

Rancang Bangun Prototipe Sistem Pengusir Burung Cerdas (*Smart Bird Deterrent*) di Area Airside Bandara Internasional I Gusti Ngurah Rai, Kabupaten Badung, Bali

I Kadek Rika Artawan¹, Lie Jasa²

[Submission: 11-06-2026, Accepted: 30-06-2026]

Abstract— *Bird strike incidents in airport airside areas pose a high-risk safety threat. Current conventional mitigation efforts rely on open-loop acoustic alarm devices that operate continuously for 24 hours. This method causes massive electrical energy waste and triggers habituation phenomena in target wildlife. This research aims to design a smart bird deterrent system prototype based on artificial intelligence utilizing a closed-loop control architecture. The methodology integrates the YOLOv8 Nano computer vision algorithm as a visual feedback sensor unit, deployed on low-end Edge AI computing (Intel Celeron N3350). To prevent CPU computational bottlenecks, the model is optimized into an Intermediate Representation format using Intel OpenVINO. The audio actuator trigger signal is strictly executed only when the bird detection accuracy exceeds a 0.5 confidence threshold. Test results indicate that the algorithm dynamically responds to wildlife presence, precisely halting audio emissions when the area is clear, effectively solving the habituation problem. The OpenVINO framework optimization successfully accelerated the image processing computational rate from 0.80 FPS to a stable 1.30 FPS. The system proved reliable with a 0.77-second response latency, offering an innovative control engineering output for energy efficiency in aviation facilities.*

Intisari— Insiden *bird strike* di area *airside* bandar udara merupakan ancaman keselamatan berisiko tinggi. Upaya mitigasi konvensional saat ini menggunakan perangkat alarm akustik lingkaran terbuka (*open-loop*) yang beroperasi 24 jam secara berkelanjutan. Metode ini terbukti memicu pemborosan konsumsi energi listrik dan mengakibatkan fenomena habituasi pada satwa target. Penelitian ini bertujuan merancang purwarupa sistem pengusir burung cerdas berbasis kecerdasan buatan dengan arsitektur lingkaran tertutup (*closed-loop*). Metodologi yang digunakan mengintegrasikan algoritma visi komputer YOLOv8 Nano sebagai unit sensor umpan balik (*feedback*) visual, yang dijalankan pada komputasi tepi (*Edge AI*) berspesifikasi rendah (Intel Celeron N3350). Untuk mencegah *bottleneck* komputasi CPU, model dioptimasi ke dalam format *Intermediate Representation* menggunakan Intel OpenVINO. Sinyal pemicu aktuatur audio hanya dieksekusi apabila nilai akurasi deteksi burung melampaui *confidence threshold* 0,5.

Hasil pengujian menunjukkan bahwa algoritma mampu merespons kehadiran satwa secara dinamis, menghentikan pancaran audio secara presisi saat area bersih, sehingga memecahkan masalah habituasi. Optimasi *framework* OpenVINO berhasil mengakselerasi laju komputasi pemrosesan citra secara signifikan dari 0,80 FPS menjadi laju stabil 1,30 FPS. Sistem terbukti andal dengan latensi respons 0,77 detik, menawarkan luaran rekayasa sistem kendali inovatif untuk efisiensi energi fasilitas penerbangan.

Kata Kunci— Bird Strike; Closed-loop; Computer Vision; Edge AI; OpenVINO; YOLOv8.

I. PENDAHULUAN

Insiden *bird strike* atau benturan antara satwa liar bersayap dengan pesawat udara merupakan ancaman keselamatan operasional berisiko tinggi di area *airside* bandar udara. Kerusakan akibat benturan burung dapat secara serius membahayakan integritas struktural pesawat udara [1]. Tingginya intensitas manuver pesawat yang bertepatan dengan jam aktif satwa diurnal menuntut adanya sistem mitigasi yang mutlak dan tanpa toleransi kegagalan (*zero tolerance*). Data menunjukkan tren peningkatan laporan insiden benturan burung setiap tahunnya, di mana mayoritas kejadian berlangsung di dalam atau di sekitar wilayah bandar udara [2]. Saat ini, sistem pengusir burung (*Acoustic Bird Deterrent*) yang beroperasi di lapangan masih mengandalkan arsitektur kendali lingkaran terbuka (*open-loop system*). Sistem konvensional ini memancarkan gelombang suara secara terus-menerus selama 24 jam tanpa umpan balik (*feedback*) dari kondisi lingkungan aktual. Kondisi *continuous state* ini menghasilkan inefisiensi konsumsi energi dan secara biologis memicu fenomena habituasi, di mana respons satwa menurun terhadap stimulus ancaman yang repetitif. Tantangan operasional ini secara faktual terjadi pada ekosistem pesisir Bandara Internasional I Gusti Ngurah Rai, Bali, yang menuntut adanya solusi mitigasi yang adaptif.

