fitur yang diusulkan oleh Chen & Lin yaitu seleksi fitur F-Score. Kombinasi metode tersebut (SVM dan F-Score) telah digunakan untuk mendiagnosis penyakit kanker payudara dan menghasilkan tingkat akurasi yang lebih baik dari LS-SVM untuk kasus yang sama [8]. Pada penelitian ini diajukan metode ekstraksi fitur GLCM, seleksi fitur, dan klasifikasi menggunakan SVM untuk mendeteksi *Diabetic*

Retinopathy. Metode seleksi fitur menggunakan F-Score tersebut diharapkan mampu meningkatkan hasil akurasi, mengurangi waktu komputasi, serta untuk mengetahui ciri mana yang secara signifikan mempengaruhi tingkat akurasi sistem.

2. METODE PENELITIAN

Tahapan-tahapan yang dilakukan pada penelitian ini seperti pada gambar 2.



Gambar 2: langkah-langkah deteksi diabetic retinopathy

2.1 Data Uji Coba

Pada penelitian ini, data yang digunakan adalah 50 citra fundus dari *database* Citra didapatkan dari STructured Analysis of the Retina (STARE) dataset yang terdiri dari 40 data training (20 Citra retina normal dan 20 citra retina exudate) dan data testing sebanyak 10 (5 Citra retina normal dan 5 citra retina exudate).

2.2 Praproses

2.2.1 Normalisasi citra

Normalisasi dilakukan untuk merubah ukuran citra dari 1500 x 1152 piksel menjadi 700 x 500 piksel hal ini dilakukan untuk mereduksi ukuran agar eksekusi tidak terlalu lama.

2.2.2 Konversi citra RGB ke Green Channel

Pengambilan green channel merupakan proses awal yang dilakukan pada sebuah citra retina

dengan cara mengambil green channel citra retina, yang selanjutnya akan digunakan untuk proses segmentasi. Tujuannya yaitu untuk mendapatkan citra yang menampilkan pembuluh darah yang jelas. Green channel dipilih karena menghasilkan citra yang paling baik dibanding kan dengan red channel dan blue channel. Pada persamaan (1)

$$g = \frac{G}{R + G + B} \quad (1)$$

2.3 Peningkatan Kontras citra adaptif histogram

Peningkatan kontras citra menggunakan adaptif histogram untuk memperjelas gambar pembuluh darah, optic disk, dan area exudate.

2.4 Penghilangan Pembuluh darah dan Optic Disk

Untuk penghilangan pembuluh darah dilakukan proses morfologi yaitu operasi *closing*. Operasi closing yaitu penggabungan operasi dilasi dan erosi. Fungsi operasi dilasi untuk memperluas area optic disk. Sedangkan untuk erosi untuk menghilangkan pembuluh darah. Operasi closing dari himpunan A oleh strel B dinyatakan oleh $A * B$, didefinisikan sebagai:

$$A * B = (A \oplus B) \ominus B \quad (2)$$

2.3 Ekstraksi Fitur GLCM

Pengukuran nilai tekstur yang digunakan didasarkan pada persamaan (Haralic dan corner [9]). Menggunakan notasi berikut: G adalah jumlah tingkat abu-abu yang digunakan. m adalah nilai rata-rata dari P_x , μ_x , μ_y , σ_x and σ_y adalah *means* dan *standard deviations* P_x dan P_y . $P_x(i)$ adalah entri-i dalam matrik smarginal-probabilitas yang diperoleh dengan menjumlahkan barisan $P(i, j)$. dalam penelitian ini menggunakan GLCM orde pertama dan kedua.

