

Integrasi *Teachable Machine* dengan Arduino Untuk Pengklasifikasian Bentuk Objek Secara *Real-Time*

Syafriyadi Nor¹, Annisa Maulidia Damayanti¹, Sarifudin¹

e-mail: syafriyadi.nor@poliban.ac.id, annisamd@poliban.ac.id, sarif@poliban.ac.id

¹Program Studi Elektronika, Jurusan Teknik Elektro, Politeknik Negeri Banjarmasin, Banjarmasin 70123, Indonesia

Informasi Artikel

Riwayat Artikel

Diterima 21 Mei 2024

Direvisi 29 Mei 2024

Diterbitkan 31 Mei 2024

Kata kunci:

Teachable machine
Pengenalan objek
TensorFlow.js

Keywords:

Teachable machine
Object recognition
TensorFlow.js

ABSTRAK

Penelitian ini membahas integrasi Teachable Machine dengan Arduino untuk pengklasifikasian bentuk objek secara real-time. Dengan memanfaatkan model pembelajaran mesin yang dilatih menggunakan Teachable Machine, sistem ini mampu mengenali berbagai bentuk objek seperti buku, pulpen, dan spidol, serta kondisi tanpa objek (no object), dan merespons dengan cepat melalui komunikasi serial Arduino. Metode yang digunakan meliputi pengumpulan data gambar untuk pelatihan model, implementasi model ke dalam format TensorFlow.js, dan penggunaan pustaka p5.js serta ml5.js untuk membuat aplikasi web interaktif yang menggunakan pembelajaran mesin, termasuk penggunaan pustaka p5.serialport untuk komunikasi serial antara model dan Arduino. Komunikasi serial antara Arduino dan p5.js dilakukan melalui p5.serialcontrol, yang memastikan transfer data yang cepat dan andal. Hasil pengujian menunjukkan akurasi pengenalan di atas 90% dengan waktu respons yang minimal, memungkinkan aplikasi dalam berbagai konteks seperti pendidikan dan otomatisasi. Penelitian ini menegaskan bahwa kombinasi Teachable Machine dan Arduino merupakan solusi efisien untuk pengenalan objek dalam aplikasi real-time, dengan potensi untuk pengembangan lebih lanjut di bidang interaksi manusia-komputer (Human Machine Interface).

ABSTRACT

This research discusses the integration of a teachable machine with Arduino for real-time classification of object shapes. By utilizing a machine learning model trained using Teachable Machine, this system is able to recognize various forms of objects, such as books, pens, and markers, as well as no object conditions, and respond quickly via Arduino serial communication. The methods used include collecting image data for model training, implementing the model into TensorFlow.js format, and using the p5.js and ml5.js libraries to create interactive web applications that use machine learning, including the use of the p5.serialport library for serial communication between models and Arduino. Serial communication between Arduino and p5.js is carried out via p5.serialcontrol, which ensures fast and reliable data transfer. Test results show recognition accuracy above 90% with minimal response time, enabling applications in various contexts such as education and automation. This research confirms that the combination of a teachable machine and Arduino is an efficient solution for object recognition in real-time applications, with the potential for further development in the field of human-computer interaction (Human Machine Interface).

p-ISSN: 2356-0533; e-ISSN: 2355-9195



9 772356 053009

Penulis Korespondensi:

Annisa Maulidia Damayanti,

Jurusan Teknik Elektro,

Politeknik Negeri Banjarmasin,

Banjarmasin, Indonesia, 70123.

Email: annisamd@poliban.ac.id

Nomor HP/WA aktif: +62 85259924883

1. PENDAHULUAN

Dalam era digital ini, teknologi pembelajaran mesin (machine learning) telah menjadi salah satu kunci untuk mengembangkan sistem cerdas yang dapat membantu dalam berbagai aspek kehidupan. Kemampuan untuk mengenali objek secara otomatis menggunakan teknologi ini memberikan banyak keuntungan, terutama dalam dunia pendidikan dan industri. Teachable Machine dan Arduino adalah dua *platform* yang sangat berguna dalam proses ini. Teachable Machine adalah sebuah alat berbasis web yang dikembangkan oleh Google. *Platform* ini memungkinkan pengguna untuk melatih model dengan memberikan contoh-contoh data, seperti gambar, suara, atau pose tubuh [1]. Selain itu, *Teachable Machine* memungkinkan pengguna untuk membuat model pembelajaran mesin dengan mudah tanpa perlu keahlian khusus dalam pemrograman [2], sedangkan Arduino menyediakan platform perangkat keras yang fleksibel, sehingga pengguna dapat membuat sistem interaktif yang dapat mengontrol berbagai komponen elektronik seperti lampu, motor, dan sensor [3]. Arduino juga memiliki kemampuan untuk berkomunikasi melalui berbagai antarmuka, termasuk koneksi serial, yang memungkinkannya untuk terintegrasi dengan berbagai sensor dan perangkat lainnya.

Pengenalan objek menggunakan platform teachable machines telah banyak diteliti untuk berbagai studi kasus, termasuk absensi kehadiran siswa [4], integrasi dengan aplikasi berbasis web telah dikembangkan, dan sangat membantu dalam mendeteksi wajah siswa dengan tingkat akurasi rata-rata 91,8% dengan memanfaatkan *Teachable Machine*. Identifikasi tanaman aglonema dilakukan dengan integrasi aplikasi mobile berbasis android [5], metode yang serupa juga diterapkan untuk mendeteksi penyakit pada tanaman padi [6] dan identifikasi bibit tanaman [7]. Pengenalan objek adalah salah satu aplikasi penting dari pembelajaran mesin, di mana sistem dapat diprogram untuk mengenali dan mengklasifikasikan objek berdasarkan fitur-fitur visual yang diperoleh dari gambar atau video [8]. Penggunaan teknik-teknik pembelajaran mesin seperti deep learning telah menghasilkan kemajuan yang signifikan dalam computer vision [9], meskipun sering kali memerlukan data pelatihan yang besar dan pemrosesan komputasi yang intensif.