Untuk menambal kelemahan fundamental tersebut, penelitian ini mengusulkan pembaruan sistem menuju otomasi berbasis visi komputer (*Computer Vision*) yang terintegrasi dalam arsitektur kendali lingkaran tertutup (*closed-loop*). Penelitian ini memanfaatkan algoritma deteksi objek mutakhir YOLOv8 (*You Only Look Once* versi 8) varian Nano, yang dieksekusi melalui pendekatan *Edge Computing* secara lokal. Sistem dirancang agar aktuatur alarm akustik hanya menerima sinyal pemicu (*trigger*) aktif ketika model kecerdasan buatan secara presisi memvalidasi kehadiran objek burung di area pantau. Komputasi tepi memungkinkan pemrosesan data secara

¹Mahasiswa, Jurusan Teknik Elektro Fakultas Teknik Universitas Udayana, Jln. Kampus Unud Bukit Jimbaran 80361 INDONESIA (telp: 0361-703315; fax: 0361-4321; e-mail: artawan.2305541014@student.unud.ac.id)

²Dosen, Jurusan Teknik Elektro Fakultas Teknik Universitas Udayana, Jln. Kampus Unud Bukit Jimbaran 80361 INDONESIA (telp: 0361-703315; fax: 0361-4321; e-mail: liejasa@unud.ac.id)



otonom di lokasi pengambilan data, yang sangat krusial untuk manajemen konflik manusia dan satwa liar secara *real-time* [3].

Penelitian ini bertujuan untuk merancang prototipe perangkat lunak sistem *Smart Bird Deterrent* dan menguji fungsionalitas logika *closed-loop* dalam mitigasi habituasi satwa. Selain itu, penelitian ini juga mengevaluasi akselerasi beban komputasi *Frame Per Second* (FPS) pada perangkat keras berspesifikasi rendah menggunakan optimasi *framework* Intel OpenVINO. Pembaruan ini diharapkan dapat memberikan luaran berupa rekomendasi desain sistem keamanan *airside* cerdas berskala industri yang lebih presisi dan hemat energi.

II. STUDI PUSTAKA

Kajian pustaka ini memetakan *state of the art* dari teknologi keamanan *airside* dengan membandingkan penelitian terdahulu. Penelitian sebelumnya telah mengeksplorasi rancang bangun sistem pengusir hewan berbasis *Edge-AI* tersemat untuk perlindungan area spesifik secara presisi [4]. Secara paralel, sistem mitigasi keselamatan lain telah dieksplorasi menggunakan fusi algoritma YOLO guna mendeteksi kehadiran kawanan burung secara komprehensif di lingkungan bandar udara [5]. Upaya peningkatan akurasi deteksi burung juga dilakukan melalui modifikasi arsitektur YOLOv8 dengan penambahan mekanisme atensi (*attention mechanism*) dan *feature fusion* untuk menangani target berukuran kecil [2], [6]. Letak perbedaan fundamental (*novelty*) yang diajukan dalam penelitian ini dibandingkan literatur tersebut adalah implementasi kendali aktuator lingkaran tertutup untuk mereduksi habituasi, yang dieksekusi secara efisien pada komputasi spesifikasi rendah (Intel Celeron N3350) melalui optimasi format *Intermediate Representation* Intel OpenVINO. Selain optimasi *framework*, penelitian lain juga mengeksplorasi penggunaan model hibrida seperti penggabungan YOLOv8 dengan MobileNet V3 untuk klasifikasi spesies burung pada perangkat seluler. Selain itu, teknik pemangkasan lapisan (*layer pruning*) telah terbukti efektif dalam mereduksi ukuran model dan waktu inferensi untuk perangkat tepi [7], [8].

