

PENERAPAN MODEL HIBRIDA CNN-KNN UNTUK KLASIFIKASI PENYAKIT MATA

Adil Sandy Wardhani, Fetty Tri Anggraeny*, Agung Mustika Rizki

Informatika, Universitas Pembangunan Nasional “Veteran” Jawa Timur

Jalan Raya Rungkut Madya, Gunung Anyar, Surabaya, Jawa Timur, Indonesia, kode pos: 60294

fettyanggraeny.if@upnjatim.ac.id

ABSTRAK

Penyakit mata merupakan gangguan yang menyerang organ mata akibat dari virus, bakteri, dan kebiasaan buruk. Saat ini, penggunaan teknologi kecerdasan buatan populer dalam mendiagnosa penyakit mata untuk memungkinkan penanganan lebih dini agar tidak memicu kebutaan. *Convolutional Neural Network* (CNN) merupakan algoritma klasifikasi yang paling umum digunakan karena dapat menghasilkan akurasi yang baik dalam memproses data yang berformat gambar. *K-Nearest Neighbor* (KNN) juga termasuk algoritma untuk klasifikasi dengan menggunakan parameter nilai tetangga terdekat. Pada penelitian ini, peneliti akan melakukan hibrida atau menggabungkan algoritma CNN dan KNN dengan CNN sebagai proses ekstraksi fitur serta KNN sebagai klasifikasi. Penelitian akan dilakukan dengan menggunakan beberapa parameter pada CNN dan KNN untuk mencari akurasi terbaik. Hasil akurasi terbaik dari penerapan model hibrida CNN-KNN pada penyakit mata diperoleh dengan menggunakan optimasi adam *learning rate* 0,001 dan nilai tetangga terdekat 9 dengan akurasi sebesar 94,03%.

Kata kunci : penyakit mata, CNN, KNN, Hibrida CNN-KNN

1. PENDAHULUAN

Manusia mempunyai lima organ yang biasa disebut dengan panca indra. Indra merupakan alat dalam tubuh manusia yang berfungsi untuk mengenali, dan merasakan sesuatu mengenai lingkungan di sekitarnya. Mata merupakan salah satu dari panca indra yang berguna untuk mengenali lingkungan disekitarnya dengan cara melihat. Mata memiliki peran utama dalam proses penglihatan dan pengenalan visual, yang memungkinkan manusia untuk memperoleh sekitar 80% informasi hanya dengan menggunakan indera penglihatan [1].

Dalam era modern yang dipenuhi dengan teknologi digital saat ini, penggunaan mata untuk berinteraksi dengan perangkat digital menjadi semakin umum baik dari anak-anak hingga orang dewasa. Hampir semua orang menggunakan perangkat digital sebagai alat bantu dalam kegiatan sehari-hari. Namun, tanpa menyadarinya, penggunaan yang berlebihan dapat menimbulkan masalah kesehatan pada mata [2].

Masalah kesehatan yang mempengaruhi mata dapat menyebabkan ketidaknyamanan, gangguan penglihatan, atau bahkan hilangnya penglihatan pada mata. Faktor-faktor yang menyebabkan gangguan pada kesehatan mata yaitu kurangnya perhatian masyarakat terhadap kesehatan mata, kemalasan seseorang untuk melakukan pemeriksaan kesehatan mata di rumah sakit, dan penggunaan teknologi visual yang semakin canggih yang tidak bisa dikontrol oleh manusia yang mendorong penggunaan indra penglihatan secara terus menerus digunakan [3].

Menurut data yang diperoleh dari Survei Kebutaan Rapid Assessment of Avoidable Blindness (RAAB) periode 2014-2016 yang dilakukan oleh Kementerian Kesehatan, terungkap bahwa angka kebutaan pada populasi manusia yang berusia 50 tahun ke atas mencapai 3%. Katarak adalah penyebab utama

kebutaan dalam kelompok ini karena menyumbang sebanyak 81% dari kasus tersebut. Selain katarak, penyebab lain dari kebutaan meliputi masalah refraksi, glaukoma, dan kelainan mata yang berkaitan dengan diabetes [4].

