

PROTOTYPE ALAT IDENTIFIKASI POLA NERVE RING PADA IRIS MATA MENGGUNAKAN RASPBERRY PI DAN DETECTRON2

Dastin Arjuna Wianzah ¹⁾ , Ahmad Ridha ²⁾

1) Teknologi Rekayasa Komputer Institut Pertanian Bogor, Jalan Raya Dramaga, Bogor, Indonesia

2) Ilmu Komputer Institut Pertanian Bogor, Jalan Raya Dramaga, Bogor, Indonesia

email : 07dastin@apps.ipb.ac.id¹⁾

Abstrak

Kemajuan teknologi dalam bidang pengolahan citra dan kecerdasan buatan telah membuka peluang besar dalam berbagai aplikasi medis, termasuk dalam analisis iris mata. Studi ini mengembangkan sebuah prototipe alat berbasis Raspberry Pi untuk identifikasi pola nerve ring pada iris mata. Perancangan prototipe ini mencakup requirement analysis, system and hardware design, implementation, dan testing. Alat ini mengambil citra iris mata yang kemudian diproses dengan library Detectron2 untuk mengidentifikasi pola nerve ring dan hasilnya ditampilkan pada layar. Prototipe ini berhasil mengenali keberadaan nerve ring pada 93,33% data dengan mendapatkan MSE sebesar 3,27 pada nilai NMS sebesar 0,2. Hal ini menunjukkan kinerja yang baik dalam mengidentifikasi nerve ring secara efektif untuk digunakan dalam analisis lebih lanjut.

Kata Kunci :

citra, detectron2, iris mata, machine learning, nerve ring.

Abstract

Advancements in technology within the fields of image processing and artificial intelligence have opened significant opportunities for various medical applications, including iris analysis. This study develops a Raspberry Pi-based prototype tool for identifying nerve ring patterns in the iris. The prototype design includes requirement analysis, system and hardware design, implementation, and testing. The tool captures iris images, which are then processed using the Detectron2 library to identify nerve ring patterns, and the results are displayed on a screen. The prototype successfully detected the presence of nerve rings in 93.33% of the data, with an MSE of 3.27 at an NMS value of 0.2. This demonstrates good performance in effectively identifying nerve rings for further analysis.

Keywords :

image, detectron2, iris, machine learning, nerve ring.

1. PENDAHULUAN

Selain merupakan organ yang sangat penting bagi manusia dalam berbagai aktivitas sehari-hari, mata juga dapat menjadi jendela untuk mendeteksi berbagai penyakit non-mata. Banyak penyakit sistemik memiliki gejala yang termanifestasi pada mata, memungkinkan deteksi dini dan intervensi. Penyakit auto imun seperti *rheumatoid arthritis* dan *Sjogren's syndrome* sering kali menimbulkan gejala mata kering dan inflamasi [1]. Hipertensi juga dapat menyebabkan perubahan pada pembuluh darah retina, yang dapat dideteksi melalui pemeriksaan mata rutin [2] dan bisa menjadi penanda risiko kejadian seperti *stroke* dan gagal jantung kongestif.

Mata, khususnya iris, sebagai jendela untuk melihat kondisi kesehatan menjadi konsep sentral dalam sebuah pengobatan alternatif yang disebut iridologi [3]. Pendekatan ini masih kontroversial, dan evaluasi medis secara umum belum menunjukkan kemampuan yang signifikan misalnya dalam mendeteksi penyakit ginjal [4]. Di sisi lain, kemajuan teknologi memungkinkan citra iris yang lebih jelas dan detail sehingga iridologi menjadi kajian menarik dalam bidang pengolahan citra dan kecerdasan buatan [5][6][7][8][3][9][10] dengan akurasi prediksi 80-97% walau kajian medis lebih lanjut masih diperlukan.