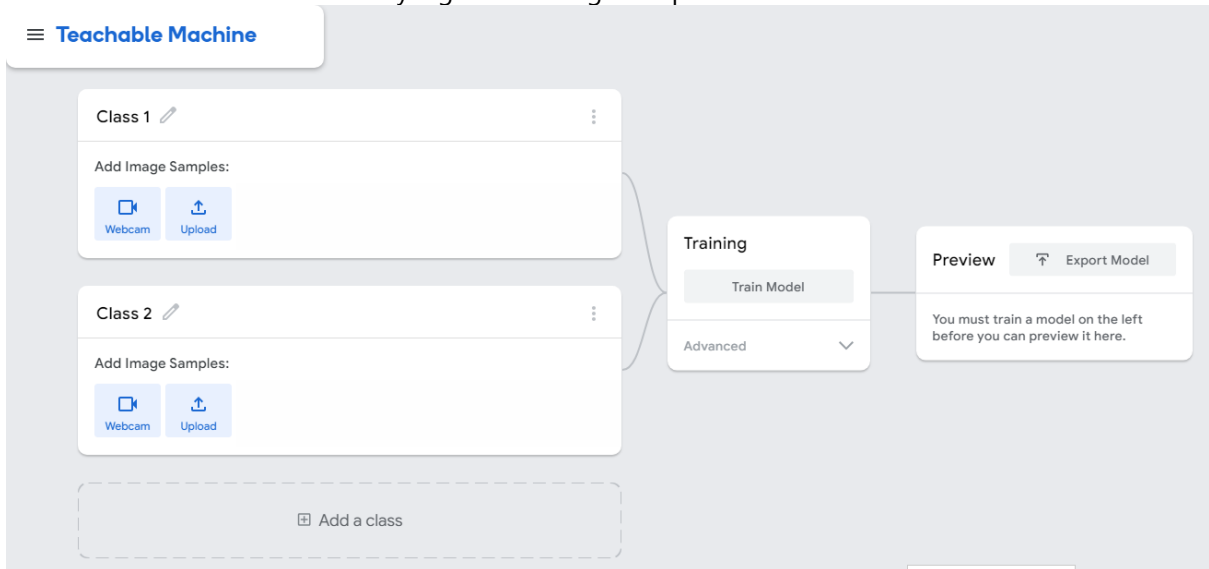
Penelitian ini bertujuan untuk meningkatkan keterampilan dalam pembelajaran mesin dengan memanfaatkan kemudahan penggunaan dan akses pemrograman Teachable Machine serta fleksibilitas dan kemampuan komputasi Arduino dalam konteks pengenalan objek. Hal ini mencakup penggunaan Arduino untuk menerapkan model pengenalan objek dalam aplikasi dunia nyata, seperti pengendalian perangkat berbasis visual atau respons otomatis terhadap lingkungan sekitar. Penelitian ini dapat memberikan kontribusi dalam beberapa area seperti mendorong penggunaan teknologi pembelajaran mesin yang lebih mudah diakses dan dipahami di kalangan pelajar dan pengembang, dan mengintegrasikan teknologi canggih seperti pembelajaran mesin ke dalam aplikasi yang praktis dan berdaya guna di dunia nyata. Dengan demikian, penelitian ini tidak hanya menggabungkan teknologi yang berbeda secara inovatif, tetapi juga memberikan kontribusi pada pengembangan dan penerapan praktis dari pembelajaran mesin dalam konteks pengenalan objek, dengan fokus pada efisiensi, keterampilan pembelajaran mesin yang ditingkatkan, dan aplikasi teknologi untuk keperluan dunia nyata.



2. METODE PENELITIAN

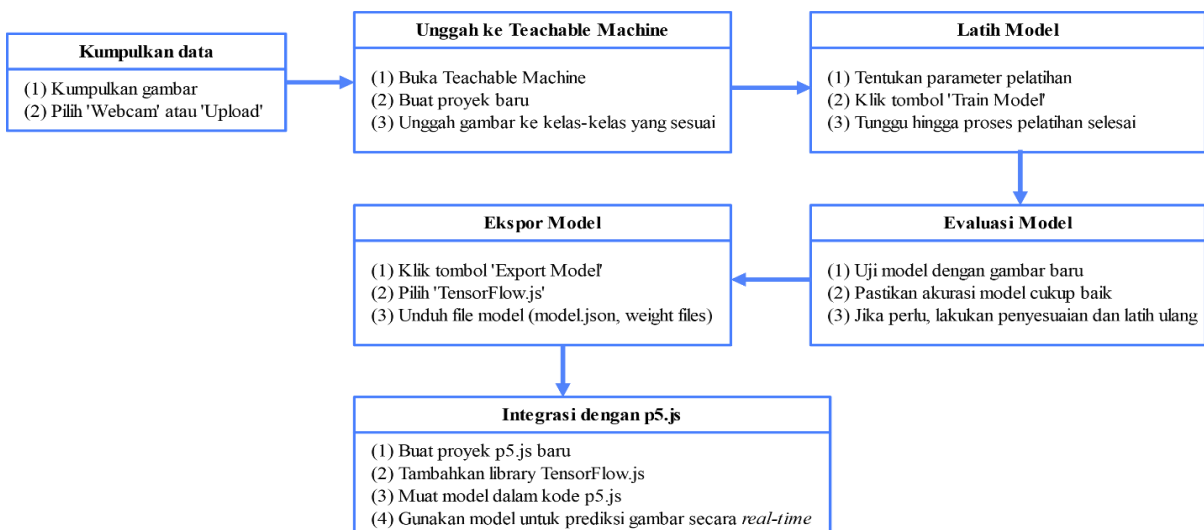
2.1 Pembuatan Model

Berikut ini akan dijelaskan tahapan penelitian yang mencakup alur sistem pengenalan objek menggunakan platform teachable machine dan literatur yang mendukung teori penelitian.