Dengan mengenali dan merawat masalah pada mata secara dini, kita dapat mencegah kemungkinan berkembangnya penyakit lebih lanjut dan mengurangi risiko kehilangan penglihatan pada mata. Mendeteksi penyakit mata secara dini dapat dilakukan dengan pembuatan sistem kecerdasan buatan yang saat ini sedang populer digunakan. Kecerdasan buatan tersebut biasa disebut dengan pengolahan citra *image classification*.

Terdapat banyak metode yang dapat digunakan untuk proses klasifikasi, salah satunya adalah *Convolutional Neural Network* atau CNN. CNN memiliki salah satu kelebihan yaitu kemampuannya dalam mengklasifikasi gambar dapat menghasilkan tingkat akurasi yang tinggi karena mampu mengurangi jumlah parameter yang tidak terbatas dan dapat menangani variasi deformasi pada gambar input seperti translasi, rotasi, dan skala [5].

K-Nearest Neighbor atau KNN juga termasuk metode yang dapat digunakan untuk klasifikasi. KNN memiliki kelebihan dalam pelatihan yang sangat cepat dan kemampuannya untuk menangani variasi data yang besar [6]. Penelitian menggunakan KNN sebelumnya pernah dilakukan oleh Taftyani Yusuf untuk klasifikasi mutu jambu biji yang memperoleh akurasi terbaik sebesar 91,25% [7].

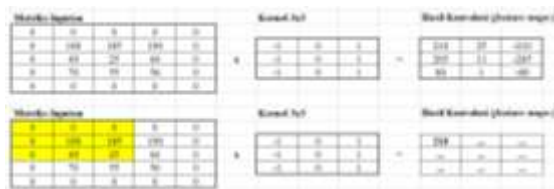
Berdasarkan uraian diatas, maka penulis dalam penelitian ini akan membuat sistem untuk klasifikasi penyakit mata dengan hibrida CNN-KNN. Hibrida sendiri adalah penggabungan antara dua model untuk memungkinkan menghasilkan akurasi yang lebih baik.

2. TINJAUAN PUSTAKA

2.1. Convolutional Neural Network

Convolutional Neural Network (CNN) adalah salah satu metode *deep learning* yang difokuskan pada analisis citra digital dan biasa diterapkan untuk mendeteksi objek. CNN merupakan evolusi dari *Multilayer Perceptron* (MLP) yang disesuaikan untuk memproses data dalam bentuk dua dimensi, seperti gambar atau suara. CNN tergolong dalam kategori *Deep Neural Network* karena strukturnya yang kompleks dan sering kali digunakan dalam analisis citra [8]. Pada CNN terdapat beberapa lapisan yaitu lapisan konvolusi, fungsi aktivasi, lapisan *pooling*, lapisan *flatten*, dan lapisan *fully connected*.

Lapisan konvolusi merupakan struktur yang terdiri dari berbagai filter konvolusi atau kernel yang digunakan untuk menghasilkan peta fitur keluaran dengan cara melakukan perkalian antara matriks dari citra *input* dengan filter. Operasi konvolusi dapat dilihat pada Gambar 1.



Gambar 1. Operasi Konvolusi

Fungsi aktivasi adalah jenis fungsi non-linear yang memungkinkan jaringan untuk menyelesaikan masalah-masalah yang tidak trivial dengan mengambil input dan menjalankan operasi matematika [9]. Operasi fungsi aktivasi dapat dilihat pada Gambar 2.



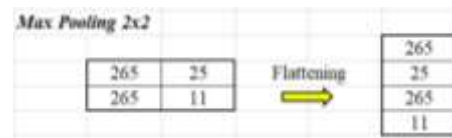
Gambar 2. Operasi Fungsi Aktivasi (ReLU)

Lapisan *pooling* digunakan untuk mereduksi dimensi citra untuk mempercepat proses komputasi dengan mengambil setiap nilai terbesar pada blok atau rata-rata. Operasi lapisan *pooling* dapat dilihat pada Gambar 3.