A. Sistem Kendali Tertutup dan Mitigasi Habituasi

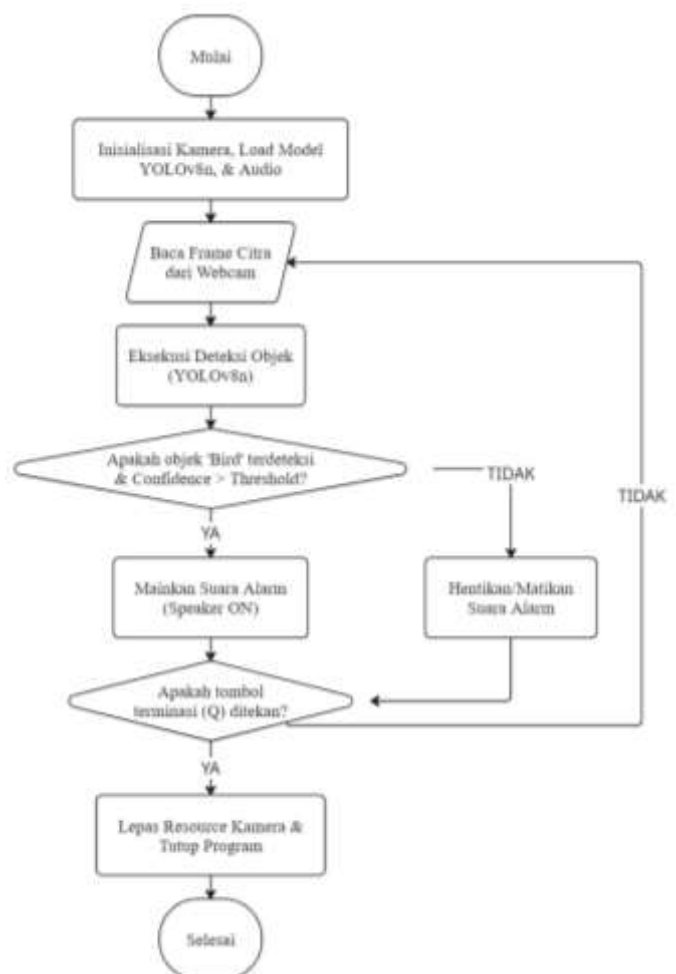
Dalam rekayasa sistem kendali, paradigma *open-loop* tidak memiliki elemen sensor untuk mendeteksi perubahan lingkungan aktual, sehingga keluaran tidak diumpanbalikkan ke masukan [9]. Pada mitigasi *bird strike*, pancaran suara bio-akustik yang diputar tanpa henti melalui sistem *open-loop* akan memicu habituasi, di mana otak burung dengan cepat beradaptasi dan meresponsnya sebagai alarm palsu [10], [11], [12]. Habituasi didefinisikan sebagai penurunan respons perilaku terhadap stimulus yang diberikan secara berulang dan intensitas yang kurang signifikan. Strategi pencegahan habituasi dapat dilakukan dengan memperkenalkan isyarat predator buatan secara dinamis berdasarkan isyarat akustik yang lebih sering memicu respons anti-predator dibandingkan isyarat visual [12], [13]. Transisi menuju arsitektur *closed-loop* menjadikan visi komputer sebagai sensor presisi yang memberikan sinyal umpan balik (*feedback*). Mekanisme ini memastikan aktuator hanya bekerja intermiten saat mendeteksi *error* (kehadiran burung), sehingga secara dramatis mencegah adaptasi satwa dan menghemat utilitas daya [4], [9], [14].

B. Arsitektur YOLOv8 Nano dan Optimasi OpenVINO

YOLOv8 YOLOv8 adalah arsitektur *Convolutional Neural Network* (CNN) bertipe *single-stage detector* yang merumuskan deteksi objek sebagai masalah regresi tunggal untuk lokalisasi koordinat spasial (*bounding box*) dan klasifikasi probabilitas kelas secara simultan [15], [16]. Varian YOLOv8 Nano (YOLOv8n) dipilih karena desain jaringan parameter pendukungnya sangat ringan, ideal untuk eksekusi komputasi tepi (*Edge AI*) pada perangkat keras dengan sumber daya terbatas tanpa unit pengolah grafis (GPU) terdedikasi [16]. Untuk menembus keterbatasan instruksi prosesor mikro lokal, model bawaan PyTorch dikompilasi ulang menggunakan *toolkit* Intel OpenVINO [17]. Proses ini mengonversi arsitektur model ke format *Intermediate Representation* (IR) sehingga instruksi matriks saraf tiruan dapat diakselerasi secara komprehensif pada lapisan perangkat keras, mencegah terjadinya latensi atau *bottleneck* komputasi selama eksekusi pemantauan [18].

III. METODE PENELITIAN

Penelitian ini dieksekusi menggunakan metode eksperimental terapan untuk merancang purwarupa sistem perangkat lunak *Smart Bird Deterrent*. Alur logika sistem (*state machine*) diuraikan ke dalam lima tahapan sistematis, yang divisualisasikan melalui diagram alir (*flowchart*) pada Gambar 1.