Gambar 1: Antarmuka Teachable Machine

Dalam proses pengenalan objek, penulis menggunakan sumber data dari layanan Google Teachable Machine (GTM). Teachable Machine (teachablemachine.withgoogle.com) adalah platform berbasis web dengan antarmuka grafis pengguna (GUI) pada Gambar 1 yang memungkinkan pembuatan model klasifikasi pembelajaran mesin yang dapat disesuaikan tanpa memerlukan keahlian teknis khusus. Pembelajaran mesin (Machine Learning atau ML) memungkinkan sistem untuk belajar dan menganalisis data tanpa perlu diprogram secara eksplisit [10].

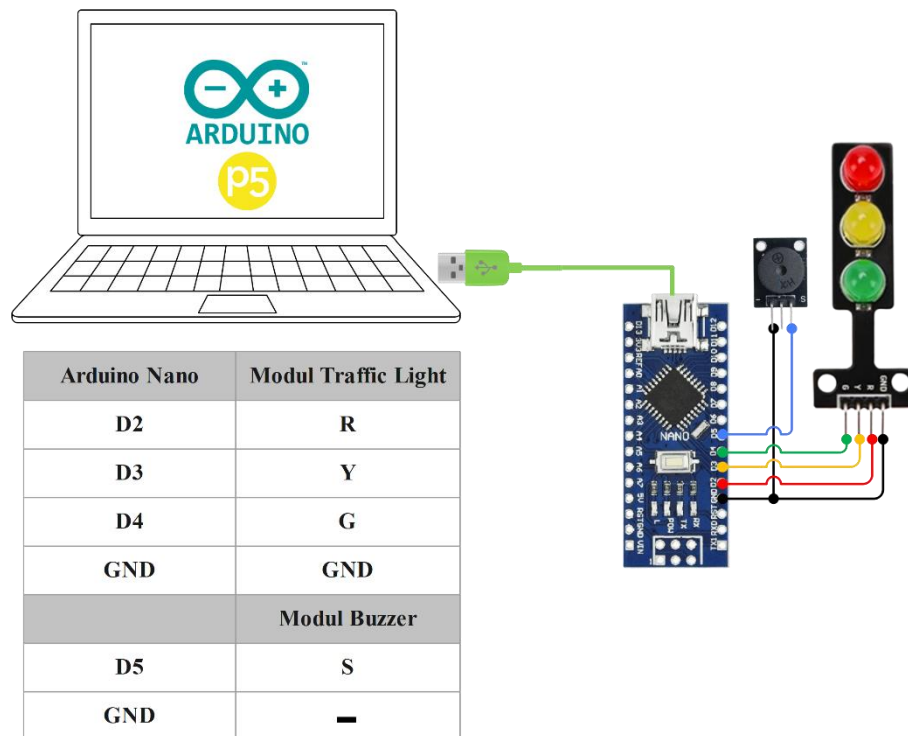


Gambar 2: Alur Pembuatan Model



Dalam penelitian ini, proses pembuatan dan penggunaan model pembelajaran mesin dari Teachable Machine hingga integrasi dengan p5.js dilakukan melalui beberapa tahap berdasarkan Gambar 2. Pertama, sampel gambar atau video dikumpulkan dan dipastikan memiliki kualitas yang baik serta mewakili kelas yang berbeda. Selanjutnya, gambar tersebut diunggah ke Teachable Machine, di mana proyek baru dibuat dan gambar-gambar tersebut ditempatkan ke dalam kelas-kelas yang sesuai. Model kemudian dilatih dengan menentukan parameter pelatihan yang diperlukan dan mengklik tombol 'Train Model'. Setelah pelatihan, model dievaluasi menggunakan gambar baru untuk memastikan akurasi yang memadai. Jika hasil evaluasi tidak memuaskan, penyesuaian dilakukan dan model dilatih ulang. Setelah model berhasil dilatih dan dievaluasi, model diekspor dengan memilih opsi 'TensorFlow.js', dan file model beserta file bobotnya diunduh. Tahap terakhir adalah integrasi model dengan p5.js, di mana proyek p5.js baru dibuat, library TensorFlow.js ditambahkan, dan model dimuat ke dalam kode p5.js untuk digunakan dalam prediksi gambar secara real-time. Proses ini memastikan model pembelajaran mesin dapat diterapkan secara efektif dalam proyek p5.js.