Gambar 3. Operasi Lapisan *Pooling* (Max Pooling)

Lapisan *Flatten* digunakan untuk mengubah struktur *feature map* dari array multidimensional menjadi vektor tunggal supaya dapat digunakan sebagai input untuk lapisan *fully connected* [10]. Operasi lapisan *flatten* dapat dilihat pada Gambar 4.



Gambar 4. Operasi *Flatten*

Lapisan *fully connected* terdiri dari sejumlah besar node yang terhubung satu sama lain, biasanya dikelompokkan menjadi beberapa lapisan, termasuk *input layer*, *hidden layer*, dan *output layer* yang bertujuan untuk mengolah data agar dapat diklasifikasikan.

2.2. K-Nearest Neighbor

K-Nearest Neighbor (KNN) merupakan algoritma yang termasuk *supervised learning*, yang bertujuan untuk mengidentifikasi pola baru dalam data dengan membandingkan pola yang telah ada sebelumnya. Saat melakukan klasifikasi, algoritma ini tidak memerlukan pembentukan model dan hanya bergantung pada pengingatan memori [11].

KNN adalah teknik untuk mengklasifikasikan data uji berdasarkan data latih dengan menggunakan nilai jarak terdekat. Metode yang digunakan untuk mengukur jarak adalah menggunakan Jarak Euclidean, yang merupakan metode yang umum digunakan untuk data numerik. Metode ini termasuk dalam klasifikasi non-parametrik dan secara komputasi lebih sederhana dibandingkan dengan pendekatan lainnya [12]. Adapun langkah-langkah dari algoritma KNN yaitu :

1. Menetapkan nilai K sebagai parameter
2. Mengukur jarak antara data uji dan data latih
3. Jarak tersebut kemudian diurutkan secara menurun
4. Memilih jarak terdekat
5. Memilih kelas dengan jumlah data terbanyak dan melakukan klasifikasi

2.3. Adam dan RMSprop

Adaptive Moment Estimation (Adam) adalah algoritma yang bisa menggantikan langkah-langkah *stochastic gradient descent* konvensional untuk memperbarui bobot jaringan berdasarkan data latih secara berulang. Cara kerja Adam dapat dijelaskan sebagai penggabungan aspek terbaik dari dua variasi *stochastic gradient descent*, yaitu *adaptive gradient algorithm* dan *root mean square propagation*. Dengan penggabungan ini, Adam mampu memberikan peningkatan pada algoritma yang dapat menangani gradien yang jarang pada masalah yang berisik [13].

RMSprop adalah sebuah metode optimisasi yang menggunakan besaran gradien terbaru untuk menormalkan gradien, dan ini memungkinkannya untuk mempertahankan rata-rata bergerak di atas nilai *root mean square* dari gradien. RMSprop dikenal sebagai salah satu metode optimisasi yang memelihara rata-rata dari kuadrat gradien untuk setiap bobot [14].

2.4. Confusion Matrix

Confusion Matrix adalah sebuah tabel yang menampilkan jumlah data uji yang diklasifikasikan dengan nilai yang benar dan salah. Contoh *confusion matrix* untuk klasifikasi dapat dilihat pada Tabel 1.