2.4 Seleksi Fitur F-Score

F-score adalah teknik sederhana yang mengukur diskriminan dua himpunan bilangan real. Pada vektor training x_k , dengan $k = 1, 2, \dots, m$, jika jumlah kasus positif dan negatif adalah n_+ dan n_- , maka F-score masing-masing fitur i didefinisikan sebagai:

$$F_i = \frac{(\bar{x}_i^{(+)} - \bar{x}_i)^2 + (\bar{x}_i^{(-)} - \bar{x}_i)^2}{\frac{1}{n_+ - 1} \sum_{k=1}^{n_+} (x_{k,i}^{(+)} - \bar{x}_i^{(+)})^2 + \frac{1}{n_- - 1} \sum_{k=1}^{n_-} (x_{k,i}^{(-)} - \bar{x}_i^{(-)})^2} \quad (3)$$

di mana $\bar{x}_i^{(+)}$, $\bar{x}_i^{(-)}$ adalah rata-rata dari fitur ke-i keseluruhan, dataset positif, dan negatif, $x_{k,i}^{(+)}$ adalah fitur ke-i dari kasus positif ke-k, dan $x_{k,i}^{(-)}$ adalah fitur ke-i dari kasus negatif ke-k. Pembilang menunjukkan diskriminasi antara himpunan positif dan negatif, dan penyebut menunjukkan fitur-fitur dalam dua himpunan. Semakin besar F-score, kemungkinan fitur lebih diskriminatif semakin besar pula.

Metode seleksi fitur dengan F-Score dilakukan dengan menghitung nilai F-Score semua fitur. Nilai F-Score masing-masing fitur tersebut kemudian diurutkan secara descending. Sesuai dengan metode hill climbing, dipilih satu fitur dengan nilai F-Score tertinggi, kemudian dimasukkan ke classifier. Fitur tersebut kemudian dikombinasikan dengan satu fitur dari fitur sisa. Hal tersebut diulang sampai semua fitur telah digabungkan. Jika didapatkan kombinasi dengan akurasi tertinggi, maka kombinasi fitur tersebut adalah kombinasi yang optimal.

2.5 Support Vector Machines (SVM)

SVM yang diusulkan oleh Vapnik [10] telah dipelajari secara ekstensif untuk klasifikasi, regresi dan estimasi kepadatan. Gambar 2. adalah arsitektur SVM. SVM memetakan pola input ke ruang fitur dimensi yang lebih tinggi melalui pemetaan nonlinier berdasar teori yang dipilih. Bidang pemisah linear ini kemudian dibangun dalam ruang fitur dimensi tinggi. Dengan demikian, SVM adalah *linear classifier* di ruang parameter, tapi itu menjadi *nonlinear classifier* sebagai akibat dari pemetaan *nonlinear* dari ruang pola input ke ruang fitur dimensi tinggi. Bila data pelatihan berdimensi m adalah x_i ($i = 1, \dots, M$) dan masing-masing kelas labelnya adalah y_i , di mana $y_i = 1$ dan $y_i = -1$ untuk kelas 1 dan 2. Jika data input terpisah secara linear di ruang fitur, maka fungsi keputusan dapat ditentukan:

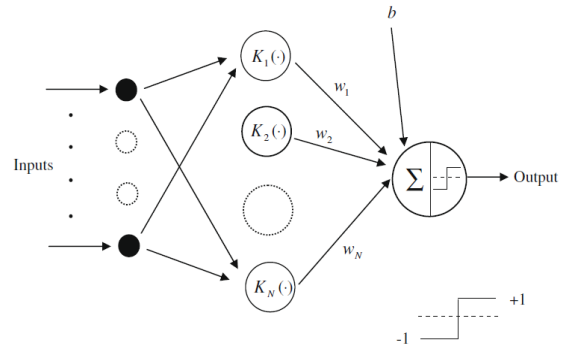
$$D(x) = w^t g(x) + b \quad (3)$$

di mana $g(x)$ adalah fungsi pemetaan yang memetakan x ke dalam ruang dimensi 1, w adalah vektor dimensi n , dan b adalah skalar. Untuk memisahkan data secara linier, fungsi keputusan memenuhi kondisi berikut:

$$y_i(w^t g(x_i) + b) \geq 1 \text{ untuk } i = 1, \dots, M \quad (4)$$

Jika masalah terpisah secara linier dalam ruang fitur, maka fungsi keputusan yang memenuhi

persamaan (16) jumlahnya tak terbatas. Di antara fungsi-fungsi tersebut, diperlukan *hyperplane* dengan margin terbesar antara dua kelas. Margin adalah jarak minimum yang memisahkan *hyperplane* terhadap data input dan ini dihasilkan dari $|D(x)|/||w||$. Sehingga didapatkan *hyperplane* pemisah dengan margin maksimal yang optimal memisahkan *hyperplane*.