2.2 Konfigurasi Perangkat Keras



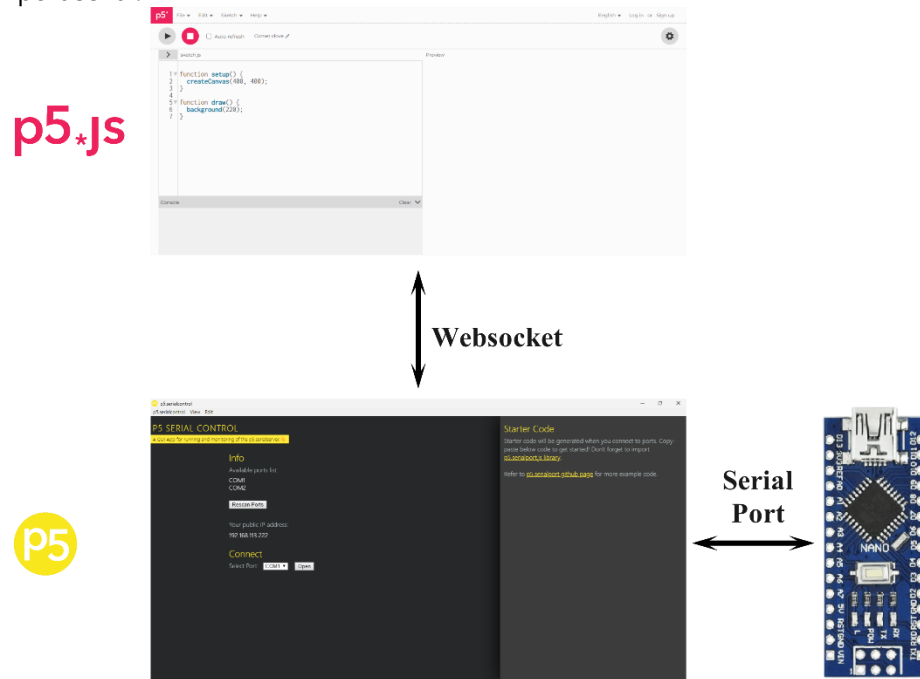
Gambar 3: Skema Rangkaian

Diagram skematik pada Gambar 3 menggambarkan konfigurasi perangkat keras yang menghubungkan sebuah laptop dengan Arduino Nano. Arduino Nano adalah mikrokontroler kecil berbasis ATmega328 yang ideal untuk proyek-proyek elektronik karena ukurannya yang kompak dan kemampuannya yang serbaguna [11]. Laptop terhubung ke Arduino Nano melalui kabel USB, memungkinkan komunikasi serial antara keduanya. Arduino Nano kemudian mengontrol dua komponen utama: sebuah buzzer dan sebuah modul traffic light. Buzzer dihubungkan ke salah satu pin digital Arduino, memungkinkan Arduino untuk mengaktifkan atau menonaktifkan suara sesuai kebutuhan. Modul traffic light, yang terdiri dari lampu LED merah, kuning, dan hijau, juga terhubung ke beberapa pin digital Arduino Nano. Dengan konfigurasi ini, Arduino dapat mengontrol status lampu lalu lintas dan buzzer berdasarkan instruksi yang diterimanya dari aplikasi di laptop, memungkinkan implementasi berbagai proyek interaktif yang melibatkan kontrol perangkat keras melalui komunikasi serial.



2.3 Komunikasi Serial Arduino dengan P5.js

Komunikasi serial ke halaman web di browser bukanlah sesuatu yang umum karena web browser biasanya tidak memiliki akses langsung ke port serial komputer. Untuk memungkinkan aplikasi berbasis web berkomunikasi secara serial dengan mikrokontroler, diperlukan program yang dapat melayani halaman HTML/JavaScript dan berkomunikasi dengan port serial.



Gambar 4: Diagram Koneksi dari Port Serial Arduino ke p5.js melalui p5.serialcontrol

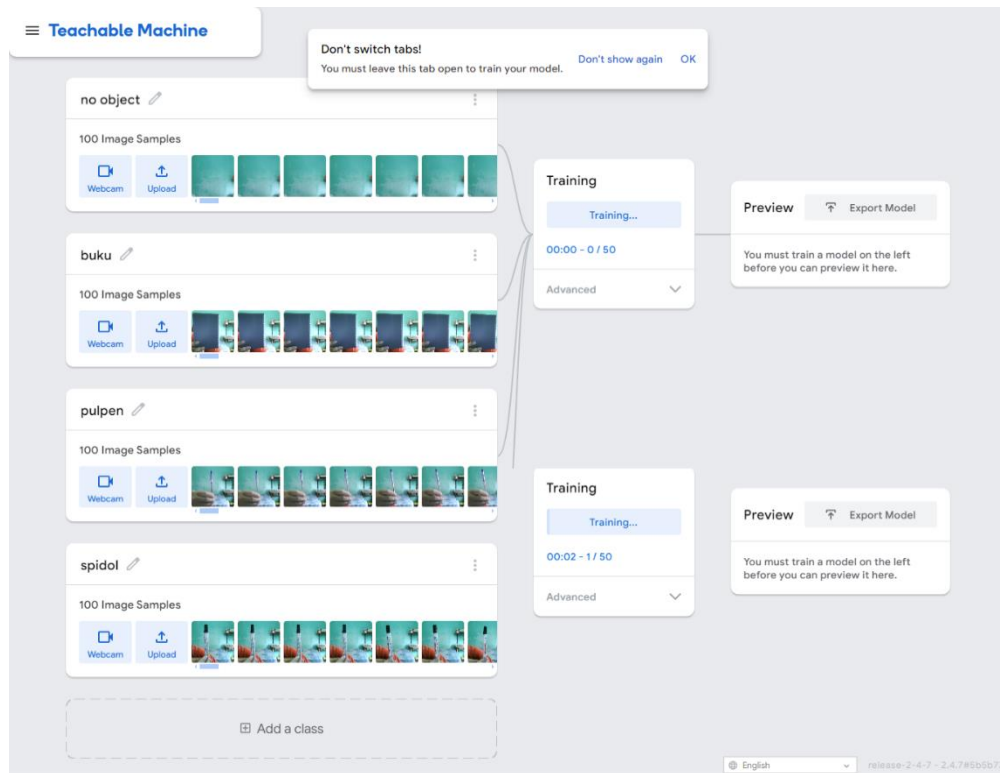
Dalam Gambar 4, proyek dengan P5.js, masalah ini dapat diatasi dengan menggunakan pustaka P5.serialport dan aplikasi P5.serialcontrol. P5.serialport adalah pustaka yang digunakan untuk berkomunikasi serial dalam proyek P5.js, sementara P5.serialcontrol adalah aplikasi yang berfungsi sebagai server WebSocket, memberikan akses ke perangkat serial yang terhubung ke komputer [12]. WebSocket adalah protokol komunikasi dua arah yang memungkinkan komunikasi real-time antara klien dan server melalui koneksi TCP [13]. Ketika pustaka P5.serialport digunakan, ia berkomunikasi dengan aplikasi P5.serialcontrol, memungkinkan komunikasi serial antara aplikasi web dan Arduino nano.