Salah satu pola unik dalam iris mata adalah *nerve ring* [10]. Pola ini dapat ditandai dengan adanya lipatan melingkar dari tepi iris hingga dekat pupil mata. Studi untuk identifikasi keberadaan *nerve ring* masih terbatas, dan *dataset* yang memadai untuk itu juga belum tersedia. Oleh sebab itu, studi ini menyusun *dataset* untuk *nerve ring* dengan menggunakan citra mata yang tersedia publik, merekam citra baru, dan melakukan augmentasi data. Studi ini juga merancang sebuah prototipe alat yang dapat merekam citra iris mata dan mengidentifikasi pola *nerve ring* pada citra tersebut dengan *machine learning*. *Library Detectron2* digunakan untuk proses identifikasi.

2. METODE / ALGORITMA

Pembuatan *Dataset*

Studi ini memerlukan data citra iris mata yang memiliki pola *nerve ring* untuk menguji alat yang dikembangkan. Citra bersumber dari *Chinese Academy of Sciences Institute of Automation (CASIA)*, Kaggle, Roboflow, dan perekaman langsung dengan kamera alat yang dikembangkan. Hanya citra mata yang memiliki pola *nerve ring* dimasukkan ke *dataset*.

Selanjutnya *dataset* diaugmentasi untuk meningkatkan keberagaman data yang tersedia untuk model pelatihan, tanpa harus mengumpulkan data baru dengan memanipulasi transformasi dimensi gambar [11]. Teknik augmentasi data juga dapat digunakan untuk meningkatkan kinerja deteksi dengan memperkenalkan variasi tambahan dalam *dataset* pelatihan melalui beberapa jenis augmentasi [12]. Empat jenis augmentasi data dilakukan, yakni *flip*, *90-degree rotation*, *greyscale*, dan *exposure*. Citra selanjutnya dianotasi dengan CVAT.ai.

Pengembangan Prototipe

Prototipe alat identifikasi *nerve ring* dikembangkan dengan metode *System Development Life Cycle (SDLC)* melalui tahapan analisis kebutuhan sistem, perancangan sistem, implementasi sistem, pengujian sistem, dan perawatan/pemeliharaan [13]. Alat yang dikembangkan masih bersifat prototipe sehingga pengembangan alat ini hanya sampai tahap pengujian sistem.

Analisis Kebutuhan Sistem

Tahap ini mengidentifikasi masalah dan kebutuhan serta menentukan solusi untuk menyelesaikan permasalahan. Analisis masalah melibatkan pencarian informasi tentang permasalahan dalam mengidentifikasi pola *nerve ring*. Analisis kebutuhan menghasilkan alat, bahan, dan komponen yang diperlukan.

Perancangan Sistem

Perancangan meliputi pembuatan diagram blok, diagram alir, serta skema rangkaian di draw.io, perancangan *casing* di Fusion360, dan perancangan sisi *software*.

Implementasi Sistem

Tahapan ini merupakan implementasi dari rancangan yang telah dilakukan pada tahap perancangan, yang mencakup merangkai komponen, uji coba alat, perakitan alat pada *casing*, pelabelan *dataset*, dan melatih model *machine learning*. Tujuannya adalah pembuatan *prototype* alat pengidentifikasi pola iris mata. Pelatihan dilakukan di Google Colab dengan *library* PyTorch, Cython, PyYAML, COCOAPI, dan Detectron2.

Pengujian Sistem

Tahapan ini menguji apakah *prototype* alat sudah dapat berjalan sebagai mestinya atau tidak. Pengujian dilakukan dengan menggunakan alat untuk merekam citra dengan kamera hingga hasil inferensi ditampilkan.

Evaluasi Model

Akurasi model *machine learning* yang dihasilkan dievaluasi dengan membagi banyaknya citra yang teridentifikasi memiliki pola *nerve ring* dengan banyaknya semua citra. Kinerja juga dievaluasi dengan *Mean Square Error* (MSE) untuk memeriksa apakah jumlah *nerve ring* yang terdeteksi sesuai dengan labelnya. MSE dihitung sebagai berikut:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

dengan:

- n = jumlah data.
- y_i = adalah banyaknya *nerve ring* pada citra ke- i .
- \hat{y}_i = adalah banyaknya *nerve ring* yang teridentifikasi pada data ke- i .