Gambar 2. Arsitektur SVM
Dengan asumsi bahwa margin adalah ρ , kondisi berikut harus memenuhi:

$$\frac{y_i D(x_i)}{||w||} \geq \rho \text{ untuk } i = 1, \dots, M \quad (5)$$

Hasil perkalian produk dari ρ dan $||w||$ adalah tetap:

$$\rho ||w|| = 1 \quad (6)$$

Untuk mendapatkan *hyperplane* pemisah yang optimal dengan margin maksimal, w dengan $||w||$ yang memenuhi persamaan (6) harus ditemukan. Persamaan (7) mengarahkan ke pemecahan masalah optimasi berikutnya. Dengan meminimalkan

$$\frac{1}{2} w^t w \quad (7)$$

dan mengikuti batasan:

$$y_i(w^t g(x_i) + b) \geq 1 \text{ untuk } i = 1, \dots, M$$

Bila data pelatihan tidak linier dipisahkan, digunakan *slack variable ξ_i ke persamaan (8):*

$$y_i(w^t g(x_i) + b) \geq 1 - \xi_i \quad \xi_i \geq 0 \text{ untuk } i = 1, \dots, M$$

Hyperplane pemisah yang optimal telah ditentukan sehingga maksimalisasi dari margin dan meminimalisasi dari kesalahan *training* didapatkan. Dengan meminimalkan

$$\frac{1}{2} w^t w + \frac{C}{2} \sum_{i=1}^n \xi_i^2 \quad (8)$$

mengikuti batasan:

$y_i(w^T g(x_i) + b) \geq 1 - \xi_i$, $\xi_i \geq 0$ untuk $i = 1, \dots, M$
 di mana C adalah parameter yang menentukan *tradeoff* antara margin maksimum dan kesalahan klasifikasi minimum dan ρ adalah 1 atau 2. Jika $\rho = 1$, SVM disebut SVM dengan *soft margin* L1 (L1-SVM), dan jika $\rho = 2$, SVM dengan *soft margin* L2 (L2-SVM). Pada SVM konvensional, *hyperplane* pemisah yang optimal diperoleh dengan memecahkan masalah pemrograman kuadratik.

Fungsi kernel memungkinkan operasi yang akan dilakukan di ruang input bukan di ruang fitur dimensi tinggi. Beberapa contoh fungsi kernel adalah $K(u, v) = v^T u$ (SVM linier); $K(u, v) = (v^T u + 1)^n$ (SVM polinomial derajat n); $K(u, v) = \exp(-||u - v||^2 / 2\sigma^2)$ (SVM fungsi *radial bases* – SVM RBF); $K(u, v) = \tanh(Kv^T y + o)$ (*neural SVM* dua layer) di mana σ, κ, o adalah konstanta (Vapnik, 1995; Cortes, 1995). Namun, fungsi kernel yang tepat untuk suatu masalah tertentu tergantung pada data, dan sampai saat ini belum ada metode yang baik tentang cara memilih fungsi kernel.

2.8 Evaluasi Kinerja Sistem

Evaluasi dilakukan dengan tujuan untuk mengevaluasi efektivitas metode dan sistem yang telah dibuat. Evaluasi dilakukan terhadap tingkat akurasi klasifikasi dan tingkat kesalahan klasifikasi. Ukuran atau parameter yang digunakan untuk evaluasi antara lain akurasi klasifikasi, sensitivitas, spesifisitas, nilai prediksi positif, nilai prediksi negatif, kurva ROC dan matriks konfusi (*confusion matrix*). Matriks konfusi berisi informasi tentang klasifikasi yang sebenarnya dan yang diperkirakan dari hasil sistem klasifikasi. Tabel 2.1 menunjukkan matriks konfusi untuk dua kelas klasifikasi. Akurasi klasifikasi, sensitivitas, spesifisitas, nilai prediksi positif dan nilai prediksi negatif dapat didefinisikan menggunakan elemen-elemen matriks konfusi sebagai berikut:

- Klasifikasi akurasi (%) = $\frac{TP+TN}{TP+FP+FN+TN}$,
- Sensitivitas (%) = $\frac{TP}{TP+FN}$,
- Spesifisitas (%) = $\frac{TN}{FP+TN}$,
- Nilai prediksi positif = $\frac{TP}{TP+FP} \times 100$,
- Nilai prediksi negatif = $\frac{TN}{FN+TN} \times 100$.