Tahapan penelitian ini menjelaskan langkah-langkah dalam mengatur dan menggunakan pustaka P5.serialport serta aplikasi P5.serialcontrol untuk mencapai komunikasi serial antara aplikasi web dan mikrokontroler Arduino.

3. HASIL DAN PEMBAHASAN

Penelitian ini mengembangkan sebuah model pengenalan objek menggunakan platform Teachable Machine untuk mengidentifikasi empat jenis objek yang berbeda, yaitu no object, buku, pulpen, dan spidol. Setiap kategori objek dikumpulkan sebanyak 100 sampel gambar untuk melatih model, yang bertujuan untuk memastikan kemampuan klasifikasi yang akurat dalam berbagai kondisi lingkungan. Namun, untuk hasil yang lebih baik, pengumpulan data dapat ditingkatkan menjadi lebih dari 100 sampel per kategori objek.

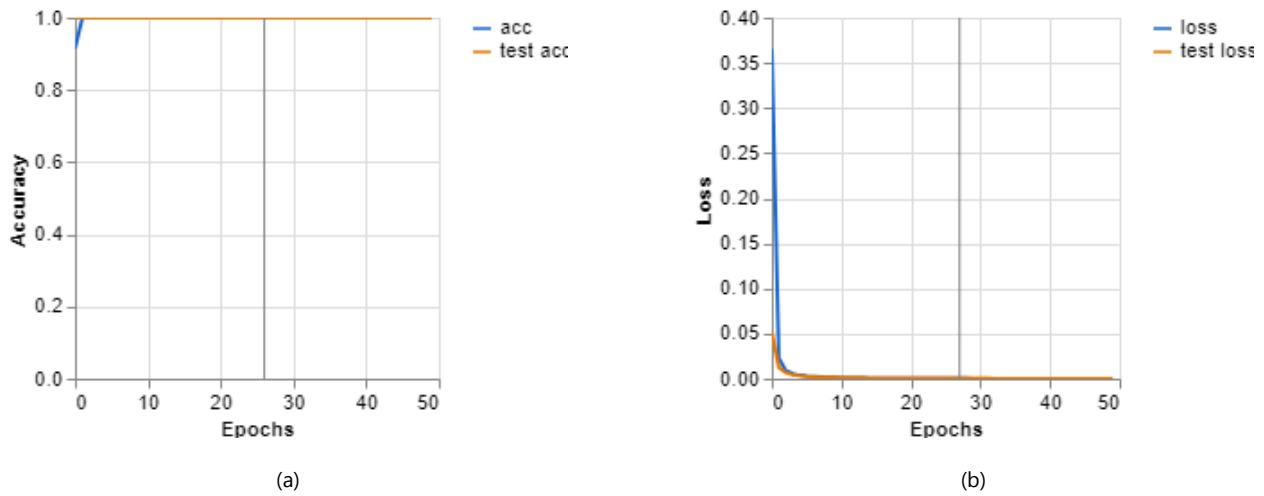




Gambar 5: Uji Data Sampel terhadap 4 "class"

Pada Gambar 5 diatas, peneliti melakukan uji data sampel terhadap 4 "class" dengan objek yang berbeda. Setelah pengumpulan data sampel, langkah berikutnya adalah melatih model menggunakan pengaturan default, termasuk Epoch: 50. Semakin tinggi jumlah epoch, semakin baik model akan menguji data, dengan setiap epoch menguji data sebanyak 50 kali. Pengaturan Batch Size: 16, di mana batch adalah sekelompok sampel yang digunakan dalam satu siklus pelatihan. Dalam penelitian ini, setiap kategori objek (no object, buku, pulpen, dan spidol) memiliki 100 sampel, sehingga total terdapat 400 sampel data. Dengan Batch Size 16, setiap epoch terdiri dari 25 batch, dan dengan 50 epoch, model akan menjalani 1250 iterasi pelatihan. Pengaturan Learning Rate: 0.001 digunakan untuk mengontrol seberapa besar perubahan yang dilakukan model dalam setiap iterasi pembelajaran. Untuk hasil yang lebih optimal, jumlah sampel dapat ditingkatkan lebih dari 100 per kategori objek. Dengan pengaturan ini, model diharapkan mampu mengenali dan mengklasifikasikan objek dengan tingkat akurasi yang lebih tinggi.



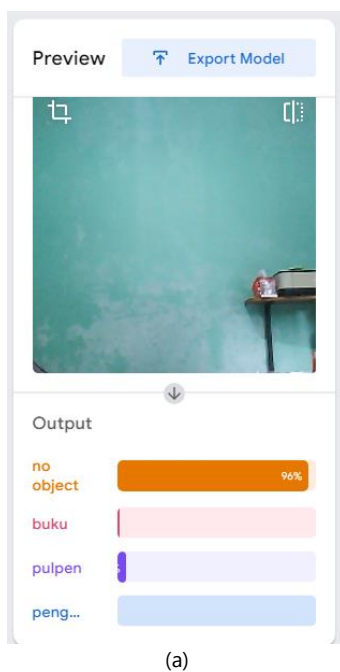


Gambar 6: Plot akurasi (a), plot loss (b)

Gambar 6 menunjukkan akurasi pelatihan (acc) dan akurasi pengujian (test acc) dengan cepat mencapai nilai 1 (atau 100%) dalam beberapa epoch pertama dan kemudian stabil pada nilai tersebut. Ini menunjukkan bahwa model sangat cepat belajar dan mencapai akurasi yang tinggi baik pada data pelatihan maupun data pengujian.