Gambar 1: Diagram alir logika sistem *Smart Bird Deterrent*

A. Tahapan I: Inisialisasi Environment dan Model

Sistem memulai urutan operasional dengan melakukan inisialisasi pada antarmuka *webcam* sebagai sensor visual. Secara simultan, sistem memuat (*load*) arsitektur kecerdasan buatan YOLOv8 Nano ke dalam memori. Arsitektur YOLOv8 Nano terdiri dari komponen *backbone*, *neck*, dan *head* yang dioptimalkan untuk keseimbangan antara kecepatan dan akurasi pada perangkat keras dengan sumber daya terbatas [8], [19]. Guna mencegah *bottleneck* instruksi pada CPU Intel Celeron, model yang dimuat adalah varian yang telah dikompilasi menggunakan *toolkit* Intel OpenVINO ke dalam format *Intermediate Representation* (IR). Pengujian berbagai varian YOLO pada komputer papan tunggal (*Single-Board Computers*) menunjukkan perlunya tolok ukur kinerja yang ketat untuk memastikan fungsionalitas sistem secara *real-time* di lingkungan tepi [20].

B. Tahapan II: Akuisisi Citra (*Image Fetching*) dan Prapemrosesan

Sistem mengevaluasi ketersediaan umpan video dari sensor visual. Apabila bingkai citra (*frame*) tidak tertangkap, sistem akan terus melakukan penungguan (*looping*) hingga mendapatkan sinyal visual. Setelah *frame* berhasil ditangkap, algoritma melakukan prapemrosesan (*pre-processing*) untuk menyesuaikan dimensi matriks gambar agar kompatibel dengan tensor masukan (*input tensor*) yang diwajibkan oleh model deteksi.

C. Tahapan III: Inferensi dan Evaluasi Probabilitas (*Thresholding*)

Algoritma YOLOv8 mengeksekusi deteksi objek pada citra yang telah diproses. Pada tahap ini, sistem memasuki blok percabangan kondisional (*decision node*). Model akan mengkalkulasi probabilitas (*Confidence Score*) atas keberadaan anatomi burung. Nilai matematis tersebut kemudian dievaluasi terhadap parameter ambang batas (*threshold*), yakni di atas 0,5 (50%).

D. Tahapan IV: Eksekusi Logika Aktuator (*Closed-Loop*)

Berdasarkan hasil evaluasi *threshold* pada Tahapan III, unit pengontrol membagi respons keluaran menjadi dua *state* aktuasi yang berlawanan:

1. Jika Ya (*Confidence Score* > 0.5): Sistem mengklasifikasikan status area sebagai "Bahaya" (*Error* > 0) dan secara otomatis mengirimkan sinyal penyalan *speaker* alarm (ON) untuk memancarkan bio-akustik.
2. Jika Tidak (*Confidence Score* < 0.5 atau objek nihil): Sistem mengklasifikasikan status area sebagai "Aman" (*Error* = 0) dan secara seketika memutus daya *speaker* alarm (OFF). Mekanisme intermiten ini adalah kunci utama dalam mencegah fenomena habituasi satwa.

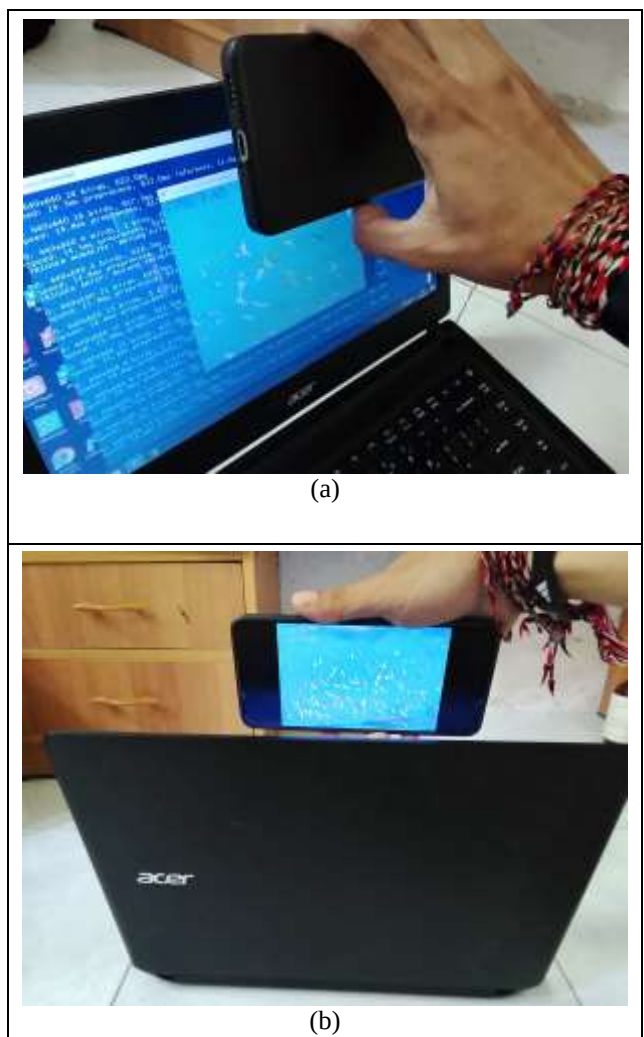
E. Tahapan V: Pengulangan dan Terminasi Sistem (*Graceful Shutdown*)

