

Penerapan Metode YOLOv5 dan Text-To-Speech untuk Aplikasi Pengenalan Abjad dan Objek Sekitar Pada Anak Usia Dini

Atika Rahmi Putri¹, Ratna Dewi^{2*}, Ramiati³
^{1,2,3} Jurusan Teknik Elektro, Politeknik Negeri Padang
*Corresponding Author email : ratnadewi@pnp.ac.id

Abstrak— Perkembangan teknologi telah memberikan manfaat yang signifikan bagi berbagai lapisan masyarakat, termasuk anak-anak usia dini. Terdapat beberapa faktor internal dan eksternal yang dapat menghambat anak dalam mengingat abjad dan objek sekitar. Untuk mengatasi tantangan tersebut, diperlukan pendekatan yang inovatif dan menarik untuk meningkatkan minat dan keterlibatan anak dalam pembelajaran pengenalan abjad dan objek. Telah dilakukan penelitian dengan mengembangkan sebuah media pembelajaran interaktif yang ditujukan untuk anak usia dini dengan menggunakan pendekatan kecerdasan buatan (*Artificial Intelligence*). Algoritma You Only Look Once (YOLO) akan digunakan untuk mendeteksi abjad dan objek sekitar secara real-time, sementara teknologi Text-to-Speech (TTS) digunakan untuk mengubah teks menjadi suara. Dataset terdiri dari 2511 gambar dengan 36 kelas, meliputi abjad, buah-buahan, hewan, dan alat tulis. Dalam penelitian ini, metode YOLOv5s mendapatkan tingkat akurasi yang signifikan, mencapai 85% sehingga dapat bekerja dengan baik dalam mendeteksi objek.

Kata kunci : Anak usia dini, YOLO, Text To Speech.

Abstract— *Technological developments have provided significant benefits for various levels of society, including young children. There are several internal and external factors that can hinder children from remembering the alphabet and objects around them. To overcome these challenges, an innovative and interesting approach is needed to increase children's interest and involvement in learning to recognize alphabets and objects. Research has been carried out to develop an interactive learning media aimed at early childhood using an artificial intelligence approach. The You Only Look Once (YOLO) algorithm will be used to detect letters and surrounding objects in real-time, while Text-to-Speech (TTS) technology is used to convert text into sound. The dataset consists of 2511 images with 36 classes, including alphabets, fruits, animals, and stationery. In this research, the YOLOv5s method obtained a significant level of accuracy, reaching 85% so it could work well in detecting objects.*

Keywords: Early Childhood, YOLO, Text-to-Speech

© 2023 Elektron Jurnal Ilmiah

I. PENDAHULUAN

Pendidikan Anak Usia Dini adalah langkah pengembangan yang difokuskan pada anak sejak lahir hingga usia 6 (enam) tahun. Pada periode ini, terdapat berbagai aspek perkembangan yang perlu diperhatikan, salah satunya adalah perkembangan Bahasa. Pada anak usia dibawah 6 tahun, terdapat beberapa indikator perkembangan anak, mencakup kemampuan pengenalan simbol, pengenalan suara, menghasilkan coretan yang bermakna, serta dapat menuliskan dan mengucapkan abjad A-Z [1]. Peran media sangat berpengaruh dalam proses pendidikan karena berfungsi sebagai perantara untuk mengkomunikasikan informasi dan pengetahuan kepada peserta didik. Dengan berkembangnya kemajuan teknologi yang signifikan, maka teknologi dapat dimanfaatkan sebagai salah satu perkembangan yang dapat diterapkan dalam dunia pendidikan. Media pembelajaran interaktif ini dapat dikembangkan dengan menambahkan unsur kecerdasan buatan atau *Artificial Intelligence*.

Saat ini, telah banyak penelitian diberbagai bidang yang dijalankan untuk menghadirkan kemudahan dalam aktivitas sehari-hari dengan menggunakan teknologi YOLO. Aplikasi yang dirancang untuk mendeteksi bahasa isyarat dalam pengenalan huruf hijaiyah menggunakan metode YOLO telah terbukti mampu secara konsisten mengenali objek dengan tingkat akurasi yang tinggi, mencapai 95% [2]. Selain itu ada juga penelitian yang melakukan deteksi objek untuk anak usia dini yang bertujuan untuk mengkaji pengenalan objek-objek disekitar, seperti buku, gunting, dll [3]. Dalam perkembangan teknologi yang semakin pesat, teknologi *Text to Speech* juga banyak digunakan diberbagai bidang. Salah satu penelitian mengenai TTS yang dilakukan adalah perancangan aplikasi pendeteksi huruf Jepang (hiragana, katakana dan kanji). Penelitian ini memanfaatkan algoritma YOLO untuk mendeteksi tulisan pada gambar dan mengubahnya menjadi suara menggunakan *text to speech* [4]. Akurasi yang dihasilkan pada penelitian ini sekitar 86% dan 91%.