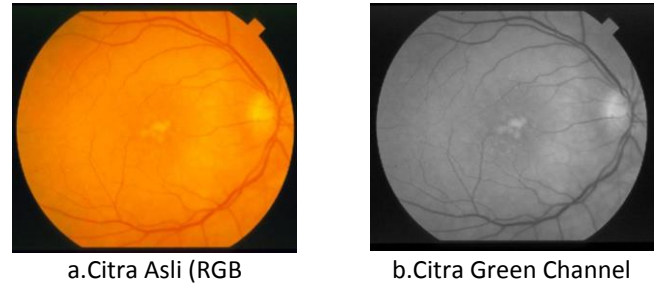
Tabel 1: Matriks Konfusi

Aktual	Prediksi	
	Positif	Negatif
Positif	True Positive (TP)	False Negative (FN)
Negatif	False Positive (FP)	True Negative (TN)

3. HASIL DAN PEMBAHASAN

3.1 Praproses

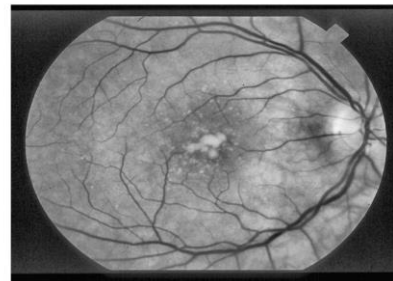
3.1.1 Konversi citra RGB ke Green Channel



a.Citra Asli (RGB) b.Citra Green Channel

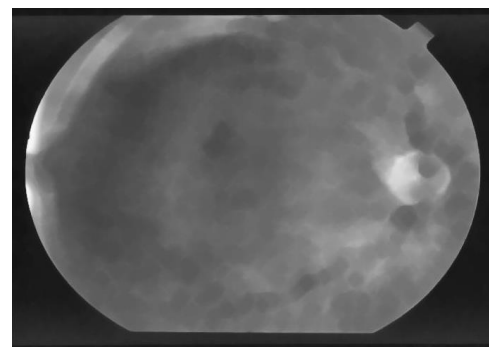
Gambar 3: Hasil konversi citra RGB ke green channel

3.1.2 Peningkatan kontras



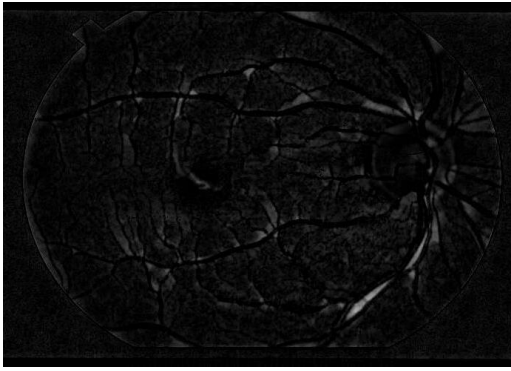
Gambar 4: hasil proses peningkatan kontras citra

3.1.3 Penghapusan pembuluh darah dan optic disk



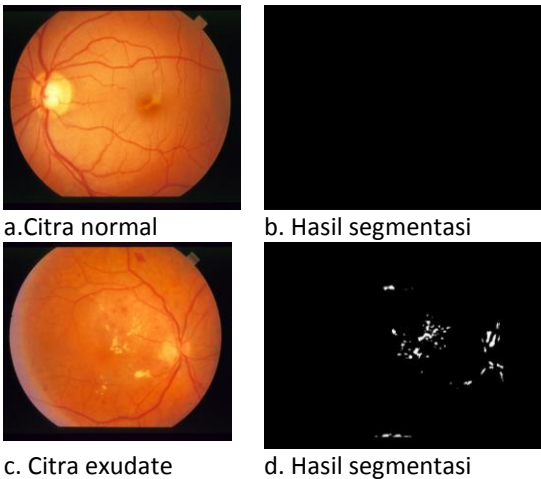
Gambar 5: hasil proses penghapusan pembuluh darah (28)