3. HASIL DAN PEMBAHASAN

Pembuatan *Datasest*

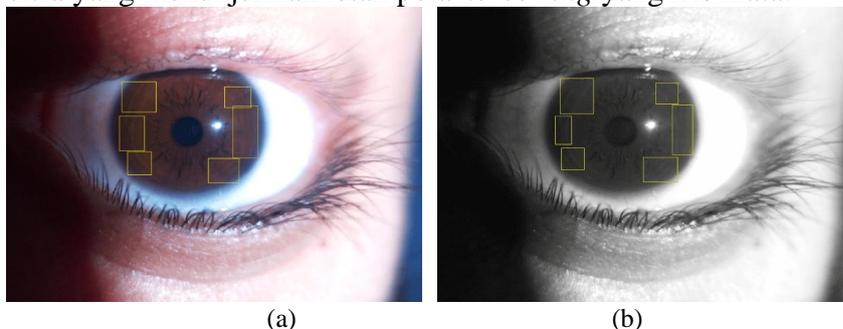
Jumlah citra yang terkumpul dari CASIA, Kaggle, Roboflow, dan perekaman langsung dengan kamera alat yang dikembangkan disajikan pada Tabel 1. Citra yang terkumpul kemudian dipilah untuk menghasilkan kumpulan citra mata yang memiliki pola *nerve ring*. Jumlah data setelah augmentasi bertambah dari 673 menjadi 1.348 citra.

Datasest ini mencakup citra *red green blue* (RGB) dan *greyscale*. Pola *nerve ring* pada iris mata ditandai dengan kotak-kotak pembatas (*bounding box*) berwarna kuning (lihat Gambar 1). Pola *nerve ring* pada iris mata sering kali terlihat sebagai garis-garis melingkar atau semilunar di sekitar pupil.

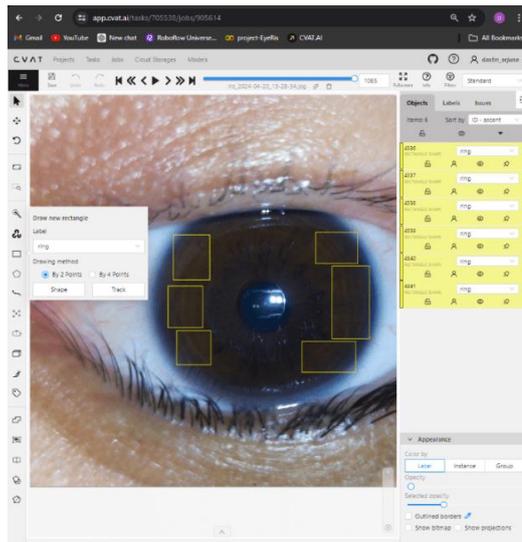
Tabel 1 Sumber data citra

Sumber Data	Jumlah Data Keseluruhan	Jumlah Data dengan Pola Nerve ring
CASIA	1032	202
Kaggle	325	170
Kamera Alat	124	64
Roboflow	2313	237
Total	4254	673

Tahap selanjutnya anotasi *dataset* citra iris mata di platform CVAT.ai seperti yang terlihat pada Gambar 2. Setelah nama dan warna label diatur, persegi panjang digunakan sebagai penanda pada citra yang menunjukkan letak pola *nerve ring* yang iris mata.