Tabel 1. *Confusion Matrix*

		Kelas Prediksi	
		1	0
Kelas Sebenarnya	1	TP	FN
	0	FP	TN

Keterangan :

TP (*True Positive*) = jumlah dokumen yang sebenarnya berasal dari kelas 1 dan berhasil diklasifikasikan sebagai kelas 1

TN (*True Negative*) = jumlah dokumen yang sebenarnya berasal dari kelas 0 dan berhasil diklasifikasikan sebagai kelas 0

FP (*False Positive*) = jumlah dokumen yang sebenarnya berasal dari kelas 0 namun salah diklasifikasikan sebagai kelas 1

FN (*False Negative*) = jumlah dokumen yang sebenarnya berasal dari kelas 1 namun salah diklasifikasikan sebagai kelas 0

Rumus *confusion matrix* untuk menghitung *accuracy*, *precision*, *recall*, dan *f1 score* adalah sebagai berikut:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (1)$$

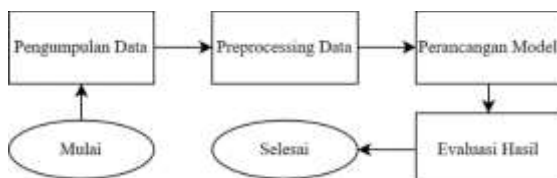
$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F1 \text{ Score} = \frac{2 \times (Recall + Presisi)}{(Recall + Presisi)} \quad (4)$$

3. METODE PENELITIAN

Dalam penelitian ini, dilakukan penjabaran mengenai langkah-langkah yang akan diambil dalam jalannya penelitian. Hal ini bertujuan untuk memastikan kelancaran proses penelitian dan mencapai tujuan yang diinginkan. Tahapan yang akan dilakukan dapat dilihat pada Gambar 5.

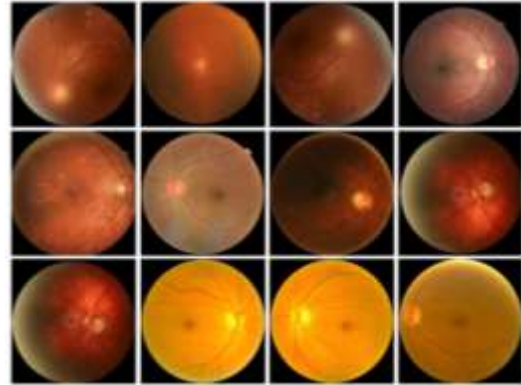


Gambar 5. Alur Penelitian

3.1. Pengumpulan Data

Pengumpulan data dilakukan dengan mencari data citra penyakit mata. Data penyakit mata yang digunakan dalam penelitian ini merupakan data sekunder yang diperoleh dari *website* Kaggle. Dataset ini terdiri dari 4183 gambar dengan format file JPG dan PNG. Data tersebut memuat citra mata jenis

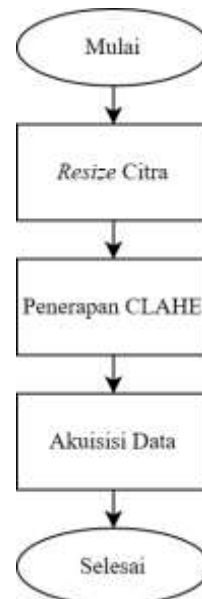
fundus yang terbagi ke dalam 4 kategori, yakni *cataract* (1006 gambar), *diabetic retinopathy* (1098 gambar), *glaucoma* (1007 gambar), dan mata normal (1072 gambar). Setiap data memiliki tiga saluran warna RGB dan dimensi gambar yang beragam. Citra mata tersebut dapat dilihat pada Gambar 6.



Gambar 6. Sample Citra Mata

3.2. Preprocessing Data

Preprocessing data dilakukan dengan maksud untuk memproses data gambar sehingga dapat dibaca secara optimal oleh model yang akan dimanfaatkan nantinya. Alur *preprocessing* data pada penelitian ini dapat dilihat pada Gambar 7.