(28)
 (29)
 (30)
 (31)
 (32)



Gambar 6: hasil proses penghapusan optic disk

3.1.4 Hasil deteksi citra normal dan eksudat



a. Citra normal

b. Hasil segmentasi

c. Citra exudate

d. Hasil segmentasi

Gambar 7: hasil proses deteksi citra normal dan Diabetic Retinopathy

3.2 Ekstraksi Fitur

Proses ekstraksi fitur dilakukan terhadap 20 data *training* dan 20 data *testing* yang mana setiap data menghasilkan 25 fitur menggunakan metode GLCM. Dari hasil ekstraksi 25 fitur tersebut yang nantinya dijadikan untuk seleksi fitur klasifikasi. Tabel 2 adalah salah satu ekstraksi fitur.

Tabel 2: Hasil ekstraksi fitur dari salah satu citra dataset *training*

No	Fitur ciri	Nilai
1	Autocorrelation	0.0246
2	Contrast	0.0161
3	Correlation	0.0027
4	Correlation	0.0027
5	Cluster Prominence	0.0267
6	Cluster Shade	0.0261

7	Dissimilarity	0.0161
8	Energy	0.0217
9	Entropy	0.0195
10	Homogeneity	0.0161
11	Homogeneity	0.0161
12	Maximum probability	0.0217
13	Sum of squares: Variance	0.0236
14	Sum average	0.0236
15	Sum variance	0.0164
16	Sum entropy	0.0197
17	Difference variance	0.0161
18	Difference variance	1.0000
19	Difference entropy	0.0142
20	Information measure of correlation1	0.0010
21	Information measure of correlation2	0.0157
22	maximal correlation coefficient	1.0000
23	Inverse difference (INV) is homom	1.0000
24	Inverse difference normalized (INN)	0.0161
25	Inverse difference moment normalized	0.0161

3.3 Seleksi Fitur

Proses seleksi fitur dilakukan dengan menghitung nilai *F-Score* dari data *training*. Perhitungan nilai *F-Score*. Dari perhitungan nilai *F-Score* diperoleh dua belas fitur. Nilai masing-masing *F-Score* yang telah dihasilkan diurutkan secara menurun (*descending*) dengan fungsi **sort** yang telah disediakan oleh **Matlab**. Hasil pengurutan tersebut digunakan untuk menentukan kombinasi fitur yang akan digunakan baik untuk proses *training* maupun *testing*.

Contoh hasil perhitungan nilai *F-Score* Tabel 3. Berdasarkan tabel *F-Score* yang sudah diurutkan tersebut dibuat kombinasi fitur seperti terlihat pada Tabel 4.3 untuk *F-Score*. . Dari Tabel 4.3 dapat dilihat bahwa kombinasi fitur model #1, dibuat dari fitur 1 (F_2), karena F_1 mempunyai nilai *F-Score* terbesar. Sedangkan kombinasi fitur model #2, dibuat dari F_1 dan F_{11} , karena F_2 dan F_{11} mempunyai nilai *F-Score* terbesar pertama dan kedua. Demikian seterusnya sehingga didapatkan 12 macam kombinasi fitur untuk *F-Score*.

Tabel 3: Nilai F-Score untuk masing-masing Fitur

No. Fitur	Fitur	F-Score
1	18	1.0000
2	22	1.0000
3	23	1.0000
4	5	0.0267
5	6	0.0261
6	1	0.0246
7	13	0.0236
8	14	0.0236
9	12	0.0217
10	8	0.0217
11	16	0.0197
12	9	0.0195
13	15	0.0164
14	10	0.0161
15	11	0.0161
16	2	0.0161
17	7	0.0161
18	17	0.0161
19	24	0.0161
20	25	0.0161
21	21	0.0157
22	19	0.0142
23	4	0.0027
24	3	0.0027
25	20	0.0010

3.4 Uji Coba Klasifikasi

Uji coba menggunakan SVM dengan penentuan kernel linear, Polynimial, dan RBF digunakan untuk membandingkan akurasi, sensitifitas dan spesifitas dan waktu komputasi yang terbaik menggunakan seleksi fitur maupun tanpa menggunakan seleksi fitur. Hasil yang didapat dapat dilihat pada tabel 4 dan 5.