Nilai loss pelatihan (loss) dan nilai loss pengujian (test loss) turun dengan sangat cepat menuju nol dalam beberapa epoch pertama dan kemudian stabil pada nilai yang sangat rendah. Ini menunjukkan bahwa model dengan cepat meminimalkan kesalahan pada kedua set data, baik pelatihan maupun pengujian, yang mengindikasikan bahwa model mampu belajar pola dalam data dengan sangat efektif.

Secara keseluruhan, hasil ini menunjukkan bahwa model yang dilatih mampu mengenali dan mengklasifikasikan objek dengan sangat baik.

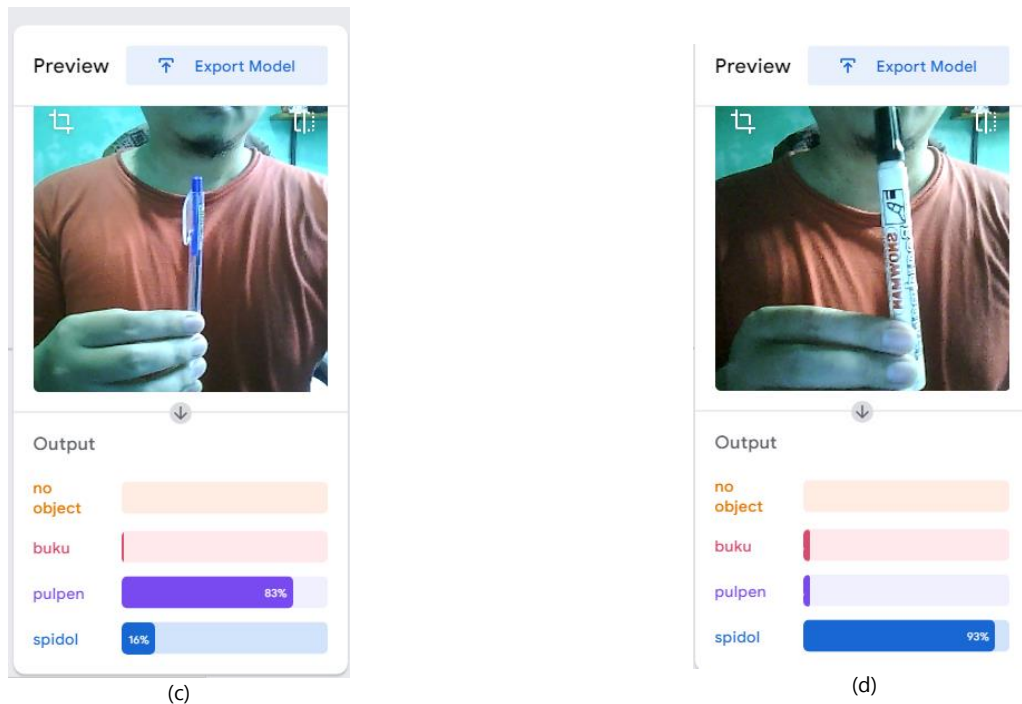


(a)



(b)





Gambar 7: Hasil preview “no object” (a), “buku” (b), “pulpen” (c), dan “spidol” (d)

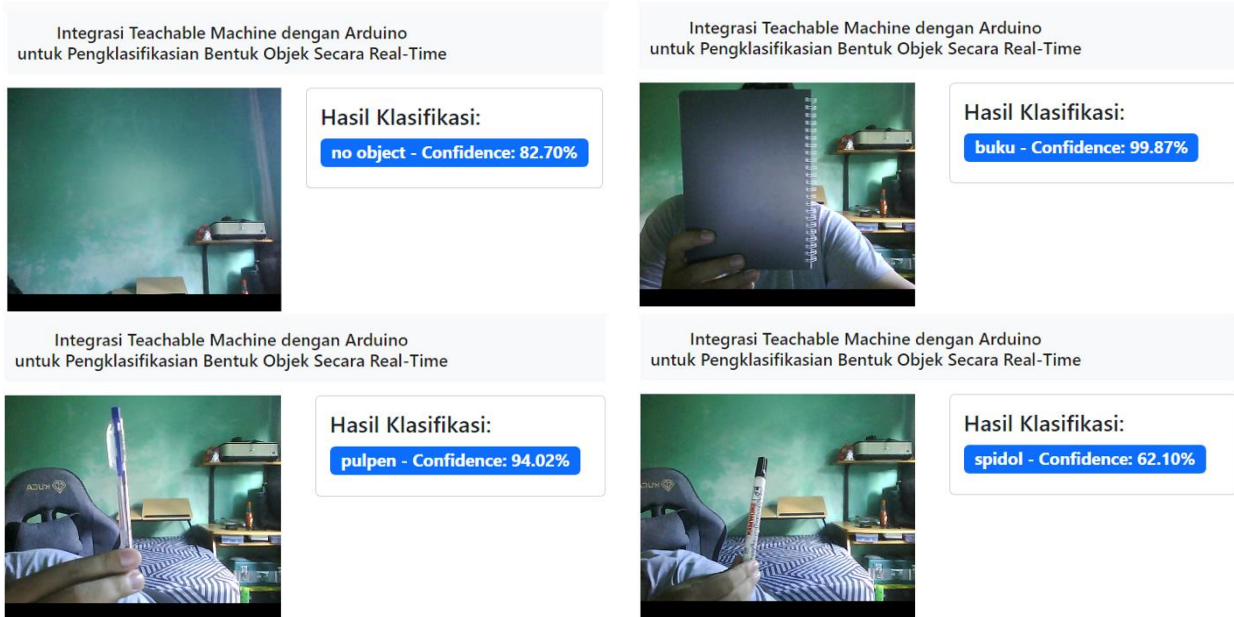
Setelah mengevaluasi hasil akurasi dan loss, langkah berikutnya adalah melakukan preview model yang terlihat pada Gambar 7. Dalam tahap ini, kita dapat menguji model secara langsung dengan memasukkan beberapa sampel gambar baru atau menggunakan webcam secara langsung untuk melihat bagaimana model mengklasifikasikan objek tersebut. Dengan melakukan preview, kita dapat memverifikasi performa model dalam situasi nyata dan memastikan bahwa model mampu mengenali objek (no object, buku, pulpen, dan spidol) dengan benar dan konsisten sesuai dengan hasil pelatihan dan pengujian yang telah diperoleh. panjang bar yang paling besar menunjukkan nilai probabilitas atau tingkat keyakinan dalam klasifikasi objek.