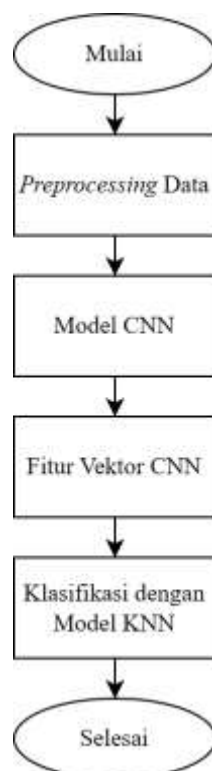
Gambar 7. Alur Preprocessing Data

Preprocessing data dimulai dari *resize* data yaitu mengubah semua data gambar yang sebelumnya memiliki ukuran gambar yang beragam menjadi sama yakni 224x224 *pixel*. Setelah itu dilakukan penerapan CLAHE pada gambar yang berfungsi untuk menambah kontras dan ketajaman gambar agar lebih baik, namun pada gambar masih mempertahankan warna RGB. Kemudian yang terakhir melakukan akuisisi data untuk membagi data menjadi data *training*, data *validation*, dan data *testing*. Pada

penelitian ini, keseluruhan data berjumlah 4183 citra, kemudian dilakukan pembagian data dengan rasio 80:10:10 untuk masing-masing data *training*, data *validation*, dan data *testing*.

3.3. Perancangan Model

Perancangan model yang dibangun pada penelitian ini adalah hibrida CNN-KNN. Hibrida bermaksud untuk menggabungkan kedua model CNN dan KNN dengan CNN yang menghasilkan fitur vektor kemudian proses klasifikasi menggunakan KNN. Alur perancangan model dapat dilihat pada Gambar 8.



Gambar 8. Alur Perancangan Model

Perancangan model dimulai setelah preprocessing data telah selesai dilakukan. Model CNN merupakan model awal yang akan dibangun. Model CNN disini menggunakan arsitektur MobileNetV2 yang memiliki parameter sedikit namun dapat memberikan akurasi yang baik serta penggunaan berbagai parameter optimasi dan *learning rate* untuk mencari nilai akurasi terbaik. Fitur vektor didapatkan dari proses ekstraksi fitur pada model CNN. Fitur vektor tersebut kemudian digunakan untuk bahan klasifikasi yang akan dilakukan oleh model KNN. Klasifikasi yang dilakukan menggunakan parameter nilai tetangga terdekat 5, 7, dan 9 untuk mencari akurasi yang terbaik.

3.4. Evaluasi Model

Langkah terakhir dalam penelitian ini adalah melakukan evaluasi terhadap hasil pelatihan model yang diperoleh. Tujuan dari evaluasi ini adalah untuk

menentukan seberapa akurat sistem yang telah dibangun menggunakan data *testing* untuk klasifikasi. Evaluasi dilakukan dengan menampilkan *classification report* dan menampilkan tingkat akurasi, ketepatan (*precision*), kecocokan (*recall*), dan *f1 score*.

4. HASIL DAN PEMBAHASAN

Pembahasan mengenai hasil penelitian ini mencakup tingkat keakuratan yang tercapai saat melakukan klasifikasi penyakit mata menggunakan model hibrida CNN-KNN. Pada model CNN, menggunakan beberapa parameter seperti optimasi adam dan rmsprop serta *learning rate* 0,001 dan 0,001. Sementara pada model KNN, menggunakan parameter nilai tetangga terdekat 5, 7, dan 9.

Untuk hasil akurasi dari hibrida CNN-KNN dengan optimasi adam dan *learning rate* 0,001 dapat dilihat pada Tabel 2.

Tabel 2. Optimasi Adam dan *Learning Rate* 0,001

Model 1	Optimasi	LR	Model 2	Nilai K	Acc
CNN	Adam	0,001	KNN		92,6%
				K = 5	91,41%
				K = 7	93,32%
				K = 9	94,03%

Pada Tabel 2 terlihat hasil akurasi yang diperoleh dari pengujian CNN-KNN dengan menggunakan optimasi adam dan *learning rate* 0,001, serta beberapa parameter nilai tetangga terdekat yang diterapkan pada model KNN. Hasil klasifikasi terbaik pada CNN-KNN diperoleh dengan penggunaan parameter nilai K = 9 dengan akurasi sebesar 94%.