Tabel 4: Hasil Klasifikasi terbaik tanpa menggunakan seleksi fitur

Klasifikasi	Akurasi (%)	Spesifitas (%)	Sensitivitas (%)	Waktu (detik)
SVM-linear	50	0	100	0.031
SVM-Polynimial	50	60	40	0.491
SVM-RBF	70	80	60	0.030

Tabel 5: Hasil Klasifikasi terbaik menggunakan seleksi fitur

Klasifikasi	Model Fitur	Akurasi	Spesifitas	Sensitivitas	Waktu (detik)
SVM-linear	rata-rata Semua fitur	50	0	100	0.010
SVM-Polynimial	rata-rata Semua fitur	50	0	100	0.010
SVM-RBF	23 dan 24	90	80	100	0.010

3.5 Pembahasan

Sesuai dengan tujuan penelitian ini yaitu menghasilkan sistem klasifikasi massa pada citra mammografi menggunakan kombinasi seleksi fitur, dan SVM, maka evaluasi dilakukan terhadap tingkat akurasi, waktu komputasi dan model kombinasi yang dihasilkan.

3.5.1 Tingkat Akurasi Klasifikasi

Berdasarkan hasil ujicoba yang dilakukan akurasi terbaik terletak pada klasifikasi SVM menggunakan kernel RBF dengan tingkat akurasi 90% dengan penambahan seleksi fitur, tanpa penambahan seleksi fitur SVM-RBF mendapatkan akurasi 70%. Pada tabel 4 dan 5 dapat disimpulkan bahwa sensitivitas hasil klasifikasi lebih kecil dari tingkat spesifitas. Perbedaan tingkat sensitivitas dan spesifitas ini terjadi karena persamaan nilai varian antara fitur hasil ekstraksi citra retina normal dengan citra retina exudate.

Dari hasil perbandingan diatas bahwa menggunakan seleksi fitur mampu meningkatkan akurasi klasifikasi dikarenakan tidak semua fitur digunakan. Namun, untuk kombinasi seleksi fitur pada klasifikasi SVM dengan penggunaan kernel RBF tingkat akurasi terbaik didapat pada kombinasi seleksi fitur model #23 #24.

3.5.2 Waktu Komputasi

Bentuk tabulasi data waktu yang dibutuhkan untuk proses klasifikasi (proses *training* dan *testing*) terhadap model kombinasi dari uji coba untuk F-Score dan tanpa seleksi fitur masing-masing dapat diketahui yaitu waktu komputasi dengan SVM membutuhkan waktu 0,031 detik (kernel linear), 0.491 detik (kernel polynomial), 0.30 detik (kernel RBF) detik. Setelah dilakukan penambahan seleksi fitur menggunakan F-Score fitur membutuhkan waktu 0.10 detik. Hal

tersebut dibuktikan bahwa seleksi fitur sangat berpengaruh terhadap waktu komputasi.

3.5.3 Model Kombinasi

Evaluasi model kombinasi ini bertujuan untuk menguji apakah model kombinasi dengan tingkat akurasi tertinggi tersebut merupakan kombinasi fitur yang tetap. Model kombinasi untuk klasifikasi SVM RBF #23 yaitu F18F22F23F5F6F1F13F14F12F8F16F9F15F10F11F2F7F17F24F25F21F19F4F3.

3.5.4 Hubungan Kernel dengan Tingkat Akurasi Klasifikasi

Hubungan kernel dengan tingkat akurasi klasifikasi sangat berpengaruh terhadap tingkat akurasi. Terbukti bahwa penggunaan kernel RBF mampu menghasilkan akurasi terbaik daripada penggunaan kernel linear dan Polynomial. Karena pemilihan kernel akan menentukan *feature space* dimana fungsi klasifikasi akan dicari. Selagi fungsi kernelnya lagimate, SVM akan beroperasi secara benar meskipun tidak tahu map apa yang digunakan untuk satu per satu data.