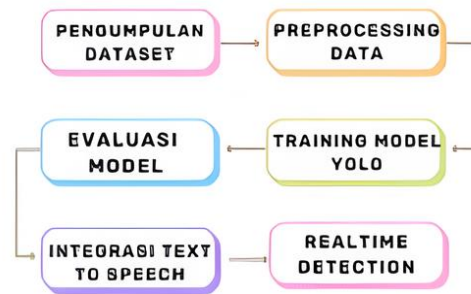
Anak mengalami kesulitan dalam mengingat abjad disebabkan oleh faktor internal dan eksternal. Faktor internal melibatkan perkembangan kognitif, motivasi, minat belajar, dan emosi anak. Faktor eksternal melibatkan lingkungan keluarga, sekolah, dan masyarakat [1]. Hal ini dapat menjadi penyebab menurunnya minat belajar anak. Kecerdasan buatan atau *Artificial Intelligence* (AI) adalah upaya untuk membuat mesin dapat meniru dan melakukan tugas yang membutuhkan kecerdasan manusia [5]. YOLO (*You Only Look Once*) adalah salah satu algoritma dalam bidang *Computer Vision* (penglihatan komputer) yang digunakan untuk melakukan deteksi objek dalam gambar atau video [6]. Algoritma YOLO memiliki keunggulan dalam kecepatan dan akurasi deteksi objek yang tinggi sehingga banyak digunakan dalam aplikasi seperti pemantauan lalu lintas, pengawasan keamanan, dan deteksi wajah manusia [7]. Algoritma YOLO juga dikembangkan untuk dapat melakukan deteksi secara *realtime*. *Text-to-Speech* (TTS) adalah salah satu teknologi di bidang kecerdasan buatan (AI) yang digunakan untuk mengubah teks menjadi suara. TTS merupakan salah satu bagian dari *Natural Language Processing* (NLP), yaitu cabang AI yang berfokus pada pemahaman dan pengolahan bahasa manusia [4].

Adapun dari penelitian-penelitian sebelumnya, masih terdapat kekurangan dalam mendukung aspek pengenalan objek secara *real-time* dan interaktif. Selain itu, mungkin juga terdapat kendala dalam integrasi antara teknologi deteksi objek dan penghasil suara yang memadai untuk mendukung pembelajaran anak-anak. Ide yang ditawarkan dalam penelitian ini adalah mengatasi kekurangan-kekurangan tersebut dengan memanfaatkan teknologi terkini seperti YOLOv5 dan *Text-to-Speech* (TTS) dalam sebuah aplikasi pembelajaran interaktif untuk anak usia dini. TTS digunakan untuk menghasilkan keluaran suara berdasarkan abjad atau kata yang terdeteksi oleh model YOLO. Melalui integrasi antara algoritma YOLO dan *Text to Speech* akan menciptakan sebuah media pembelajaran interaktif yang dapat mendeteksi objek yang ingin dikenali kemudian menghasilkan output suara. Dengan demikian, penelitian ini bertujuan untuk mengembangkan solusi yang lebih efektif dan menarik dalam pengenalan abjad dan objek sekitar untuk anak usia dini, yang dapat mengatasi kelemahan-kelemahan yang ada dalam aplikasi sebelumnya.

II. METODE

Gambar 1 berikut menampilkan tahapan yang akan dilakukan dalam penelitian ini. Proses *training* pada penelitian ini, dilakukan menggunakan Google Colab dengan penerapan metode YOLOv5 dan versi YOLOv5s. YOLO merupakan sebuah jaringan yang digunakan untuk mendeteksi objek pada gambar [5]. Sementara itu, YOLOv5s adalah salah satu varian model deteksi objek berbasis *deep learning* yang lebih

ringan dan lebih cepat dibandingkan dengan pendahulunya, YOLOv4. Model YOLOv5s adalah salah satu varian model YOLOv5 yang termasuk dalam kategori "*Small*" atau ringan.



