

IMPLEMENTASI CONVOLUTIONAL NEURAL NETWORK (CNN) UNTUK DETEKSI RETINOPATI DIABETIK

Stevanus Frangky Handono¹⁾, Fetty Tri Anggraeny²⁾, Basuki Rahmat³⁾

Email: ¹⁾handonofranky@gmail.com, ²⁾fettyanggraeny.if@upnjatim.ac.id,

³⁾basukirahmat.if@upnjatim.ac.id

^{1,2,3}Teknik Informatika, Fakultas Ilmu Komputer, UPN Veteran Jawa Timur

Abstrak

Retina adalah selapis tipis sel yang terletak pada bagian belakang bola mata yang memiliki fungsi untuk menangkap bayangan obyek yang dilihat mata dan meneruskannya ke syaraf penglihatan. Oleh karena itu retina berperan penting dalam proses penglihatan. Salah satu penyakit yang kerap menyerang retina adalah Retinopati Diabetik (RD). Penyakit ini disebabkan oleh adanya Diabetes Melitus pada tubuh dan biasanya ditandai dengan adanya kerusakan dan sumbatan pembuluh darah retina. Untuk mendeteksi penyakit tersebut biasanya perlu dilakukan pengamatan manual pada foto retina dan memerlukan waktu relatif lama untuk melakukannya. Hal tersebut dapat diatasi dengan memanfaatkan sistem cerdas seperti CNN yang mampu mengekstrak fitur dari sebuah citra dan menggunakannya untuk proses lebih lanjut seperti klasifikasi citra, deteksi objek, dsb. Penelitian ini menggunakan CNN untuk mengklasifikasi antara citra retina yang terindikasi RD dengan citra tanpa penyakit. Dari pengujian terhadap 90 citra, didapatkan akurasi sebesar 58%, spesifisitas sebesar 0,61, dan sensitivitas sebesar 0,44.

Kata kunci : Retinopati Diabetik, Klasifikasi Citra, CNN (Convolutional Neural Network)

1. PENDAHULUAN

Retina merupakan organ yang berfungsi untuk membentuk bayangan dari obyek yang dilihat oleh mata kemudian mengubahnya menjadi sinyal dan disalurkan menuju otak melalui saraf usi yang tepat. Menurut Community Eye Health (2003) penyakit retina telah menjadi faktor terbesar yang menyebabkan kebutaan pada anak – anak diseluruh dunia. Hal ini disebabkan karena banyaknya jenis penyakit yang menyerang retina dan kesulitan dalam mengidentifikasinya.

Retinopati diabetik (RD) adalah salah satu contoh penyakit yang terjadi pada retina yang ditandai dengan kerusakan dan sumbatan pembuluh-pembuluh darah retina (Dwi, et al., 2016). Menurut Khan dkk (2017), Retinopati diabetik merupakan faktor terbesar penyebab hilangnya penglihatan dalam periode lima puluh tahun terakhir. RD memiliki beberapa gejala seperti mikroaneurisma, kebocoran pembuluh darah, pembengkakan retina, pertumbuhan pembuluh darah baru yang abnormal, dan jaringan saraf yang rusak (Chakrabarty, 2018) Hal – hal itulah yang menjadi pembeda antara penyakit yang disebabkan oleh diabetes dengan penyebab lain. Namun, gejala - gejala dari RD tidak dapat dideteksi secara kasat mata sehingga memerlukan foto fundus. Proses identifikasi dari foto fundus tersebut memerlukan waktu relatif lama sehingga diperlukan sebuah sistem yang dapat membantu menganalisis dan mengklasifikasikan RD dengan penyakit lain.

Deep learning adalah salah satu cabang dari ilmu Machine Learning yang terdiri dari algoritma pemodelan abstraksi tingkat tinggi pada data menggunakan sekumpulan fungsi transformasi non-linear yang ditata berlapis-lapis dan mendalam. Pendekatan ini