Setelah mengeksekusi aktuator, program mengevaluasi ketersediaan instruksi interupsi pengguna (*keyboard interrupt*) berupa tekanan tombol 'q'. Apabila tidak ada instruksi terminasi, program akan kembali berulang ke Tahapan II untuk menangkap *frame* berikutnya secara berkesinambungan (*infinite loop*). Sebaliknya, jika interupsi diterima, sistem akan melakukan penghentian secara aman (*graceful shutdown*), membebaskan alokasi memori, dan mengakhiri seluruh proses inspeksi.

IV. HASIL DAN PEMBAHASAN

A. Akurasi Deteksi Visual

Pengujian akurasi fungsionalitas algoritma dilakukan melalui metode simulasi fisik. Stimulus visual diberikan dengan menghadapkan layar gawai pintar (*smartphone*) yang menampilkan citra burung ke depan sensor *webcam* bawaan perangkat komputasi pada jarak kurang lebih 10 cm, sebagaimana ditunjukkan pada Gambar 2.



Gambar 2: Setup pengujian fisik menggunakan simulasi layar gawai; (a) Tampak depan, (b) Tampak belakang



Gawai tersebut kemudian digerakkan secara lateral untuk merepresentasikan manuver burung yang sedang terbang bergerak. Hasil inferensi logikal menunjukkan bahwa jaringan saraf tiruan YOLOv8n mampu mengisolasi anatomi objek kelas burung (*bird*) dengan tingkat presisi probabilitas (*confidence score*) yang berfluktuasi secara stabil pada rentang 0,51 hingga 0,75 (51%–75%). Hasil visualisasi *bounding box* yang berhasil mengisolasi objek tersebut ditampilkan pada Gambar 3. Rentang pengukuran empiris ini mendemonstrasikan bahwa algoritma secara konsisten berhasil melampaui parameter *Confidence Threshold* (0,5), yang mana menjamin bahwa sinyal pemicu hanya akan diaktifkan secara valid tanpa memicu rentetan *false positive* (alarm palsu) yang dapat mengganggu lalu lintas udara.



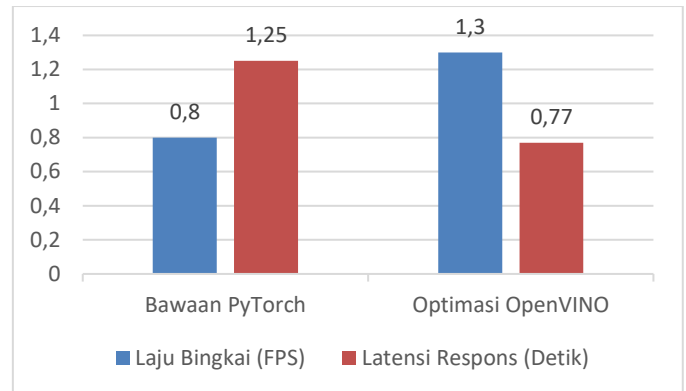
Gambar 3: Hasil deteksi visual YOLOv8n pada objek burung

B. Analisis Kinerja Komputasi (Frame Per Second)

Evaluasi laju *throughput* pemrosesan merupakan metrik paling krusial. Hal ini dikarenakan inferensi *Edge AI* dieksekusi murni oleh CPU perangkat (tanpa dukungan akselerator *Graphics Processing Unit* terdedikasi). Pada fase awal pengujian menggunakan model dasar berekstensi .pt dari PyTorch, sistem mengalami *bottleneck* komputasi dengan mencetak laju pemrosesan matriks rata-rata di angka 0,80 FPS. Hal ini mengindikasikan adanya latensi waktu respons sebesar 1,25 detik untuk mengeksekusi satu bingkai tangkapan.