Gambar 1. Alur Penelitian

Varian ini memiliki ukuran yang lebih kecil dan jumlah parameter yang lebih terbatas dibandingkan dengan varian lain seperti YOLOv5m (Medium) atau YOLOv5x (*Extra-large*). Meskipun lebih ringan, YOLOv5s tetap mengutamakan kecepatan dan kemudahan dijalankan pada perangkat dengan sumber daya terbatas. Kelebihan dari YOLOv5s adalah kemampuannya untuk melakukan deteksi objek secara *real-time* pada berbagai skenario, termasuk pada gambar dan video.

A. Pengumpulan Dataset

Pada tahap awal pengembangan sistem, hal yang pertama dilakukan adalah mengumpulkan gambar-gambar yang akan digunakan sebagai data pelatihan (*train*) dan validasi. Dataset tersebut dikumpulkan melalui proses *scraping* (pengumpulan data secara otomatis) dari kumpulan gambar terbuka di Google Image, serta melakukan pengunduhan gambar secara manual dari internet satu per satu [5]. Dataset yang akan digunakan dalam proses pelatihan data terdiri dari 36 kelas objek, di mana setiap kelas memiliki sekitar 50 hingga 80 gambar. Dengan demikian, total gambar yang digunakan dari 36 objek adalah sekitar 2511 gambar dalam format PNG.

B. Labelling Dataset

Setelah semua gambar yang dibutuhkan terkumpul dan sudah dalam format yang sama, selanjutnya adalah melakukan proses *pre processing* pada gambar dengan cara memberi label pada setiap objek dalam gambar menggunakan *LabelImg*. *LabelImg* adalah sebuah perangkat lunak sumber terbuka (*open-source*) yang digunakan untuk melakukan anotasi atau pelabelan objek pada gambar [8]. Tujuan dari memberikan label pada gambar adalah untuk mengidentifikasi fitur khas yang mewakili setiap objek yang ada pada gambar [8]. Fitur khas ini akan digunakan sebagai bagian dalam pembelajaran (*learning*) saat melatih model, serta akan digunakan dalam tahap pengujian.

menjadi data *training* sebanyak 75% dan data *validation* sebanyak 25%. Semua gambar dan file anotasi yang telah terkumpul di *split* menggunakan *roboflow* dengan hasil pada gambar 7 sebagai berikut :

Source Images	images : 2.511
	Classes : 36
	Unannotated : 0
Train/Test Split	Training Set : 1.9k images
	Validation Set : 633 images
	Testing Set : images

Gambar 7. Hasil Split Data

D. Pelatihan Model YOLO

Langkah awal dalam memulai proses pelatihan adalah dengan membuat file Google Colab dan mengatur *runtime* untuk menggunakan GPU. Mengaktifkan GPU ini dilakukan agar *notebook* Colab dapat menjalankan sistem YOLO dengan kecepatan deteksi 100 kali lebih cepat daripada menggunakan CPU [9]. Proses *training* yang dilakukan pada penelitian ini yaitu menggunakan 100 *epoch*. Adapun tujuan dari proses *training* data ini adalah untuk mengoptimalkan bobot (*weight*) dari model deteksi YOLO sehingga model tersebut dapat dengan akurat dan efisien dalam mendeteksi objek tertentu [9].

Setelah proses *training* selesai, dapat dilakukan evaluasi dengan mencoba mendeteksi gambar yang tidak dimasukkan kedalam dataset, agar mengetahui seberapa baik model YOLO dapat melakukan deteksi. Evaluasi model pada YOLO melibatkan penggunaan beberapa metrik, seperti *Mean Average Precision* (mAP) untuk mengukur akurasi deteksi, *Confusion Matrix* untuk menghitung *presisi*, *recall*, dan akurasi, *IoU* (*Intersection over Union*) untuk mengevaluasi tumpang tindih antara kotak pembatas prediksi dan objek sebenarnya, serta *F1 Score* untuk menilai keseimbangan antara *presisi* dan *recall*. Metrik-metrik ini memberikan wawasan penting tentang kinerja keseluruhan model dalam mendeteksi objek, membantu dalam membandingkan model, dan memungkinkan penyesuaian dan perbaikan untuk mencapai performa yang lebih baik [10].