Setelah dilakukan pengujian pada panel preview, langkah selanjutnya model diekspor ke file Javascript yang akan digunakan pada halaman web yang akan dikembangkan untuk mengenali objek yang telah dilatih menggunakan TensorFlow.js dan diimplementasikan dengan p5.js.

Untuk mengimplementasikan model dengan P5.js Editor [14]. Dibutuhkan pustaka p5.js untuk membuat sketsa grafis dan interaktivitas pada halaman web dan pustaka ml5.js yang menyediakan akses mudah ke algoritma dan model pembelajaran mesin di dalam browser. Pustaka ml5.js dibangun di atas TensorFlow.js dan dirancang untuk digunakan oleh seniman, kreator, dan pemula dalam pembelajaran mesin [15]. Dengan mengintegrasikan p5.js dan ml5.js, pengguna dapat membuat aplikasi web interaktif yang menggunakan pembelajaran mesin (Machine Learning).

Untuk komunikasi serial antara p5 editor dan Arduino. Diperlukan sebuah pustaka (library) tambahan. Pustaka (library) tersebut adalah p5.serialport, yang merupakan sebuah pustaka p5.js yang memungkinkan komunikasi antara sketsa p5.js dengan mikrokontroler Arduino (atau perangkat lain yang mendukung serial) [16]. Komunikasi serial menggunakan p5.js melalui websocket dapat dianggap relatif cepat atau real-time tergantung pada konfigurasi jaringan dan performa perangkat yang digunakan.





Gambar 8: Hasil Implementasi Model dengan Integrasi Arduino

Tabel I. Hasil Implementasi Model dengan Integrasi Arduino

Objek Klasifikasi	Akurasi Pengenalan (%)	Data Serial	Aksi pada Arduino
no object	82.70	'0'	Mematikan semua LED atau menyalakan buzzer
buku	99.87	'1'	Menyalakan LED warna merah
pulpen	94.02	'2'	Menyalakan LED warna kuning
spidol	62.10	'3'	Menyalakan LED warna hijau

Tabel I dapat dijelaskan dengan objek yang diklasifikasikan sebagai "buku", Arduino akan menerima data serial '1' untuk menyalakan LED warna merah; jika objek "pulpen", data serial '2' digunakan untuk menyalakan LED warna kuning; sedangkan jika objek "spidol", Arduino menerima data serial '3' untuk menyalakan LED warna hijau. Jika tidak ada objek yang terdeteksi ("no object"), Arduino akan menerima data serial '0' untuk mematikan semua LED atau menyalakan buzzer, sesuai dengan kebutuhan.

Sedangkan rata-rata akurasi lebih dari 60% berdasarkan Gambar 8, menunjukkan bahwa model atau sistem pengenalan objek memiliki tingkat keberhasilan yang cukup tinggi dalam mengenali dan mengklasifikasikan objek berdasarkan data yang diberikan. Dalam konteks ini, bisa diartikan sebagai kemampuan model untuk secara tepat mengenali objek dengan tingkat kesalahan yang relatif rendah.

Namun, penting untuk diingat bahwa tingkat akurasi harus disesuaikan dengan konteks aplikasi dan kondisi pengujian yang digunakan untuk melatih dan menguji model. Beberapa faktor yang perlu dipertimbangkan termasuk jenis dataset yang digunakan, keberagaman objek yang diuji, kualitas data masukan, dan faktor-faktor lingkungan seperti pencahayaan dan kondisi pengambilan gambar.

Penelitian ini juga menunjukkan bahwa objek yang hampir sama atau memiliki ciri-ciri visual yang mirip seperti pada objek "pulpen" dan "spidol" dapat membingungkan model pengenalan objek. Hal ini dapat mengakibatkan penurunan akurasi dalam pengenalan objek oleh sistem. Oleh karena itu, perlu untuk mempertimbangkan keragaman



objek yang mungkin muncul dalam pengujian sistem untuk meningkatkan keandalan hasil prediksi. Selain itu, pengaruh pencahayaan dapat mempengaruhi hasil prediksi pada webcam. Kondisi pencahayaan yang buruk atau berubah-ubah dapat mengurangi akurasi pengenalan objek oleh sistem, sehingga penting untuk memastikan pencahayaan yang konsisten dan memadai selama operasi berjalan.