3.5.5 Hubungan Kernel dengan Waktu Komputasi

Hubungan kernel dengan waktu komputasi sangat berpengaruh. Itu terbukti bahwa waktu yang dihasilkan untuk klasifikasi masing-masing kernel mempunyai nilai waktu yang berbeda. Bisa dilihat pada tabel 4 dan 5 terbukti bahwa penggunaan kernel RBF waktu yang dibutuhkan lebih baik daripada kernel Linear dan Polynomial pada klasifikasi SVM dengan seleksi fitur maupun tanpa seleksi fitur.

3.5.6 Hubungan Jumlah Fitur dengan Tingkat Akurasi Klasifikasi

Hubungan jumlah fitur dengan tingkat akurasi pada klasifikasi SVM RBF dapat dilihat pada tabel 4, 5. Bahwa jumlah fitur berpengaruh terhadap tingkat akurasi yang dihasilkan. Semakin banyak fitur yang digunakan semakin tinggi tingkat akurasi yang dihasilkan, tetapi setelah mencapai model #23.

3.5.7 Hubungan Jumlah Fitur dengan Waktu Komputasi

Hubungan jumlah fitur dengan waktu komputasi pada klasifikasi SVM RBF diperoleh dari hasil uji coba klasifikasi dengan kombinasi seleksi fitur. Menunjukkan bahwa jumlah fitur berpengaruh terhadap waktu komputasi.

Semakin banyak jumlah fitur yang digunakan semakin naik waktu komputasi yang dibutuhkan.

4. KESIMPULAN

Berdasarkan hasil yang didapatkan dari penelitian ini, maka disimpulkan sebagai berikut ini:

1. Penggunaan fitur ciri dari ekstraksi fitur GLCM untuk input klasifikasi masih belum mencapai akurasi maksimal disebabkan karena hasil segmentasi citra fundus belum maksimal.
2. Penambahan metode kombinasi seleksi fitur, pemilihan kernel, serta penggunaan parameter terbukti berpengaruh pada tingkat akurasi dan penurunan waktu komputasi.
3. Klasifikasi SVM dengan seleksi fitur mampu meningkatkan akurasi dan menurunkan waktu komputasi daripada tanpa seleksi fitur dengan nilai akurasi 90% dan waktu komputasi 0.010 detik.
4. Diperlukan penggunaan metode lainnya untuk segmentasi citra fundus normal dan exudate agar akurasi segmentasi sempurna.
5. Diperlukan perbandingan menggunakan metode seleksi fitur selain f-score.

DAFTAR PUSTAKA

- [1] Selvati, dkk, "Automated Detection of Diabetic Retinopathy for Early Diagnosis using Feature Extraction and Support Vector Machine", 2012, India: Mepco Schelenk Engineering College.
- [2] Amin, M. A. & Yan, H., "High speed detection of retinal blood vessels in fundus image using phase congruency", 2011, Soft Computing, 15(6), pp. 1217-1230.
- [3] Siahaan, Rodeo Valentino. "Prevalensi Diabetic Retinopathy di RSUP H. Adam Malik", 2010.
- [4] Kemenkes, "InfoDATIN. Pusat Data dan Informasi Kementerian Kesehatan RI", 2014.
- [5] Mahendran Gandhi, Dr. R. Dhanasekaran, "Diagnosis of Diabetic Retinopathy Using Morphological Process and SVM Classifier", International conference on

- [6] C. Layadi, M. Fajar, Hasniati, I. A. Musdar," Analisis data pada jaringan sensor nirkabel menggunakan metode *support vector machine*", *STMIK KHARISMA Makassar, JURNAL RESISTOR Vol. 1 No 1 –April 2018 p-ISSN 2598-7542 (Print), e-ISSN 2598-9650 (Online)*

- [7] Wahyudi, dkk, "Sistem Deteksi *Diabetic Retinopathy* Menggunakan Support Vector Machine". *Jurnal Sistem Informasi Bisnis* 03. 2012.

- [8] Chen, Y. W., & Lin, C. J. "*Combining SVMs with various feature selection strategies*". Available from <http://www.csie.ntu.edu.tw/~cjlin/papers/features.pdf>.2005.

- [9] Albrechtsen, F, "Statistical Texture Measures Computed from Gray Level Cooccurrence Matrices". *Image Processing Laboratory Department of Informatics. University of Oslo*,hal 1-14.2008.

- [10] Vapnik, V. "*The nature of statistical learning theory*". New York: Springer-Verlag. 1995.