Jika akurasi yang didapatkan sudah sesuai dengan yang diharapkan dalam proses penelitian, model YOLO dapat disimpan dalam bentuk *weight* (.pt). File *weight* adalah berat dari model YOLO yang telah dilakukan proses *training*/pelatihan. File *weight* ini berisi informasi tentang bobot dan bias yang telah dioptimalkan selama pelatihan. File *weight* ini nantinya dapat digunakan untuk memuat model YOLO dan digunakan untuk mendeteksi objek pada gambar atau video baru. Dengan menggunakan file *weight* yang

sesuai, maka YOLO dapat diimplementasikan dalam berbagai bahasa pemrograman atau menggunakan *framework* seperti PyTorch atau TensorFlow. File *weight* inilah yang memungkinkan model YOLO untuk digunakan tanpa harus melalui proses pelatihan ulang, sehingga lebih efisien dan praktis dalam penggunaannya. Dalam penelitian ini, file *weight* akan digunakan untuk mendeteksi objek secara *realtime*.

Gambar 8 menunjukkan *Confusion matrix* yang digunakan untuk mengevaluasi kinerja model pada dataset uji. Ini menyediakan informasi tentang sejauh mana hasil deteksi model cocok dengan hasil deteksi yang sebenarnya. *Confusion matrix* memberikan wawasan yang penting tentang kualitas prediksi model dan membantu dalam memahami kinerja keseluruhan model pada tugas klasifikasi [11].

		Actual Values	
		1 (Positive)	0 (Negative)
Predicted Values	1 (Positive)	TP (True Positive)	FP (False Positive) <i>Type I Error</i>
	0 (Negative)	FN (False Negative) <i>Type II Error</i>	TN (True Negative)

Gambar 8. Confusion Matrix

E. Integrasi Text-to-Speech

Output suara pada penelitian ini dilakukan menggunakan *library* *pyttsx3*. Setelah itu menginisialisasi konfigurasi sintesis suara seperti yang terlihat pada Gambar 9.

```
# Inisialisasi konfigurasi sintesis suara
engine = pyttsx3.init()
voices = engine.getProperty('voices')
# Pilih suara bahasa Indonesia
for voice in voices:
    if 'Indonesia' in voice.languages:
        engine.setProperty('voice', voice.id)
        break
```

Gambar 9. Integrasi TTS

Output suara yang digunakan pada penelitian ini adalah Bahasa Indonesia. Ketika objek terdeteksi oleh sistem, maka label yang terdeteksi langsung dikonversi menjadi suara.

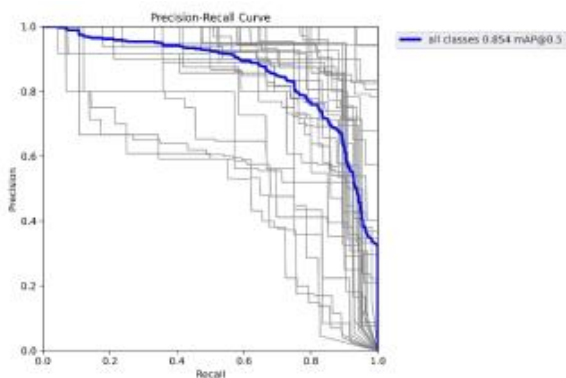
III. HASIL DAN PEMBAHASAN

Bagian ini akan memaparkan analisis mengenai data kurva hasil pelatihan dan hasil pengujian *real-time* dari model YOLOv5.

A. Analisis Hasil Penelitian

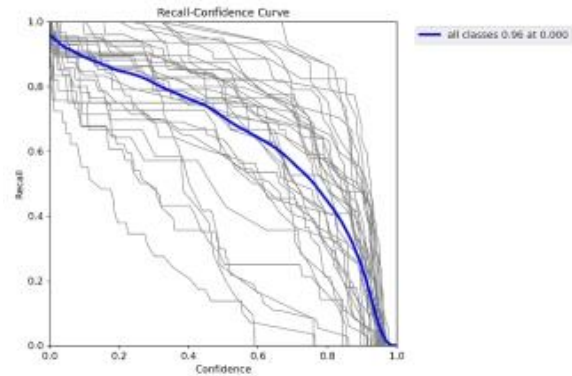
Setelah melalui proses *training* YOLO dengan menggunakan 2511 gambar berbeda yang termasuk dalam 36 kelas, selama 100 *epoch* dengan ukuran *batch* 24, hasil yang didapatkan sangat baik. Model yang telah dilatih mampu mengenali objek-objek dalam gambar dengan tingkat presisi dan *recall* yang tinggi. Hasil evaluasi juga menunjukkan *F1 score* yang cukup baik, yang berarti bahwa model ini dapat menjalankan tugas deteksi objek dengan akurat.