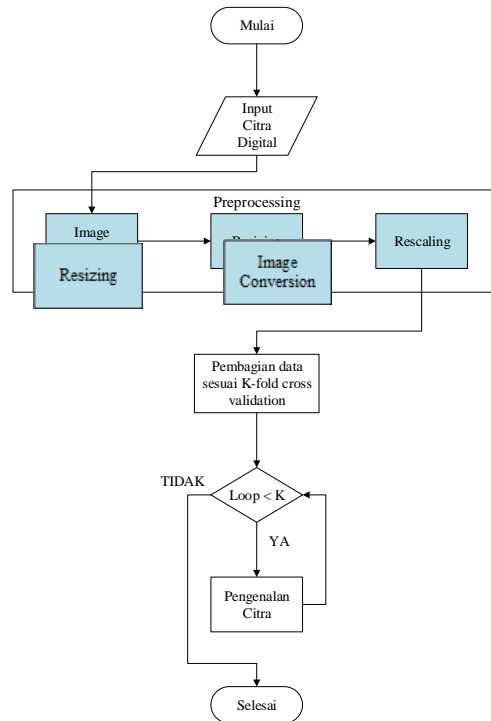
memiliki beberapa model, salah satunya adalah Convolutional Neural Network (CNN). Model ini memanfaatkan proses konvolusi dengan menggerakkan sebuah kernel konvolusi berukuran tertentu kepada sebuah Gambar. Dari hasil perkalian antara citra dengan filter itulah komputer akan mendapat informasi baru yang dapat digunakan untuk proses lebih lanjut seperti klasifikasi citra, deteksi obyek, dsb.

2. METODE PENELITIAN

2.1 Analisa Data

Data yang digunakan untuk penulisan ini menggunakan dataset sebagai data latih, data validasi, dan data uji. Dataset didapatkan dari MESSIDOR. Data citra yang digunakan dalam penelitian ini berjumlah 900 dengan format .tif. Dataset terbagi menjadi dua kelas yang meliputi 393 citra normal dan 507 citra terindikasi RD. Pembagian antara data latih dan data uji adalah sebagai berikut: 810 data latih dan 90 data uji yang terdiri 45 citra normal dan 45 citra terindikasi RD. Untuk evaluasi program, digunakan metode *K-fold cross validation*. Menurut Rodriguez dkk (2010), nilai K yang cocok untuk mengukur nilai *error* adalah $K = 5$ ataupun $K = 10$ karena memiliki nilai bias yang lebih kecil daripada $K = 2$ dan *cost computational* yang lebih rendah daripada $K = n$. Berdasarkan hal tersebut, data latih akan dibagi menjadi 5 bagian dengan rincian : empat bagian untuk pelatihan sistem dan satu bagian untuk menguji validasi sistem.

2.2 Analisa Sistem



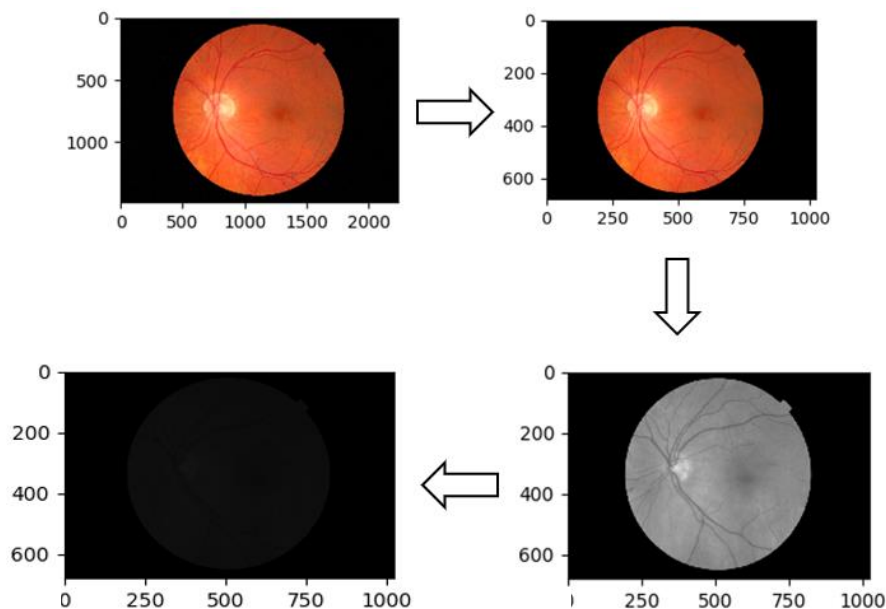
Gambar 1 Gambaran Umum Sistem

Alur kerja yang ditunjukkan pada Gambar 1. yaitu pengguna memasukkan data citra digital yang akan diidentifikasi, kemudian akan dilakukan tahap *preprocessing* yang terdiri

dari beberapa proses, selanjutnya citra yang siap diproses akan dibagi sesuai aturan *K-fold CV*. Pada penelitian ini, nilai *K* yang digunakan adalah 5. Setelah data terbagi menjadi data latih dan data validasi, data kemudian dimasukkan pada arsitektur CNN untuk dideteksi fitur serta diklasifikasikan. Hasil dari sistem berupa pernyataan bahwa citra terindikasi RD atau tidak.