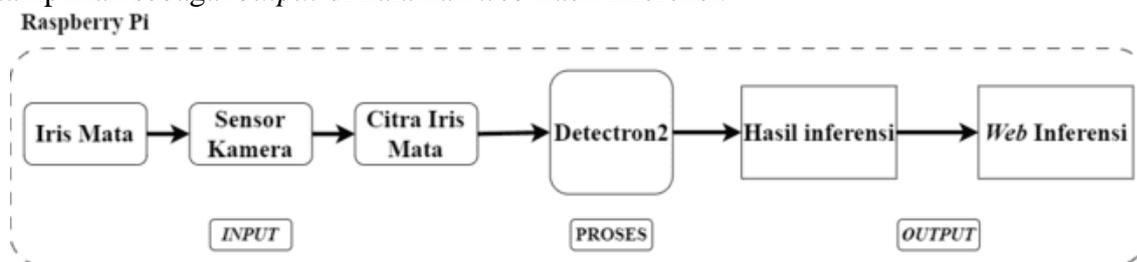
Gambar 1 Contoh data citra yang telah dilabeli: (a) RGB, dan (b) *greyscale*



Gambar 2 Proses anotasi citra iris mata di CVAT.ai

Perancangan Sistem

Prototipe alat dirancang dalam bentuk diagram blok, diagram alir, skema rangkaian, pembuatan desain 3D *casing* alat, dan perancangan sisi *software*-nya. Diagram blok pada Gambar 3 menyajikan hubungan antara *input* dan *output* alat ini. Raspberry Pi berfungsi sebagai pusat pemrosesan, termasuk pengambilan dan pemrosesan citra dari kamera. Kamera menangkap citra iris mata yang kemudian diunggah ke *web* inferensi untuk diproses oleh Detectron2. Hasil deteksi berupa citra dengan kotak pembatas yang menyoroti pola *nerve ring* ditampilkan sebagai *output* di halaman *web* hasil inferensi.



Gambar 3 Diagram blok

Diagram alur sistem identifikasi pola *nerve ring* iris mata pada Gambar 4 menunjukkan langkah-langkah prosesnya. Sistem dimulai dengan inisiasi perangkat, diikuti dengan akses kamera melalui Raspberry Pi untuk mengambil citra iris mata. Citra ini kemudian diunggah ke *web* untuk proses akuisisi sebelum inferensi. Inferensi menggunakan model *machine learning* yang telah dilatih untuk menginterpretasi data baru dan memberikan *output* berdasarkan pengetahuan dari proses pelatihan.



Gambar 4 Diagram alir

Pengembangan model dimulai dengan penyiapan data latih, data uji, dan data validasi. *Dataset* dibagi menjadi tiga bagian: data latih (80%), uji (10%), dan validasi (10%) (Tabel 2). Model dilatih menggunakan Faster R-CNN dari *library* Detectron2.

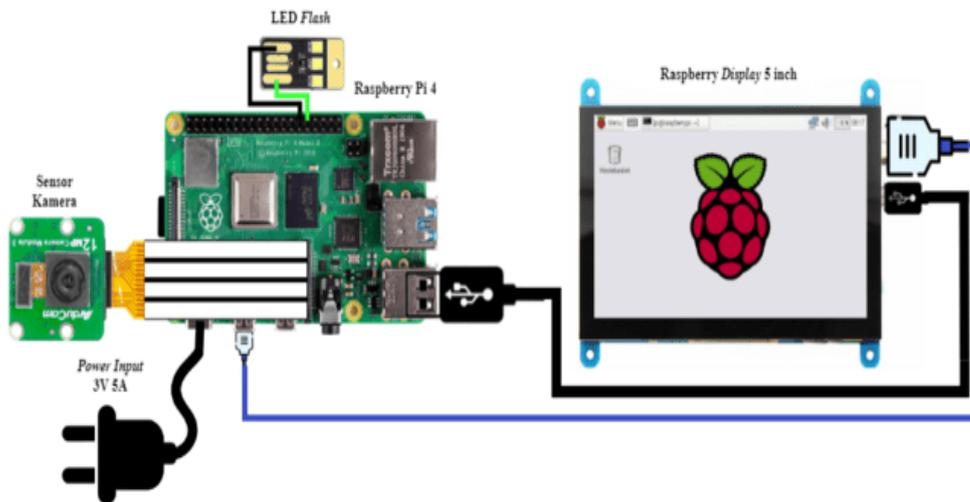
Tahap inferensi melibatkan penggunaan model yang sudah dilatih untuk mengolah data baru di *web* inferensi. Hasil deteksi ditampilkan dalam bentuk kotak pembatas (*bounding box*) pada iris mata yang menunjukkan pola *nerve ring*, beserta nilai persentase *confidence*-nya.