Pada Gambar 11 menunjukkan hasil *Precision-Recall Curve*, yaitu grafik yang menunjukkan hubungan antara *precision* dan *recall* pada berbagai *threshold confidence* yang digunakan untuk deteksi objek. Dari hasil yang didapatkan, "All classes 0.85 mAP@0.5", yang artinya model deteksi objek dengan tingkat akurasi sebesar 0.854 pada semua kelas objek dalam dataset, dengan menggunakan *IoU threshold* sebesar 0.5 untuk mengukur akurasi deteksi. Semakin tinggi skor mAP, semakin baik performa deteksi objek model tersebut[6].



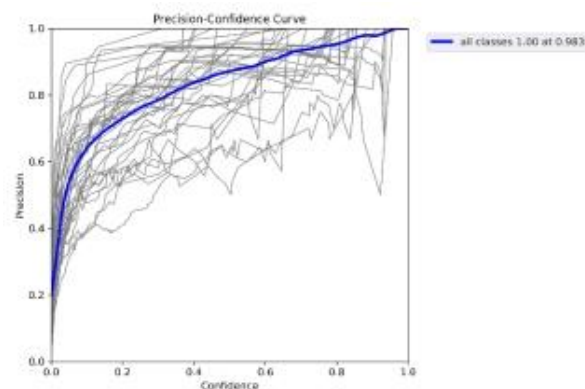
Gambar 10. Precision-Recall Curve

Selanjutnya pada Gambar 12 menunjukkan *Recall Confidence Curve*, yaitu grafik yang menunjukkan hubungan antara *recall* dan tingkat kepercayaan (*confidence*) pada deteksi objek. Dari hasil *training* yang didapatkan, *Recall Confidence Curve* menunjukkan nilai 0.96 pada tingkat kepercayaan (*confidence*) 0.00, hal ini mengindikasikan bahwa tingkat *recall* yang tinggi (sebesar 0.96) dapat dicapai dengan menggunakan *threshold confidence* yang sangat rendah (sebesar 0.00).



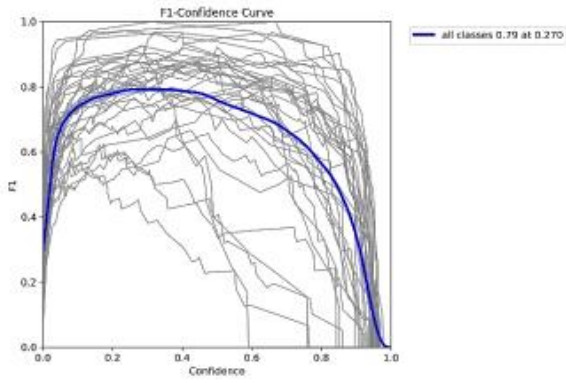
Gambar 11. Recall-Confidence Curve

Gambar 13 menunjukkan *Precision Confidence Curve*, yaitu grafik yang menunjukkan hubungan antara *precision* dan tingkat kepercayaan (*confidence*) pada deteksi objek. Dari hasil *training*, *Precision Confidence Curve* menunjukkan nilai 1.00 pada tingkat kepercayaan (*confidence*) 0.983 untuk semua kelas, artinya model deteksi objek memiliki tingkat presisi yang sempurna pada *threshold confidence* 0.983 untuk semua kelas objek. Hasil ini menunjukkan bahwa model memiliki tingkat akurasi yang sangat tinggi dalam mengenali dan mengklasifikasikan objek.



Gambar 12. Precision-Confidence Curve

Pada Gambar 14 menunjukkan *F1 Confidence Curve*, yaitu grafik yang menunjukkan hubungan antara skor F1 dan tingkat kepercayaan (*confidence*) pada deteksi objek. Dari hasil *training*, *F1 Confidence Curve* menunjukkan nilai 0.79 pada tingkat kepercayaan (*confidence*) 0.270 untuk semua kelas, artinya model deteksi objek memiliki skor F1 yang sebesar 0.79 pada *threshold confidence* 0.270 untuk semua kelas objek. Nilai F1 yang tinggi menunjukkan bahwa model memiliki keseimbangan yang baik antara deteksi yang akurat (presisi) dan pengenalan objek yang benar (*recall*) pada *threshold confidence* 0.270.