2.3 Preprocessing Data

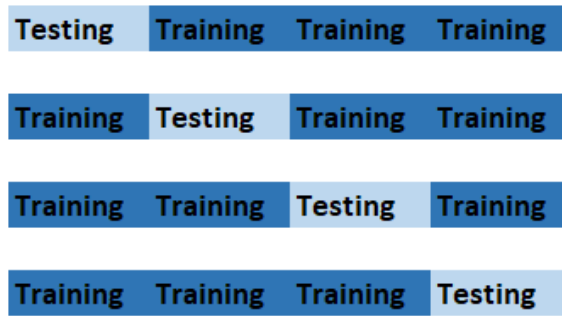
Tahap *preprocessing* terbagi kedalam tiga tahap yaitu: *resizing*, *image conversion*, dan *rescaling*. *Resizing* merupakan proses pengubahan ukuran panjang dan tinggi citra. Tahap ini digunakan untuk mempercepat komputasi serta standarisasi dataset. *Image conversion* merupakan proses konversi citra RGB ke *grayscale*. Tahap ini dapat menyederhanakan jumlah matriks pada gambar, bila citra RGB memiliki tiga matriks yang masing – masing mewakili satu *channel* warna, maka citra *grayscale* hanya memiliki satu matriks. Pada jenis gambar ini, warna dinyatakan dengan intensitas yang berkisar antara 0 sampai dengan 255. Nilai 0 menyatakan hitam dan nilai 255 menyatakan putih. (Nafi'iyah , 2015). *Rescaling* merupakan tahap pengubahan nilai piksel dari citra sehingga memiliki nilai antara nol dan satu. Proses ini dapat dilakukan dengan cara membagi tiap - tiap nilai piksel dengan besaran 255. Tahapan *preprocessing* dapat dilihat pada Gambar 2.



Gambar 2 Preprocessing citra

2.4 K-fold cross validation

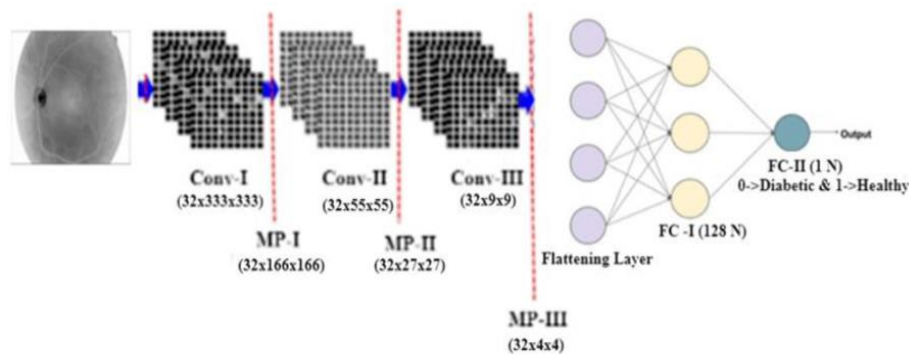
Menurut Yadav dan Shukla (2016), *Cross validation* merupakan sebuah prosedur untuk meng-generalisasi performa beberapa model yang berbeda. Pada *K-fold cross validation (K-fold CV)* data dibagi menjadi sejumlah nilai *K*, kemudian salah satu dari pecahan data tersebut dijadikan sebagai data uji sedangkan sisanya dijadikan data latih. Jumlah proses pelatihan dan testing akan dilakukan sesuai dengan nilai *K* yang telah ditentukan. Untuk mengetahui akurasi model, maka setiap hasil dari beberapakali proses pelatihan tersebut akan dijumlahkan dan dirata-rata. Skema dari *K-fold CV* dapat dilihat pada Gambar 3.