Tabel 2 Jumlah data sebelum dan sesudah augmentasi

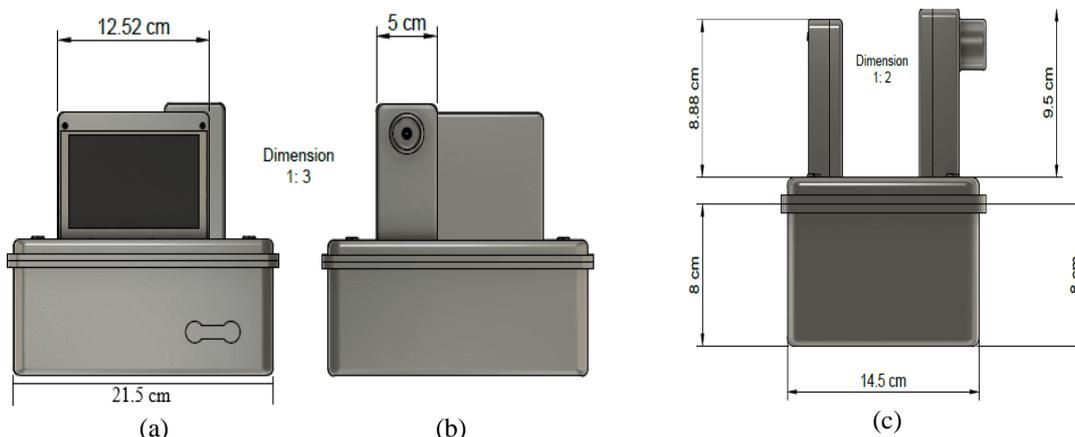
Jenis Data	Jumlah Data Sebelum Augmentasi	Jumlah Data Setelah Augmentasi
Data Latih	539	1078
Data Uji	67	135
Data Validasi	67	135
Total	673	1348

Skema rangkaian untuk *prototype* alat pengidentifikasi pola *nerve ring* dibuat di draw.io (lihat Gambar 5) untuk memvisualisasikan koneksi antara pin/*port* yang terdapat pada Raspberry Pi dengan komponen lainnya. Desain *casing* alat dibuat dengan Fusion360 (lihat Gambar 6). *Casing* berukuran panjang 21,5 cm, lebar 14,5 cm, dan tinggi 8 cm, difungsikan untuk menyusun rangkaian komponen. Bagian atas *casing* memiliki tempat khusus untuk kamera dan LED, serta layar LCD sentuh sebagai antarmuka untuk mengakses Raspberry Pi. Di samping LCD terdapat lubang untuk kabel HDMI dan *micro* USB, sementara bagian belakang *casing* memiliki lubang untuk kabel daya USB Type-C.

Antarmuka sistem terdiri atas dua bagian: antarmuka akses kamera dan antarmuka *web* inferensi. Antarmuka akses kamera menyediakan pratinjau sebelum pengambilan citra, lengkap dengan tombol pengambilan citra. Antarmuka *web* inferensi memiliki fitur-fitur seperti mengunggah citra, mengatur nilai *confidence* dan *overlap*, menampilkan hasil inferensi dalam citra atau JSON, menampilkan keterangan *bounding box*, dan menampilkan hasil inferensi di bagian bawah halaman *web*.