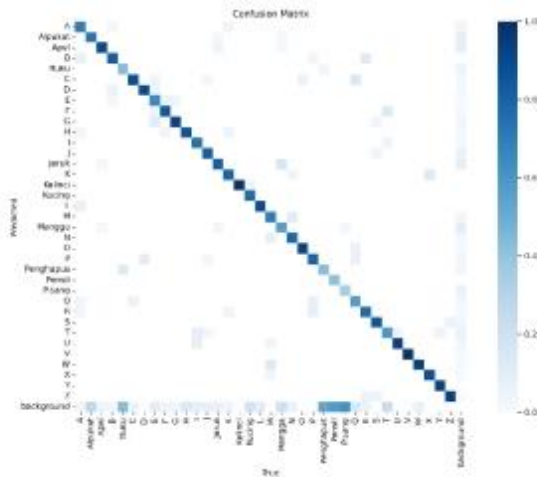
Gambar 13. F1 Confidence Curve

Dari penjabaran diatas, dapat disimpulkan hasil evaluasi metrik dari pelatihan YOLO pada Tabel 1 sebagai berikut

Tabel 1. Hasil Evaluasi Metrik

Class	mAP	Precision	Recall	F1
All Class	0,854	1,00	0,96	0,79

Gambar 15 dibawah menunjukkan *confusion matrix* yang didapatkan dari proses pelatihan data.



Gambar 14. Confusion Matrix

Dalam mengevaluasi peforma algoritma YOLO, dapat menggunakan acuan *Confusion Matrix*. *Confusion Matrix* merepresentasikan prediksi dan kondisi sebenarnya(aktual) dari data yang dihasilkan oleh algoritma.

Untuk menghitung nilai *accuracy*, digunakan persamaan berikut :

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

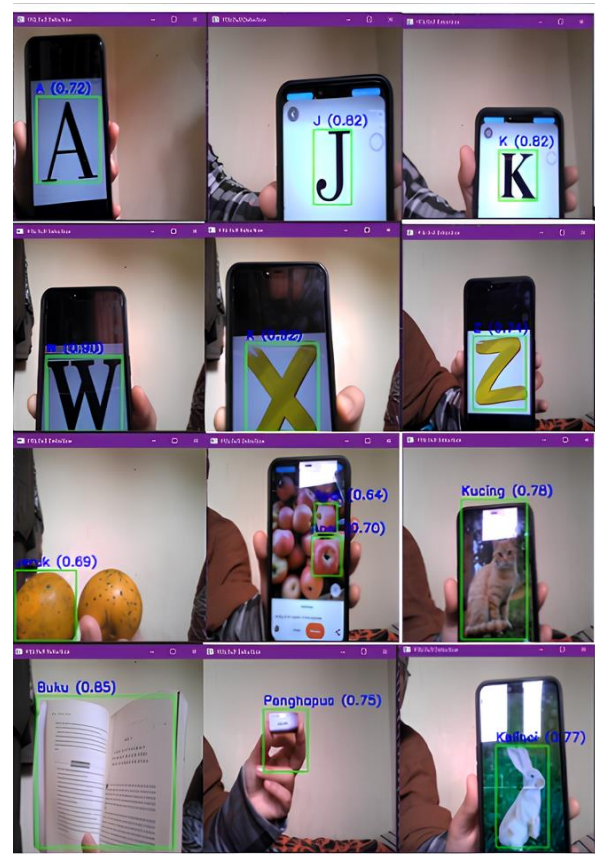
$$Accuracy = \frac{27,41 + 4,21}{27,41 + 4,21 + 0,59 + 4,79} = \frac{31,62}{37} = 0,854 \times 100\%$$

$$Accuracy = 85 \%$$

Jadi pada penelitian ini, didapatkan nilai *accuracy* yang cukup baik dalam mendeteksi abjad dan objek sekitar, yaitu 85%.