Gambar 3 Ketentuan *K-fold cross validation*

2.5 CNN (Convolutional Neural Network)

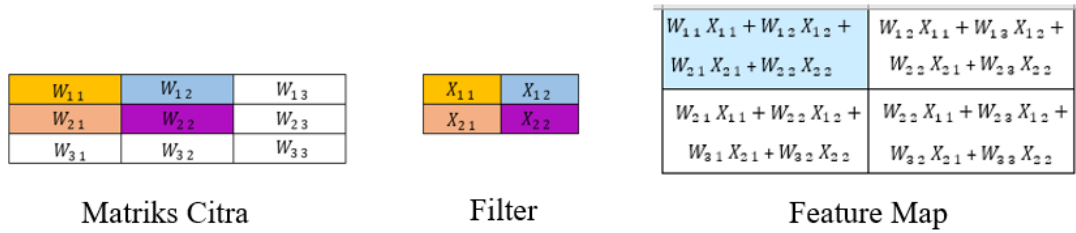
Convolutional Neural Network (CNN) merupakan variasi dari jaringan saraf tiruan yang memiliki bobot dan beberapa lapisan tersembunyi yang disusun menjadi arsitektur. (Limantoro, et al., 2018). Terdapat beberapa lapisan pada model CNN yaitu lapisan konvolusi, fungsi aktivasi, lapisan *pooling*, lapisan *flatten*, dan lapisan *fully conected*. Secara umum, skema dari CNN dapat dilihat pada Gambar 4.



Gambar 4 Skema dari CNN (Chakrabarty, 2018)

Lapisan pertama adalah lapisan konvolusi yang bekerja dengan cara melakukan perkalian antara matriks dari citra *input* dengan filter, kemudian menjumlahkan hasil dari perkalian tersebut dan menyimpannya kedalam matriks baru yang disebut *feature map*. Proses perkalian tersebut dilakukan secara konvolusi sesuai dengan ukuran dimensi citra *input*. Cara kerja lapisan konvolusi dapat dilihat pada Gambar 5 serta fungsi persamaan dari lapisan ini dapat dilihat pada Persamaan 1.

$$(f * g)(n) = \sum_{-\infty}^{\infty} f(m)g(n - m) \tag{1}$$

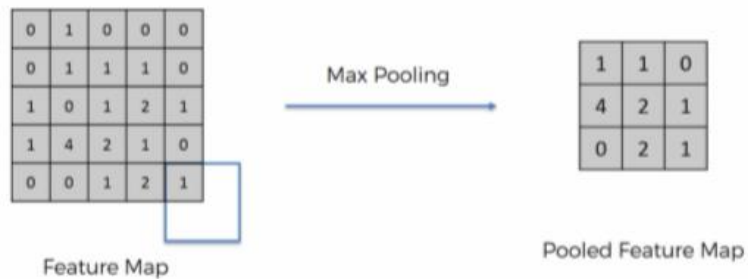


Gambar 5 Proses pada lapisan konvolusi

Fungsi aktivasi atau fungsi transfer merupakan fungsi non-linier yang memungkinkan sebuah jaringan untuk dapat menyelesaikan permasalahan permasalahan non trivial. Setiap fungsi aktivasi mengambil sebuah nilai dan melakukan operasi matematika (Zufar & Setiyono, 2016). Fungsi aktivasi yang digunakan pada penelitian ini adalah ReLU. Fungsi tersebut akan mengubah nilai negatif atau sama dengan nol menjadi nol dan akan memperahankan nilai diatas nol. Persamaan dari ReLU dapat dilihat pada Persamaan 2.

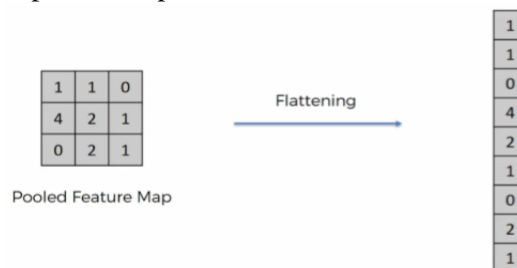
$$f_n = \max(0, n)$$

Lapisan *pooling* bekerja dengan blok spasial yang bergerak sepanjang ukuran *feature pattern*. Ukuran pergeseran blok pada umumnya adalah ukuran pada dimensi blok ($H \times H$) i (2) sendiri sehingga tidak ada *overlapping* seperti pada lapisan konvolusi. Lapisan *pooling* yang digunakan pada penelitian ini adalah *maxpool*. *Maxpool* akan mencari nilai terbesar dari blok spasial yang terdapat pada *feature map*. Proses pada lapisan ini dapat dilihat pada Gambar 6.