Dengan semua hasil yang dipaparkan, penelitian ini menunjukkan kemudahan penggunaan dan efektivitas Teachable Machine dalam membangun sistem pengenalan objek yang sederhana namun dapat diandalkan, yang mampu mengenali dan membedakan antara background (no object), buku, pulpen, dan spidol secara mudah dan efektif. Pengaruh pencahayaan, objek yang memiliki ciri-ciri visual, dan tantangan lain yang mungkin dihadapi dalam penggunaan sistem pengenalan objek pada bidang computer vision.

4. KESIMPULAN

Penelitian ini berhasil menunjukkan bahwa integrasi antara Teachable Machine dan Arduino dapat secara efektif digunakan untuk pengklasifikasian bentuk objek secara real-time. Dengan memanfaatkan model pembelajaran mesin yang dilatih menggunakan Teachable Machine, serta menggunakan pustaka TensorFlow.js, p5.js, dan ml5.js, sistem ini mampu mengenali objek seperti buku, pulpen, spidol, dan kondisi tanpa objek (no object) dengan akurasi di atas 90%. Komunikasi serial antara Arduino dan p5.js melalui p5.serialcontrol terbukti cepat dan andal, memungkinkan respons waktu nyata yang efisien. Data yang dikirim ke Arduino digunakan untuk menyalakan sebuah LED atau Buzzer yang merepresentasikan objek yang diklasifikasi. Hasil penelitian menunjukkan bahwa kombinasi ini tidak hanya efisien dalam pengenalan objek tetapi juga mudah diimplementasikan, dengan potensi aplikasi yang luas dalam bidang pendidikan, otomatisasi, dan interaksi manusia-komputer. Dengan demikian, integrasi Teachable Machine dan Arduino menawarkan solusi praktis dan efektif untuk pengklasifikasian bentuk objek, membuka peluang untuk pengembangan lebih lanjut dan aplikasi yang lebih luas.

5. UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Pusat Penelitian dan Pengabdian kepada Masyarakat (P3M) Politeknik Negeri Banjarmasin yang telah memberikannya dukungan dana melalui dana hibah penelitian untuk penelitian ini.

DAFTAR PUSTAKA

- [1] Teachable Machine, "What is Teachable Machine?," Google. Accessed: Feb. 22, 2024. [Online]. Available: <https://teachablemachine.withgoogle.com/>
- [2] M. Carney *et al.*, "Teachable Machine: Approachable Web-Based Tool for Exploring Machine Learning Classification," in *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, in CHI EA '20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1–8. doi: 10.1145/3334480.3382839.
- [3] Arduino, "What is Arduino? | Arduino." Accessed: Feb. 20, 2023. [Online]. Available: <https://www.arduino.cc/en/Guide/Introduction>
- [4] E. A. U. Malahina, R. P. Hadjon, and F. Y. Bisilisin, "Teachable Machine: Real-Time Attendance of Students Based on Open Source System," *IJICS (International J. Informatics Comput. Sci.*, vol. 6, no. 3, p. 140, 2022, doi: 10.30865/ijics.v6i3.4928.
- [5] M. B. Baihaqi, Y. Litanianda, and A. Triyanto, "Implementasi Tensor Flow Lite Pada Teachable Untuk Identifikasi Tanaman Aglonema Berbasis Android," *Komputek*, vol. 6, no. 1, p. 70, 2022, doi: 10.24269/jkt.v6i1.1143.
- [6] N. M. Farhan and B. Setiaji, "Deteksi Penyakit Tanaman Padi Berbasis Android Melalui Pemanfaatan Teachable Machine," *Indones. J. Comput. Sci.*, vol. 12, no. 2, pp. 284–301, 2023, [Online]. Available: <http://ijcs.stmikindonesia.ac.id/ijcs/index.php/ijcs/article/view/3135>
- [7] C. Chazar and M. H. Rafsanjani, "Penerapan Teachable Machine Pada Klasifikasi Machine Learning Untuk Identifikasi Bibit Tanaman," in *Prosiding Seminar Nasional Inovasi dan Adopsi Teknologi (INOTEK)*, 2022, pp. 32–40. doi: <https://doi.org/10.35969/inotek.v2i1.207>.
- [8] B. Jiang, X. Li, L. Yin, W. Yue, and S. Wang, "Object Recognition in Remote Sensing Images Using Combined Deep Features," in *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, 2019, pp. 606–610. doi: 10.1109/ITNEC.2019.8729392.



- [9] M. Dol and A. Geetha, "A Learning Transition from Machine Learning to Deep Learning: A Survey," in *2021 International Conference on Emerging Techniques in Computational Intelligence (ICETCI)*, 2021, pp. 89–94. doi: 10.1109/ICETCI51973.2021.9574066.
- [10] A. Ng, *Machine Learning Yearning*. 2016. doi: 10.1007/978-981-10-1509-0_9.
- [11] Arduino, "Arduino ® Nano Arduino ® Nano Features," *Arduino*, pp. 1–13, 2022.
- [12] Interactive Telecommunications Program (ITP) at NYU, "Using the p5.serialport library." Accessed: Jun. 10, 2024. [Online]. Available: <https://itp.nyu.edu/physcomp/labs/labs-serial-communication/lab-serial-input-to-the-p5-js-ide/>
- [13] Mozilla, "The WebSocket API (WebSockets)," MDN web docs. Accessed: May 10, 2024. [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API
- [14] P5js, "p5.js," p5js.org. Accessed: May 10, 2024. [Online]. Available: <https://p5js.org/>
- [15] ml5js.org, "ml5." GitHub, 2022. [Online]. Available: <https://github.com/ml5js/ml5-library>
- [16] S. Van Every, "p5.serialport." GitHub, 2022. [Online]. Available: <https://github.com/p5-serial/p5.serialport>