B.. Hasil Pengujian Realtime

Pendeteksian abjad dan objek sekitar secara *real-time* berjalan dengan lancar dan cukup akurat serta cepat dalam mendeteksi objek yang terlihat oleh kamera. Berikut pada Gambar 16 beberapa hasil deteksi secara *real-time*:



Gambar 15. Hasil Pengujian Real-time

Dalam implementasi deteksi secara *real-time* ini, objek akan diidentifikasi melalui *bounding box* dan dilengkapi dengan label objek beserta tingkat akurasi. Selain itu, setelah objek terdeteksi, sistem secara instan akan mengubah teks label menjadi suara, memungkinkan sistem untuk secara langsung menyebutkan nama objek yang terdeteksi. Adapun

hasil yang didapatkan melalui deteksi secara *real-time* ini berkisar antara 64-92%. Pendeteksian yang dilakukan secara *real-time* ini cukup cepat dan akurat dalam mendeteksi objek. Dengan demikian, pengguna akan mendapatkan pengalaman interaktif ketika berinteraksi dengan sistem deteksi ini.

C.. Hasil Pengujian di Aplikasi

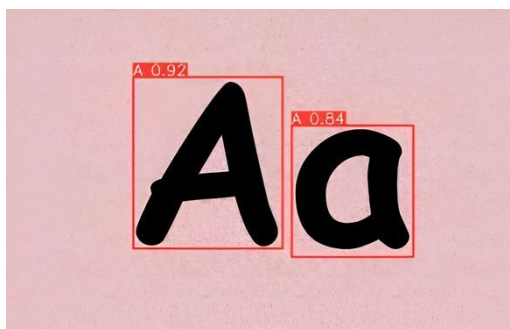
Aplikasi pendeteksi abjad dan objek ini memiliki ini terdapat beberapa *button* yang digunakan untuk menjalankan aplikasi, berikut penjelasannya :

1. *Button* “Choose File” digunakan untuk menginputkan gambar dari penyimpanan. Gambar 4.6 berikut merupakan tampilan ketika sudah menginputkan gambar yang ingin dideteksi



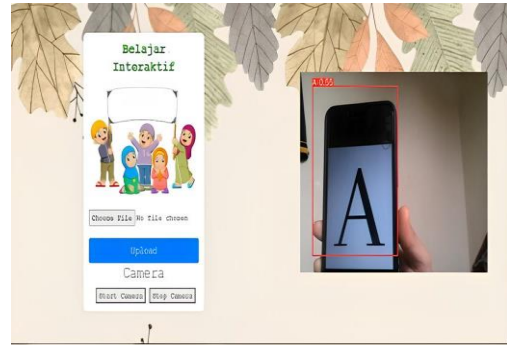
Gambar 16. Button choose file

2. *Button* “Upload” digunakan untuk mengupload gambar yang telah diinputkan sebelumnya. Ketika tombol *upload* di klik, maka gambar yang diinputkan sebelumnya akan terdeteksi seperti Gambar 4.7 berikut



Gambar 17. button upload

3. *Button* “Start Camera” digunakan untuk mengakses kamera sehingga pendeteksian dapat dilakukan secara *real-time*. Berikut pada Gambar 4.8 tampilan aplikasi ketika *button* “start camera” di klik



Gambar 18. button start camera

Ketika pendeteksian secara *real-time* ini berjalan, aplikasi akan mengenali objek-objek kemudian memberikan *bounding box* untuk setiap objek yang terdeteksi. Label objek yang terdeteksi ini akan langsung terkonversi menjadi suara secara otomatis, sehingga setiap label yang terdeteksi akan diikuti oleh suara yang menyebutkan label objek tersebut.

Dalam penelitian ini, terdapat analisis terhadap hasil utama yang diperoleh guna mengevaluasi kinerja model yang dikembangkan. Perbandingan tersebut dilakukan dengan membandingkan metrik utama, seperti akurasi, presisi, recall, dan F1-score. Hasil analisis dapat dilihat melalui Tabel 2 dibawah. Perbandingan ini dilakukan dengan mengacu pada penelitian terdahulu yang dipublikasikan oleh D. I. Mulyana, M. F. Lazuardi, and M. B. Yel, dengan judul penelitian “Isyarat Dalam Pengenalan Huruf Hijaiyah Dengan Metode YOLOV5.

Tabel 2. Perbandingan Evaluasi Metrik

Penelitian	Real-time detection	Precision	Recall	F1
Penelitian ini	0,92	1,00	0,96	0,79
Pengenalan huruf hijaiyah, YOLOv5	0,95	1,00	0,95	0,90

Dengan demikian, meskipun penelitian sebelumnya memiliki nilai akurasi yang sedikit lebih tinggi dan *F1-score* yang lebih baik, penelitian ini menunjukkan *recall* yang lebih baik. Ini menunjukkan bahwa ada *trade-off* antara presisi dan *recall* antara kedua penelitian, karena perbedaan jumlah kelas dan dataset yang digunakan.