Gambar 6 Proses pada lapisan *maxpooling*

Lapisan selanjutnya adalah lapisan *flatten*. Lapisan ini akan mengubah bentuk dari *feature map* yang semula berbentuk matriks menjadi sebuah vector satu dimensi. Hal ini dilakukan agar *feature map* dapat diproses pada lapisan *fully connected*. Proses pengubahan matriks menjadi vektor dapat dilihat pada Gambar 7.



Gambar 7 Proses pada lapisan *flatten*

Lapisan *fully connected* terdiri dari banyak node yang saling terhubung dimana secara umum node – node tersebut akan dikelompokkan ke dalam beberapa lapisan yaitu *input layer*, *hidden layer*, dan *output layer*. Setiap terjadi perpindahan data antar node akan terjadi proses seperti pada persamaan 3.

$$Output = (input * bobot) + bias$$

2.5 Confusion matrix

Menurut Lewis dan Brown (2001), *confusion matrix* adalah metode yang cocok untuk pengklasifikasian dimana sebuah pixel pada lokasi tertentu, dapat diklasifikasikan kedalam sebuah kelas dan akurasi dapat diukur berdasarkan proporsi area yang berhasil diklasifikasi dan dihitung berdasarkan jumlah pixel yang berhasil diklasifikasikan. *Confusion matrix* dapat digunakan untuk mengukur performa dari sebuah *machine learning*. Terdapat empat istilah yang digunakan untuk merepresentasikan hasil proses klasifikasi, diantaranya adalah : *True Positive* (TP), *False Positive* (FP), *True Negative* (TN), dan *False Negative* (FN). Keempat istilah tersebut dapat di gambarkan pada Tabel 1.

Tabel 1 Istilah pada *confusion matrix*

		Nilai Sebenarnya	
		Sakit	Sehat
Prediksi	Sakit	TP	FP
	Sehat	FN	TN

Dengan menggunakan istilah – istilah tersebut, performa sistem seperti akurasi, spesifisitas, dan sensitivitas dapat diukur. Akurasi merupakan tingkat kecocokan antara prediksi sistem dan nilai sebenarnya. Tingkat akurasi dapat dihitung menggunakan Persamaan (4). Nilai *Recall* / Sensitivitas merupakan rasio nilai yang diklasifikasikan secara benar kedalam *positive* diantara seluruh kelas *positive*. *Recall* dapat dihitung menggunakan Persamaan (5). *Precision* / Spesifisitas digunakan untuk mengindikasikan bahwa data yang diklasifikasikan *positive* adalah benar - benar bernilai *positive*. Spesifisitas dapat dihitung dengan menggunakan Persamaan (6). *F-Measure* digunakan untuk pengukuran yang merepresentasikan sensitivitas dan spesifisitas. Hal ini berfungsi untuk membandingkan dua model dengan nilai sensitivitas dan spesifisitas yang berbeda. *F Measure* dapat ditentukan dengan menggunakan Persamaan (7)

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

$$Recall = \frac{TP}{(TP + FN)} \tag{5}$$

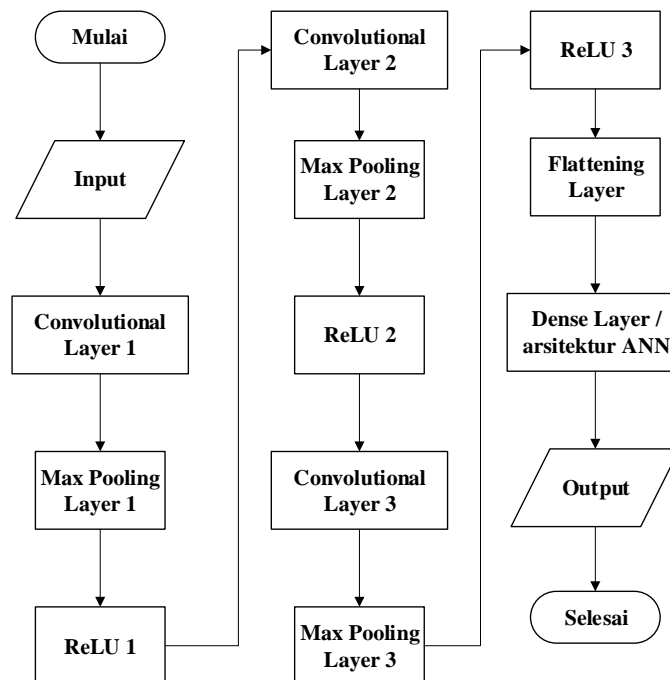
$$Precision = \frac{TP}{(TP + FP)} \tag{6}$$