Gambar 5 Skema rangkaian alat



Gambar 6 Desain 3D casing alat: (a) sisi output, (b) sisi kamera, dan (c) tampak samping

Implementasi Sistem

Langkah selanjutnya adalah tahap implementasi yang dimulai dengan menyusun prototipe rangkaian komponen. Modul kamera dihubungkan ke *port Camera Serial Interface (CSI)* menggunakan kabel 15-pin CSI. Pin *ground LED* dihubungkan ke pin GPIO 30 (*Ground*), dan pin *VCC* dihubungkan ke pin GPIO 12 (*PWM*). Terakhir, LCD layar sentuh dihubungkan ke *port HDMI* dan *port USB*.

Setelah penyusunan rangkaian, alat diuji untuk memastikan semua komponen berfungsi dengan baik. Uji coba ini melibatkan pengambilan gambar iris mata menggunakan modul kamera, penerapan pencahayaan yang tepat melalui LED, dan penyajian antarmuka Raspberry OS pada LCD layar sentuh. Jika ada komponen yang tidak berfungsi, perbaikan dilakukan hingga seluruh sistem bekerja secara optimal.

Setelah uji coba berhasil, langkah berikutnya adalah merakit komponen-komponen tersebut ke dalam casing alat. Casing alat (lihat **Gambar 7**) dirancang agar semua komponen terpasang dengan aman dan teratur, meminimalkan gangguan eksternal serta memastikan kemudahan penggunaan. Modul kamera, LED flash, dan LCD layar sentuh ditempatkan pada posisi yang strategis untuk mendukung fungsionalitas alat secara maksimal. Perakitan mencakup pemasangan kabel dan konektor secara rapi agar tidak mengganggu operasi alat. Setelah perakitan selesai, alat diuji kembali.

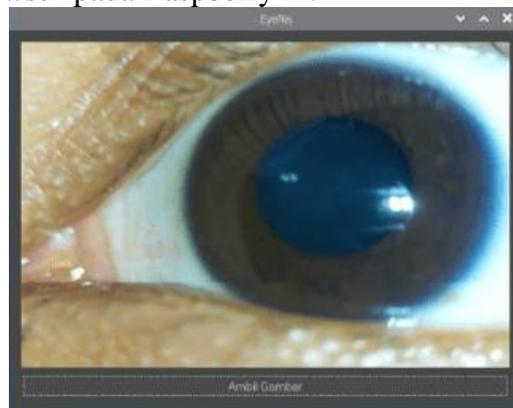
Langkah selanjutnya adalah melatih *dataset* citra iris mata di Google Colab. *Hyperparameter* model disesuaikan dengan menggunakan empat *worker*, tiga ribu iterasi, *batch size* 128, dan *learning rate* 0,0001, dengan menggunakan *pre-trained* model Faster R-CNN X101 32x8d FPN 3x dari GitHub Detectron2 Model Zoo.