Selain itu, dari Tabel 2 juga terlihat bahwa model hibrida CNN-SVM mampu meningkatkan tingkat akurasi jika dibandingkan dengan penggunaan model CNN murni, yang hanya mencapai akurasi 92,6%. Untuk detail *classification report* dari klasifikasi terbaik dapat dilihat pada Gambar 9.

Classification Report :				
	precision	recall	f1-score	support
0	0.95	0.97	0.96	100
1	0.98	0.98	0.98	110
2	0.98	0.89	0.90	102
3	0.92	0.92	0.92	107
accuracy			0.94	419
macro avg	0.94	0.94	0.94	419
weighted avg	0.94	0.94	0.94	419
Validation Accuracy CNN-KNN dengan K=9 : 0.916267942583732				
Test Accuracy CNN-KNN dengan K=9 : 0.9403341288782816				

Gambar 9. *Classification Report* Hibrida CNN-KNN (Optimasi Adam, LR 0,001 dan K = 9)

Untuk hasil akurasi dari hibrida CNN-KNN dengan optimasi adam dan *learning rate* 0,0001 dapat dilihat pada Tabel 3.

Tabel 3. Optimasi Adam dan *Learning Rate* 0,0001

Model 1	Optimasi	LR	Model 2	Nilai K	Acc
CNN	Adam	0,0001	KNN		90,45%
				K = 5	89,74%
				K = 7	90,69%
				K = 9	91,41%

Pada Tabel 3 terlihat hasil akurasi yang diperoleh dari pengujian CNN-KNN dengan menggunakan optimasi Adam dan *learning rate* 0,0001, serta beberapa parameter nilai tetangga terdekat yang diterapkan pada model KNN. Hasil klasifikasi terbaik pada CNN-KNN diperoleh dengan penggunaan parameter nilai K = 9 dengan akurasi sebesar 91,41%.

Selain itu, dari Tabel 3 juga terlihat bahwa model hibrida CNN-KNN mampu meningkatkan tingkat akurasi jika dibandingkan dengan penggunaan model CNN murni, yang hanya mencapai akurasi 90,45%. Untuk detail *classification report* dari klasifikasi terbaik dapat dilihat pada Gambar 10.

Classification Report :

	precision	recall	f1-score	support
0	0.91	0.96	0.93	100
1	0.99	0.96	0.98	110
2	0.88	0.81	0.85	102
3	0.88	0.92	0.89	107
accuracy			0.91	419
macro avg	0.91	0.91	0.91	419
weighted avg	0.91	0.91	0.91	419

Validation Accuracy CNN-KNN dengan K=9 : 0.8923444976076556
Test Accuracy CNN-KNN dengan K=9 : 0.9140811455847255

Gambar 10. *Classification Report* Hibrida CNN-KNN (Optimasi Adam, LR 0,0001, dan K = 9)

Untuk hasil akurasi dari hibrida CNN-KNN dengan optimasi rmsprop dan *learning rate* 0,001 dapat dilihat pada Tabel 4.

Tabel 4. Optimasi RMSprop dan *Learning Rate* 0,001

Model 1	Optimasi	LR	Model 2	Nilai K	Acc
CNN	RMSprop	0,001	KNN		90,69%
				K = 5	90,93%
				K = 7	92,36%
				K = 9	92,36%

Pada Tabel 4 terlihat hasil akurasi yang diperoleh dari pengujian CNN-KNN dengan menggunakan optimasi rmsprop dan *learning rate* 0,001, serta beberapa parameter nilai tetangga terdekat yang diterapkan pada model KNN. Hasil klasifikasi terbaik pada CNN-KNN diperoleh dengan penggunaan parameter nilai K = 7 dan 9 dengan akurasi sebesar 92,36%.

Selain itu, dari Tabel 4 juga terlihat bahwa model hibrida CNN-KNN mampu meningkatkan tingkat akurasi jika dibandingkan dengan penggunaan model CNN murni, yang hanya mencapai akurasi 90,69%. Untuk detail *classification report* dari klasifikasi terbaik dapat dilihat pada Gambar 11.