$$F - measure = \frac{2 * Recall * Precision}{Recall + Precision} \tag{7}$$

2.6 Arsitektur dan Parameter CNN

Arsitektur yang digunakan pada penelitian ini berjumlah tiga set kombinasi konvolusi yang masing – masing set akan berisikan lapisan konvolusi, ReLU, dan lapisan *pooling*. Lapisan konvolusi bertanggung jawab untuk membuat *convolved feature maps* yang mewakili masing – masing fitur detektor dalam lapisan tersebut. *Feature maps* yang telah didapat dari lapisan konvolusi kemudian akan diaktivasi menggunakan ReLU. Lapisan terakhir dari satu set kombinasi konvolusi adalah lapisan *pooling*. Lapisan tersebut berguna untuk mempertahankan fitur – fitur yang tersedia dan juga agar CNN memiliki independensi spasial.

Feature maps yang didapat setelah melalui tiga set kombinasi konvolusi akan diubah menjadi sebuah vektor fitur tunggal melalui lapisan *flatten*. Vektor fitur tersebut berfungsi sebagai *input* pada lapisan *fully conected* yang merupakan sebuah arsitektur Artificial Neural Network. Pada lapisan *fully conected*, dari *input layer* menuju *hidden layer* akan diaktivasi menggunakan ReLU. Data yang diteruskan dari *hidden layer* menuju *output layer* akan diaktivasi menggunakan sigmoid. Hasil dari *output layer* berupa pernyataan bahwa citra merupakan retina normal atau mengidap Retinopati Diabetik. Tahapan – tahapan dalam arsitektur CNN dapat dilihat pada Gambar 8



Gambar 8 Arsitektur CNN

Parameter yang digunakan dalam arsitektur penelitian ini dapat dilihat pada Tabel 2. Pada setiap lapisan konvolusi akan terdapat 32 filter dimana tiap – tiap filter berukuran 3×3 . Tiap lapisan *pooling* memiliki kernel berukuran 2×2 . Pada lapisan *fully conected*, *hidden layer* memiliki node sebanyak 128 dan *output layer* memiliki node sebanyak 1.

Tabel 2 Parameter pada arsitektur CNN

<i>Layer</i>	<i>Parameter</i>
Lapisan Konvolusi 1	32 filter (3×3), stride (1)
Lapisan Pooling 1	(2×2), stride (1)
Lapisan Konvolusi 2	32 filter (3×3), stride (1)
Lapisan Pooling 2	(2×2), stride (1)
Lapisan Konvolusi 3	32 filter (3×3), stride (1)
Lapisan Pooling 3	(2×2), stride (1)
Jumlah node pada fully connected	<i>hidden layer</i> (128), <i>output layer</i> (1)

3. HASIL DAN PEMBAHASAN

Pada penelitian ini, dilakukan lima kali pengujian sesuai ketentuan *K-fold cross validation*. Dalam setiap pengujian, data akan diproses sebanyak 160 epoch seperti pada Gambar 9. Hasil pengujian dapat dilihat pada Tabel 4.

epoch	acc	loss	lr	val_acc	val_loss
0	0.52597404	1.0211785520826067	1e-05	0.53125	0.9223155379295349
1	0.52597404	0.9144100059162487	1e-05	0.59375	0.867500901222229
2	0.53733766	0.8439636214987024	1e-05	0.5625	0.8544986248016357
3	0.53896105	0.8495361371473833	1e-05	0.59375	0.8408467769622803
4	0.5422078	0.8133599866520275	1e-05	0.5625	0.8341730833053589
5	0.53409094	0.8280062923183689	1e-05	0.59375	0.8161487579345703
155	0.7305195	0.5554718266833912	1e-05	0.53125	0.7074968814849854
156	0.6948052	0.5736294303621564	1e-05	0.625	0.7264565825462341
157	0.7078125	0.5806189954280854	1e-05	0.59375	0.7072066068649292
158	0.7224026	0.551620299165899	1e-05	0.59375	0.7208466529846191
159	0.6925676	0.5824123129651353	1e-05	0.53125	0.704132080078125
160	0.70941556	0.5618691583732506	1e-05	0.53125	0.6982476115226746