Sebagai intervensi teknis, model dikonversi ke dalam format *Intermediate Representation* menggunakan instruksi set *framework* Intel OpenVINO. Proses rekayasa perangkat lunak ini secara efisien mendistribusikan beban kalkulasi kecerdasan buatan secara optimal pada arsitektur perangkat keras lokal, menghasilkan peningkatan performa (*Frame Per Second*) yang signifikan menuju laju stabil 1,30 FPS, sebagaimana divisualisasikan pada Gambar 4. Meskipun laju FPS ini tergolong rendah jika dibandingkan dengan perangkat GPU terdedikasi, hasil ini menunjukkan kelayakan operasional untuk aplikasi pemantauan satwa pada perangkat komputasi hemat daya [3], [20], [21]. Akselerasi *throughput* ini memangkas nilai latensi sistemik menjadi hanya 0,77 detik. Berdasarkan parameter rekayasa, jeda latensi 0,77 detik ini menunjukkan

kemampuan sistem dalam merespons stimulus visual pada rentang waktu sub-detik (*sub-second latency*), yang secara teoretis mencukupi untuk memicu aktuator sebelum pergerakan satwa mencapai fase kritis di jalur manuver pesawat.



Gambar 4: Perbandingan kinerja komputasi model

C. Pengujian Logika Pemicu Aktuator (Audio Trigger)

Fungsionalitas subsistem tertutup divalidasi dengan memantau respons *output* terhadap *input* deteksi. Tinjauan empiris terhadap log aktivitas pada terminal komputasi, sebagaimana ditampilkan pada Gambar 5, mengonfirmasi bahwa ketika sistem mendeteksi burung secara visual, *controller* merespons secara linier dengan memunculkan status "TRIGGER AKTIF". Pada fase ini, modul secara otonom memutar berkas audio alarm. Begitu subjek burung dihilangkan dari jangkauan kamera, sistem secara instan mendeteksi perubahan *state* dan mengeksekusi status "TRIGGER NONAKTIF" untuk membungkam sistem audio. Reaksi pemutusan akustik seketika ini membuktikan bahwa sistem tidak lagi mengalami *stuck* dan telah berhasil memitigasi efek habituasi secara teknis.

```
0: 640x640 10 birds, 955.7ms
Speed: 24.8ms preprocess, 955.7ms inference, 62.2ms postprocess
TRIGGER AKTIF: Burung terdeteksi! Mengaktifkan speaker.

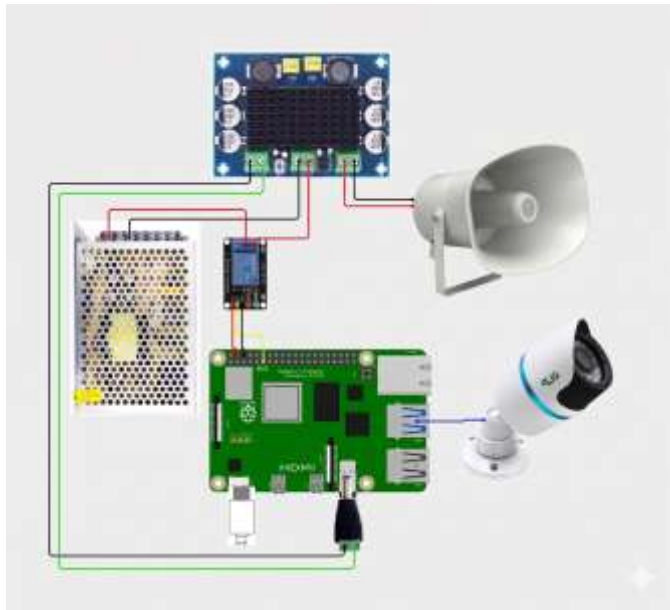
0: 640x640 1 person, 1 suitcase, 1967.4ms
Speed: 24.6ms preprocess, 1967.4ms inference, 14.1ms postprocess
TRIGGER NONAKTIF: Burung hilang. Mematikan speaker.
```

Gambar 5: Log terminal respons sistem terhadap perubahan lingkungan visual

D. Rekomendasi Efisiensi Daya Skala Industri (Industrial Grade)

Berdasarkan validasi prototipe lunak, implementasi riil untuk zona *airside* menuntut arsitektur fisik *industrial grade*. Rancangan arsitektur fisik berskala industri ini divisualisasikan pada Gambar 6 untuk memberikan gambaran integrasi perangkat keras di lapangan. Sistem diintegrasikan ke dalam arsitektur tertanam menggunakan *Single Board Computer* (Raspberry Pi 4) dan modul *Relay 5V*. *Relay* berfungsi sebagai saklar elektronik pintar (*power management*) yang dikendalikan melalui pin GPIO mikrokontroler untuk memutus dan menyambungkan arus listrik menuju *Audio Amplifier*. Desain topologi elektrik ini memastikan *amplifier* tidak menyala terus-menerus selama 24 jam, sehingga secara signifikan mereduksi konsumsi energi dan menghilangkan *noise* statis saat sistem dalam kondisi *standby*. Pengaturan

siklus tugas (*duty cycle*) dan strategi manajemen daya pada perangkat sensor nirkabel sangat penting untuk memperpanjang durasi penyebaran sistem di lapangan [3].