IV. KESIMPULAN

Berdasarkan hasil uji coba pada penelitian ini, dapat disimpulkan bahwa algoritma YOLOv5 menunjukkan kinerja yang sangat baik dalam mendeteksi abjad dan objek di sekitar. Tingkat akurasi yang berhasil dicapai oleh algoritma YOLO pada percobaan dengan 100 epoch mencapai 85%, sementara dalam pendeteksian secara *real-time*, *accuracy* berkisar antara 64-92%. Sistem *text-to-speech* yang digunakan juga terbukti akurat dan responsif dalam mengkonversi label objek menjadi suara. Dengan kombinasi kemampuan deteksi dan konversi suara yang efisien, sistem ini memiliki potensi besar sebagai alat pembelajaran interaktif yang menarik dan efektif. Meskipun demikian, perlu diperhatikan bahwa hasil dan potensi aplikasi ini dapat ditingkatkan melalui pembaruan model atau penyesuaian parameter.

REFERENSI

- [1] Intan Ika Puspita Sari, "Persepsi Guru Terhadap Sosial Emosi Anak Usia Dini Dan Faktor Yang Memengaruhi", 2020.
- [2] D. I. Mulyana, M. F. Lazuardi, and M. B. Yel, "Deteksi Bahasa Isyarat Dalam Pengenalan Huruf Hijaiyah Dengan Metode YOLOV5," *Jurnal Teknik Elektro dan Komputasi (ELKOM)*, vol. 4, no. 2, pp. 145–151, Aug. 2022, doi: 10.32528/elkom.v4i2.8145.
- [3] I. Yuni Wulandari, N. Indroasyoko, R. Mudia Alti, Y. N. Asri, and R. Hidayat, "Pengenalan Sistem Deteksi Objek untuk Anak Usia Dini Menggunakan Pemrograman Python," *remik*, vol. 6, no. 4, pp. 664–673, Oct. 2022, doi: 10.33395/remik.v6i4.11772.
- [4] S. Kahfi, "Perancangan Aplikasi Text To Speech Dalam Pengucapan Kata Bahasa Jepang Berbasis Android."
- [5] O. E. Karlina and D. Indarti, "Pengenalan Objek Makanan Cepat Saji Pada Video Dan Real Time Webcam Menggunakan Metode You Look Only Once (YOLO)," *Jurnal Ilmiah Informatika Komputer*, vol. 24, no. 3, pp. 199–208, 2019, doi: 10.35760/ik.2019.v24i3.2362.
- [6] A. Wibowo, L. Lusiana, and T. K. Dewi, "Implementasi Algoritma Deep Learning You Only Look Once (YOLOv5) Untuk Deteksi Buah Segar Dan Busuk," *Paspalum: Jurnal Ilmiah Pertanian*, vol. 11, no. 1, p. 123, Mar. 2023, doi: 10.35138/paspalum.v11i1.489.
- [7] D. I. Mulyana and M. A. Rofik, "Implementasi Deteksi Real Time Klasifikasi Jenis Kendaraan Di Indonesia Menggunakan Metode YOLOV5."
- [8] L. Rahma, H. Syaputra, A. H. Mirza, and S. D. Purnamasari, "Objek Deteksi Makanan Khas Palembang Menggunakan Algoritma YOLO (You Only Look Once)," 2021.
- [9] Khairunnas, Eko Mulyanto Yuniarno daAhmad Zaini, "Pembuatan Modul Deteksi Objek Manusia Menggunakan Metode YOLO untuk *Mobile Robot* ", 2021.
- [10] Hasbi Dawami , Ema Rachmawati , Mahmud Dwi Sulistiyo "Deteksi Penggunaan Masker Wajah Menggunakan Yolov5", 2023.
- [11] F. Rofii, G. Priyandoko, M. I. Fanani, and A. Suraji, "Vehicle Counting Accuracy Improvement By Identity Sequences Detection Based on Yolov4 Deep Neural Networks," *TEKNIK*, vol. 42, no. 2, pp. 169–177, Aug. 2021, doi: 10.14710/teknik.v42i2.37019.