Gambar 9 Proses Training

Tabel 4 Hasil pengujian

Fold	True Positive	False Negative	True Negative	False Positive
1	10	35	26	19
2	23	22	33	12
3	22	23	21	24
4	25	20	39	6
5	20	25	42	3

Berdasarkan pengujian yang telah dilakukan, fold 1 memiliki akurasi sebesar 40%, sensitivitas sebesar 0,22 dan spesifisitas sebesar 0,35. Pada fold 2 didapatkan akurasi sebesar 62% dengan sensitivitas 0,51 dan spesifisitas 0,66. Fold 3 memiliki akurasi sebesar 48% dengan nilai sensitivitas sebesar 0,49 dan spesifisitas 0,48. Dari percobaan fold ke-empat didapatkan hasil berupa akurasi sebesar 71% dengan sensitivitas senilai 0,56 dan spesifisitas senilai 0,81. Fold terakhir memiliki tingkat akurasi 69% dengan sensitivitas bernilai 0,44 dan spesifisitas bernilai 0,87. Dari data – data tersebut dapat disimpulkan bahwa sistem memiliki tingkat akurasi sebesar 57,6% dengan sensitivitas sebesar 0,44 dan spesifisitas sebesar 0,63.

4. KESIMPULAN

Berdasarkan hasil penelitian, dapat diambil beberapa kesimpulan di antaranya :

1. Arsitektur dan parameter kurang sesuai untuk mendeteksi Retinopati Diabetik karena memiliki rata – rata akurasi hasil pengujian sebesar 57,6% .
2. Hasil evaluasi dengan confusion matriks pada data uji citra sebanyak 90 citra dan jumlah epoch 160 menunjukkan bahwa sistem memiliki nilai sensitivitas sebesar 0,44 dan nilai spesifisitas sebesar 0,63.

DAFTAR PUSTAKA

- [1] Chakrabarty, N., 2018. A Deep Learning Method for the detection of Diabetic Retinopathy.
- [2] Community Eye Health, 2003. Retinal Diseases and VISION 2020. pp. 19-20.
- [3] DECENCIÈRE, E. et al., 2014. FEEDBACK ON A PUBLICLY DISTRIBUTED IMAGE DATABASE: THE MESSIDOR DATABASE 2. *Image Anal Stereol 2014*, Volume 33, pp. 231-234.
- [4] Dwi, A. C., Zuhidya, L. & Wicaksono, A., 2016. GAMBARAN KARAKTERISTIK RETINOPATI DIABETIKA DI RSUD DR. SOEDARSO PONTIANAK.

- [5] Khan , F. A. & Agrawal, V. L., 2017. Classification of Diabetic Retinopathy using Multilayer Perceptron Neural Network. *Special Issue of International Journal of Electronics, Communication & Soft Computing Science and Engineering*.
- [6] Lewis, H. G. & Brown, M., 2001. A generalized confusion matrix for assessing area estimates from remotely sensed data. *International Journal of Remote Sensing*, Volume 22, pp. 3223-3235.
- [7] Limantoro, S. E., Kristian, Y. & Purwanto, D. D., 2018. Pemanfaatan Deep Learning pada Video Dash Cam untuk Deteksi Pengendara Sepeda Motor. *JNTETI*, pp. 167-173.
- [8] Nafi'iyah , N., 2015. Algoritma Kohonen dalam Mengubah Citra Graylevel Menjadi Citra Biner. *Jurnal Ilmiah Teknologi dan Informasia ASIA*, pp. 49-55.
- [9] Rodriguez, D. J., Perez, A. & Lozano, A. J., 2010. Sensitivity Analysis of k-Fold Cross Validation in Prediction Error Estimation. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, Volume 32, pp. 569 - 575.
- [10] Yadav, S. & Shukla, S., 2016. Analysis of k-fold cross-validation over hold-out validation on colossal datasets for quality classification. *IEEE 6th International Advanced Computing Conference*, pp. 78-83.
- [11] Zufar, M. & Setiyono, B., 2016. Convolutional Neural Networks untuk Pengenalan Wajah Secara Real-Time. *JURNAL SAINS DAN SENI ITS* , pp. 72-77.