Classification Report :

	precision	recall	f1-score	support
0	0.94	0.95	0.95	100
1	0.98	0.98	0.98	110
2	0.86	0.88	0.87	102
3	0.91	0.88	0.90	107
accuracy			0.92	419
macro avg	0.92	0.92	0.92	419
weighted avg	0.92	0.92	0.92	419

Validation Accuracy CNN-KNN dengan K=9 : 0.9234449760765551
Test Accuracy CNN-KNN dengan K=9 : 0.9236276849642004

Gambar 11. *Classification Report* Hibrida CNN-KNN (Optimasi RMSprop, LR 0,001 dan K = 9)

Untuk hasil akurasi dari hibrida CNN-KNN dengan optimasi rmsprop dan *learning rate* 0,0001 dapat dilihat pada Tabel 5.

Tabel 5. Optimasi RMSprop dan *Learning Rate* 0,0001

Model 1	Optimasi	LR	Model 2	Nilai K	Acc
CNN	RMSprop	0,0001	KNN		91,64%
				K = 5	89,74%
				K = 7	90,93%
				K = 9	90,93%

Pada Tabel 5 terlihat hasil akurasi yang diperoleh dari pengujian CNN-KNN dengan menggunakan optimasi rmsprop dan *learning rate* 0,0001, serta beberapa parameter nilai tetangga terdekat yang diterapkan pada model KNN. Hasil klasifikasi terbaik pada CNN-KNN diperoleh dengan penggunaan parameter nilai K = 7 dan 9 dengan akurasi sebesar 90,93%.

Selain itu, dari Tabel 5 juga terlihat bahwa model hibrida CNN-KNN tidak mampu meningkatkan akurasi jika dibandingkan dengan penggunaan model CNN murni, yang mencapai akurasi sebesar 91,64%. Untuk detail *classification report* dari klasifikasi hibrida terbaik dapat dilihat pada Gambar 12.

Classification Report :

	precision	recall	f1-score	support
0	0.92	0.96	0.94	100
1	0.97	0.96	0.97	110
2	0.85	0.87	0.86	102
3	0.89	0.84	0.87	107
accuracy			0.91	419
macro avg	0.91	0.91	0.91	419
weighted avg	0.91	0.91	0.91	419

Validation Accuracy CNN-KNN dengan K=9 : 0.911483253585168
Test Accuracy CNN-KNN dengan K=9 : 0.9093078758949881

Gambar 12. *Classification Report* Hibrida CNN-KNN (Optimasi RMSprop, LR 0,0001 dan K = 9)

5. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian dapat disimpulkan bahwa penggunaan mode hibrida CNN-KNN dalam klasifikasi dapat menghasilkan tingkat akurasi yang lebih baik dibandingkan dengan model CNN murni, namun harus menggunakan parameter yang sesuai.

Penggunaan optimasi adam dan *learning rate* 0,001 menghasilkan tingkat akurasi yang lebih tinggi dibandingkan dengan optimasi RMSprop dan *learning rate* 0,0001. Percobaan dengan akurasi tertinggi terjadi pada model hibrida CNN-KNN dengan menggunakan optimasi adam, *learning rate* 0,001, dan parameter nilai tetangga terdekat 9 dengan mencapai akurasi sebesar 94,03% diikuti dengan nilai *precision* 94%, *recall* 94%, dan *f1 score* 94%.

Untuk penelitian berikutnya, disarankan untuk mempertimbangkan penggunaan hyperparameter pada model CNN-KNN guna meningkatkan akurasi dalam model yang digunakan. Hal ini disebabkan karena hyperparameter menyediakan beragam parameter yang dapat disesuaikan. Selain itu, disarankan juga untuk memilih dataset yang lebih beragam agar model dapat memahami data dengan lebih baik selama proses pelatihan, sehingga dapat meningkatkan akurasi hasil klasifikasi.