Gambar 6: Rancangan arsitektur perangkat keras skala industri

V. KESIMPULAN

Penelitian ini telah berhasil merancang dan membangun purwarupa perangkat lunak *Smart Bird Deterrent* berbasis *Computer Vision* menggunakan algoritma YOLOv8 Nano. Berdasarkan pengujian kinerja, algoritma secara konsisten mendeteksi objek burung secara *real-time* dengan rentang akurasi probabilitas 51% hingga 75%, melampaui parameter *confidence threshold* 0,5 yang ditetapkan. Optimasi model komputasi menggunakan format *Intermediate Representation* dari Intel OpenVINO terbukti esensial, sukses mendongkrak kecepatan pemrosesan dari 0,80 FPS menjadi 1,30 FPS pada spesifikasi keras lokal kelas bawah (Intel Celeron N3350). Akselerasi *throughput* ini menghasilkan metrik latensi respons sistem sebesar 0,77 detik, yang divalidasi sangat fungsional untuk mengeksekusi logika pemicu (*trigger*) aktuator alarm secara instan dan dinamis ketika mendeteksi kehadiran target visual.

Sistem kendali secara teknis terbukti andal dalam memutuskan dan menyambung pancaran audio secara otonom berdasarkan *feedback* visual, menyelesaikan limitasi pengoperasian tanpa henti pada perangkat *open-loop* konvensional. Mekanisme ini merealisasikan efisiensi daya melalui deteksi objek, yang secara paralel memitigasi potensi terjadinya habituasi satwa. Kebaruan (*novelty*) dari penelitian ini terletak pada integrasi sistem otomasi *Edge AI* yang mampu beroperasi efisien tanpa *Graphics Processing Unit* (GPU) terdedikasi, dipadukan dengan rekayasa penghentian aktuator cerdas sebagai inovasi mitigasi terdesentralisasi bagi keselamatan operasional di area *airside* bandar udara.

Artawan: Rancang Bangun Sistem Prototipe ...

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada manajemen PT Angkasa Pura Indonesia Kantor Cabang Bandar Udara Internasional I Gusti Ngurah Rai, secara khusus kepada *Electronic Services Department* (Unit ELBAN), atas penyediaan fasilitas observasi, dukungan infrastruktur pengujian, serta bimbingan teknis selama pelaksanaan Kerja Praktik. Apresiasi juga disampaikan kepada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Udayana atas dukungan akademis dalam penyelesaian riset ini.