Gambar 7 Perangkaian prototipe alat: (a) sisi *output*, (b) sisi kamera

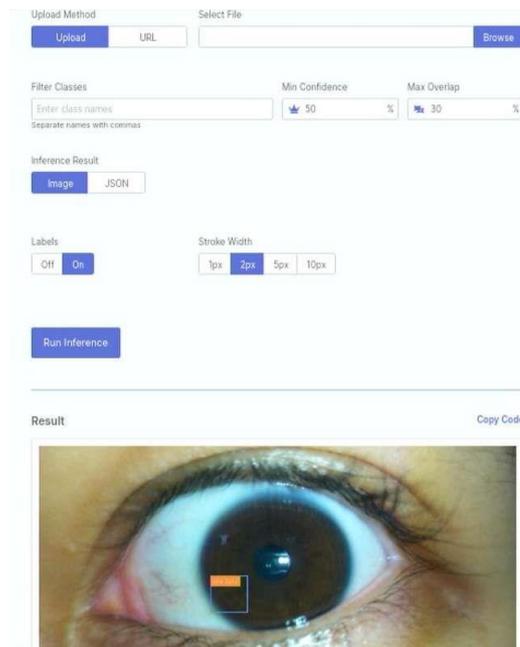
Pengujian Sistem

Setelah tahap implementasi, langkah selanjutnya adalah menguji prototipe alat. Pengujian dilakukan dengan mendeteksi *nerve ring* pada iris mata melalui citra yang diambil menggunakan modul sensor kamera. Proses pengambilan citra iris mata, seperti terlihat pada Gambar 8, menggunakan program yang sudah dibuat pada tahap sebelumnya. Citra iris mata diambil dengan menekan tombol pada alat. Citra disimpan pada penyimpanan internal Raspberry Pi di dua folder yang berbeda dengan format penamaan citra sesuai dengan foldernya (RGB dan *greyscale*). Citra yang diambil kemudian diunggah ke *web inference* Roboflow melalui *web browser* pada Raspberry Pi.



Gambar 8 Tampilan antarmuka kamera

Setelah proses inferensi selesai, hasilnya akan menampilkan citra iris mata yang sudah memiliki *bounding box* dan label berupa nama kelas dan nilai *confidence* model dalam mendeteksi objek seperti yang divisualisasikan pada Gambar 9. Informasi ini dapat digunakan untuk mengetahui seberapa baik model dalam mengidentifikasi *nerve ring* secara efektif untuk digunakan dalam analisis lebih lanjut.



Gambar 9 Hasil inferensi citra pada web

Evaluasi Model

Model yang digunakan pada studi deteksi pola *nerve ring* ini berjenis *object detection*, dengan bisa ada lebih dari satu objek di dalam tiap citra. Model yang dihasilkan berhasil mendeteksi keberadaan *nerve ring* pada 126 dari 135 data uji yang berarti model memiliki akurasi 93,33%. Ini menunjukkan bahwa model cukup mampu mendeteksi pola *nerve ring* pada iris mata. Selanjutnya, model ini menghasilkan MSE 5,07. Pemeriksaan terhadap hasil identifikasi menunjukkan adanya *bounding box* yang tumpang tindih sehingga ada *nerve ring* yang terhitung ganda.

Untuk meminimalkan perhitungan ganda tersebut, *Non-Maximum Suppression* (NMS) digunakan dengan parameter awal 0,1, 0,2, dan 0,5. Hasilnya menunjukkan MSE terkecil di NMS 0,2 yakni 3,27. Nilai MSE di aneka parameter NMS disajikan pada Tabel 3. Model ini menggunakan *confidence threshold* sebesar 0,5 untuk memastikan bahwa hanya deteksi dengan tingkat kepercayaan minimal tersebut yang dipertimbangkan.

Tabel 3 Hasil perhitungan MSE berdasarkan nilai NMS

NMS	MSE
Tanpa NMS	5,07
0,100	3,73
0,125	3,39
0,150	3,30
0,175	3,41
0,200	3,27
0,500	5,09

Penggunaan NMS dengan berbagai nilai diterapkan untuk mengurangi deteksi ganda karena bertumpang tindih (*overlap*), yang diukur menggunakan metrik *Intersection over Union* (IoU). NMS mengurangi deteksi ganda objek dengan mempertahankan hasil deteksi dengan nilai *confidence* tertinggi dan menghapus hasil deteksi lainnya dalam area yang berdekatan karena dianggap redundan. Penggunaan NMS yang bervariasi membantu mengevaluasi

efeknya terhadap akurasi identifikasi, memastikan bahwa pendeteksian tidak hanya akurat tetapi juga relevan dan bebas dari *noise* [14].

4. KESIMPULAN

Studi ini telah berhasil mengembangkan sebuah prototipe alat identifikasi *nerve ring* berbasis Raspberry Pi. Untuk pengujian, sebuah *dataset* citra iris mata dengan pelabelan *nerve ring* yang terdiri atas 1348 citra telah disusun. Identifikasi dengan Detectron2 yang berhasil mendeteksi keberadaan *nerve ring* pada 93,33% citra uji dengan nilai MSE terendah sebesar 3,27 untuk nilai NMS sebesar 0,2.