DAFTAR PUSTAKA

- [1] Muhlashin, M. N. I., & Stefanie, A. (2023). Klasifikasi Penyakit Mata Berdasarkan Citra Fundus Menggunakan YOLO V8. JATI (Jurnal Mahasiswa Teknik Informatika), 7(2), 1363-1368.
- [2] Amanaturohim, A., & Wibisono, S. (2021). Penentuan Parameter Terbobot Menggunakan Pairwise Comparison Untuk CBR Deteksi Dini Penyakit Mata. J-SAKTI (Jurnal Sains Komputer Dan Informatika), 5(1), 280-294.
- [3] Dewi, R. I. S., & Novia, V. R. (2020). Pengaruh Senam Yoga Mata Terhadap Penurunankelelahan Mata Pada Pekerja Di Divisi Redaksi Padang Ekspres. Jurnal Kesehatan Mesencephalon, 6(1).
- [4] KEMENTERIAN KESEHATAN RI, PETA JALAN PENANGGULANGAN GANGGUAN PENGLIHATAN DI INDONESIA TAHUN 2017 -2030, Jakarta, 2017
- [5] Andika, L. A., Pratiwi, H., & Handajani, S. S. (2019). Klasifikasi penyakit pneumonia menggunakan metode convolutional neural network dengan optimasi adaptive momentum. Indonesian Journal of Statistics and Its Applications, 3(3), 331-340
- [6] Jaya, T. S. (2021). Klasifikasi Nanas Layak Jual Dengan Metode Naïve Bayes Classifier Dan K-Nearest Neighbor. JSI: Jurnal Sistem Informasi (E-Journal), 13(1)
- [7] Prahudaya, T. Y., & Harjoko, A. (2017). Metode Klasifikasi Mutu Jambu Biji Menggunakan Knn Berdasarkan Fitur Warna Dan Tekstur. Jurnal Teknosains, 6(2), 113-123.
- [8] Danendra, F. A., Anggraeny, F. T., & Maulana, H. (2023). Klasifikasi Citra Penyakit Daun Cabai Rawit Dengan Menggunakan CNN Arsitektur AlexNet dan SqueezeNet. Syntax: Jurnal Informatika, 12(01), 50-61.
- [9] Handono, S. F., Anggraeny, F. T., & Rahmat, B. (2020). Implementasi Convolutional Neural Network (CNN) untuk Deteksi Retinopati Diabetik. Jurnal Informatika dan Sistem Informasi, 1(2), 669-678.
- [10] Fitri, A. A. (2022). PERBANDINGAN ARSITEKTUR VGG-16 DAN RESNET-50 DENGAN OPTIMASI ADAM DAN RMSPROP PADA KLASIFIKASI CITRA PENYAKIT DAUN PADI. UPN Veteran Jawa Timur.
- [11] Liantoni, F. (2015). Klasifikasi Daun Dengan Perbaikan Fitur Citra Menggunakan Metode K-Nearest Neighbor. Ultimatics: Jurnal Teknik Informatika, 7(2), 98-104.
- [12] Rizki, A. M. (2021). Sistem Pakar Diagnosa Penyakit Menular Seksual Menggunakan K-NN. Scan: Jurnal Teknologi Informasi dan Komunikasi, 16(2), 34-37.
- [13] Harshith, "Text Preprocessing in Natural Language Processing," 2019. <https://towardsdatascience.com/text-preprocessing-in-natural-language-processing-usingpython-6113ff5decd>
- [14] Witantoa, K. S., ERa, N. A. S., Karyawatia, A. E., Arya, I. G. A. G., Kadyanana, I., & Astutia, L. G. (2022). Implementasi LSTM pada Analisis Sentimen Review Film Menggunakan Adam dan RMSprop Optimizer. Jurnal Elektronik Ilmu Komputer Udayana p-ISSN, 2301, 5373.
- [15] Normawati, D., & Prayogi, S. A. (2021). Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter. J-SAKTI (Jurnal Sains Komputer Dan Informatika), 5(2), 697-711.