REFERENSI

- [1] F. Di Caprio, A. Sellitto, S. Saputo, M. Guida, and A. Riccio, "A sensitivity analysis of the damage behavior of a leading-edge subject to bird strike," *Applied Sciences*, vol. 10, no. 22, Nov. 2020, doi: 10.3390/app10228187.
- [2] Z. U. Rehman, A. Ghafar, A. Syed, A. Tayab, and M. G. Rabbi, "An Optimized YOLOv8-based Method for Airport Bird Detection: Incorporating ECA Attention, MBC3 Module, and SF-PAN for Improved Accuracy and Speed," *International Journal of Intelligent Systems and Applications*, vol. 17, no. 2, pp. 1–13, Apr. 2025, doi: 10.5815/ijisa.2025.02.01.
- [3] A. Vuillienet, K. E. Jones, and D. Wilson, "Future of Edge AI in biodiversity monitoring," pp. 1–41, Feb. 2026, [Online]. Available: <http://arxiv.org/abs/2602.13496>
- [4] D. Adami, M. O. Ojo, and S. Giordano, "Design, Development and Evaluation of an Intelligent Animal Repelling System for Crop Protection Based on Embedded Edge-AI," *IEEE Access*, vol. 9, pp. 132125–132139, Sep. 2021, doi: 10.1109/ACCESS.2021.3114503.
- [5] H. Sun, Y. Wang, J. Du, and R. Wang, "MFE-YOLO: A Multi-feature Fusion Algorithm for Airport Bird Detection," *ICCK Transactions on Intelligent Systematics*, vol. 2, no. 2, pp. 85–94, May 2025, doi: 10.62762/tis.2025.323887.
- [6] J. Ma, J. Guo, X. Zheng, and C. Fang, "An Improved Bird Detection Method Using Surveillance Videos from Poyang Lake Based on YOLOv8," *Animals*, vol. 14, no. 23, Nov. 2024, doi: 10.3390/ani1423353.
- [7] A. F. Félix-Jiménez, V. S. Sánchez-Lee, H. A. Acuña-Cid, I. Ibarra-Belmonte, E. Arredondo-Morales, and E. Ahumada-Tello, "Integration of YOLOv8 Small and MobileNet V3 Large for Efficient Bird Detection and Classification on Mobile Devices," *AI*, vol. 6, no. 3, Mar. 2025, doi: 10.3390/ai6030057.
- [8] A. Kh Al-Zihairy, A. E. Abdelkareem, and C. Author, "ENHANCED YOLOV8 FOR EDGE COMPUTING: A DEEP LEARNING APPROACH FOR MEMORY OPTIMIZATION," *Iraqi Journal of Information and Communications Technology(Iss)*, vol. 8, no. 1, pp. 23–37, Apr. 2025, doi: 10.31987/ijict.8.1.283.
- [9] I. Nurkarima, H. Priyatman, and B. Kurniawan, "PEMODELAN SISTEM LEVEL AIR HEAD TANK MENGGUNAKAN PENGENDALI PID PADA PEMBANGKIT LISTRIK TENAGA MINI HIDRO MERASAP."
- [10] R. C. Beason, "What Can Birds Hear?," *Proceedings of the 21st Vertebrate Pest Conference*, vol. 21, pp. 92–96, 2004.
- [11] P. E. Schlichting *et al.*, "Efficacy of an acoustic hailing device as an avian dispersal tool," *Wildl. Soc. Bull.*, vol. 41, no. 3, pp. 453–460, Sep. 2017, doi: 10.1002/wsb.797.
- [12] J. I. Ramirez *et al.*, "Applied ecology of fear: A meta-analysis on the potential of facilitating human-wildlife coexistence through nonlethal tools," *Ecological Solutions and Evidence*, vol. 5, no. 2, Mar. 2024, doi: 10.1002/2688-8319.12322.
- [13] C. H. Rankin *et al.*, "Habituation revisited: An updated and revised description of the behavioral characteristics of habituation," *Neurobiol. Learn. Mem.*, vol. 92, no. 2, pp. 135–138, Sep. 2009, doi: 10.1016/j.nlm.2008.09.012.
- [14] Q. Chen *et al.*, "An experimental study of acoustic bird repellents for reducing bird encroachment in pear orchards," *Front. Plant Sci.*, vol. 15, Sep. 2024, doi: 10.3389/fpls.2024.1365275.

p-ISSN:1693 – 2951; e-ISSN: 2503-2372



9 772503 237160

- [15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection." [Online]. Available: <http://pjreddie.com/yolo/>
- [16] Y. Chen and H. Watanabe, "OPTIMIZING YOLOV8N FOR ENHANCED PRECISION IN SMALL OBJECT DETECTION ON CUSTOM DATASETS," *European Journal of Emerging Data Science and Machine Learning*, vol. 1, no. 1, pp. 63–80, Dec. 2024.
- [17] C. G. Raju, A. Devarapalli, R. Mahendran, S. Madhusudan, and O. Prasad, "Optimizing YOLOv8: OpenVINO standard quantization vs accuracy-controlled for edge deployment," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 40, no. 3, pp. 1567–1575, Dec. 2025, doi: 10.11591/ijeecs.v40.i3.pp1567-1575.
- [18] Intel Corporation, "Conventional AI Workflow," OpenVINO™ Documentation. Accessed: Jun. 16, 2026. [Online]. Available: <https://docs.openvino.ai/2026/openvino-workflow.html>
- [19] Ultralytics, "Brief summary of YOLOv8 model structure," GitHub (ultralytics/ultralytics). Accessed: Jun. 16, 2026. [Online]. Available: <https://github.com/ultralytics/ultralytics/issues/189>
- [20] O. Shalash *et al.*, "Benchmarking YOLOv8–YOLOv12 for Real-Time Object Detection on Single-Board Computers," *Preprints.org*, May 2026, doi: 10.20944/preprints202605.0936.v1.
- [21] Y. H. Hsieh, C. C. Kao, C. H. Lai, K. P. Lin, S. Y. Yang, and S. M. Yuan, "Low-FPS Multi-Object Multi-Camera Tracking via Deep Learning," *Electronics (Basel)*, vol. 14, no. 7, Apr. 2025, doi: 10.3390/electronics14071373.