Studi lebih lanjut dapat dilakukan untuk menambahkan citra dalam *dataset* dan mengecek pelabelan dengan pakar. Prototipe juga bisa ditingkatkan dengan lensa makro beresolusi tinggi untuk menangkap citra detail agar pola *nerve ring* pada iris mata terlihat lebih jelas. Metode *deep learning* lain dapat dieksplorasi untuk meningkatkan kinerja alat.

5. REFERENSI

- [1] S. J. Patel and D. C. Lundy, "Ocular manifestations of autoimmune disease," *Am. Fam. Physician*, vol. 66, no. 6, pp. 991–998, 2002.
- [2] Bhargava, M., Ikram, M. K., & Wong T. Y. (2012). How does hypertension affect your eyes? *Journal of Human Hypertension*, 26:71-83.
- [3] R. ANUGRAH, "Sistem Pendukung Diagnosa Iridologi Dengan Menggunakan Pengolahan Citra Digital Metode Transformasi Hough," *Insa. Pembang. Sist. Inf. dan Komput.*, vol. 9, no. 2, 2022.
- [4] Simon, A., Worthen, D. M., & Mitas, J. A. (1979). An Evaluation of Iridology. *JAMA*, 242(13):1385–1389. doi:10.1001/jama.1979.03300130029014
- [5] Avhad, V. V., & Bakal, J. W. (2024). Iridology based human health conditions predictions with computer vision and deep learning. *Biomedical Signal Processing and Control*, 96, Part A, October 2024, 106656.
- [6] Alphonse, S., Venkatesan, R., & Jebaseeli, T. J. (2023). A Methodical Review of Iridology-Based Computer-Aided Organ Status Assessment Techniques. *Eng Proc*, 59, 9.
- [7] Önal, M. N., Güraksin, G. E., & Duman R. (2023). Convolutional neural network-based diabetes diagnostic system via iridology technique. *Multimedia Tools and Applications*, 82:173:194.
- [8] Esteves, R. B., Morero, J. A. P., Pereira, S. d. S. P., Mendes, K. D. S., Hegadoren, K. M., & Cardoso, L. (2021). Parameters to increase the quality of iridology studies: A scoping review. *European Journal of Integrative Medicine*, 43(April 2021):101311.
- [9] "12871-Article Text-24884-1-10-20201110.pdf" .
- [10] I. Agustian, F. Hadi, and M. K. A. Rosa, "Pre-Diagnosis Gangguan Ginjal Melalui Citra Iris Mata Menggunakan Raspberry PI Dengan Metode Convolutional Neural Network (CNN)," *J. Amplif. J. Ilm. Bid. Tek. Elektro Dan Komput.*, vol. 9, no. 1, pp. 16–25, 2019.
- [11] Mikołajczyk, A., & Grochowski M. (2018). Data augmentation for improving deep learning in image classification problem. *2018 International Interdisciplinary PhD Workshop (IIPHDW)*, pp. 117-122, doi: 10.1109/IIPHDW.2018.8388338.
- [12] M. Toyib, T. D. K. Pratama, and I. Aqil, "Penerapan Algoritma CNN untuk Mendeteksi Tulisan Tangan AngkaRomawi dengan Augmentasi Data," *Algoritm. J.*

Mat. Ilmu Pengetah. Alam, Kebumian dan Angkasa, vol. 2, no. 3, pp. 108–120, 2024.

- [13] G. G. Salindeho and T. Wellem, “Perancangan Dan Implementasi Sistem Pendeteksi Dan Peringatan Kebakaran Berbasis Iot Menggunakan Nodemcu Esp8266 Dan Sensor Api,” *IT-Explore J. Penerapan Teknol. Inf. dan Komun.*, vol. 2, no. 3, pp. 179–191, 2023.
- [14] Zheng Z., Wang P., Ren D., Liu W., Ye R., Hu Q., & Zuo W. (2021). Enhancing Geometric Factors in Model Learning and Inference for Object Detection and Instance Segmentation. *IEEE Trans Cybern.* 52(8):8574–8586. doi: 10.1109/TCYB.2021.3095305.