

Deteksi Objek Bahasa Isyarat Alfabet BISINDO Menggunakan *Deep learning* dan Arsitektur *YOLO* *Bisindo Alphabet Sign Language Object Detection Using Deep Learning and YOLO Architecture*

Alicia Kinanti¹, Mas Aly Afandi^{1*}, Indah Permatasari¹, Nicolas Yonara Tarigan¹
Program Studi S1 Teknik Telekomunikasi, Fakultas Teknik Telekomunikasi dan Elektro¹
Institut Teknologi Telkom Purwokerto
E-mail: aly@ittelkom-pwt.ac.id

Abstrak

Bahasa isyarat merupakan cara bagi penyandang disabilitas tuna rungu dan tuna wicara berkomunikasi. Bahasa isyarat umumnya dipelajari oleh penyandang disabilitas dan tidak dipelajari oleh masyarakat normal. Kondisi ini menjadi masalah saat dibutuhkan komunikasi antara penyandang disabilitas dan masyarakat normal. Salah satu cara untuk membantu mempermudah komunikasi antara penyandang disabilitas dan masyarakat normal adalah menggunakan teknologi *deep learning* dalam menerjemahkan bahasa isyarat. Bahasa isyarat yang digunakan memiliki berbagai macam interpretasi mulai dari huruf, kata dan kalimat. Potensi *deep learning* dalam mengenali bahasa isyarat dengan input gambar dapat dimulai dari dasar untuk mengenali alfabet BISINDO. YOLOv4 sebagai salah satu arsitektur *deep learning* yang memiliki performa tinggi di berbagai pengenalan citra akan diuji untuk mengenali alfabet BISINDO. Tujuan penelitian ini adalah menguji performa arsitektur YOLOv4 dalam mengenali bahasa isyarat. Hasil dari penelitian ini menunjukkan bahwa nilai *accuracy*, *precision*, dan *recall* yang didapatkan menggunakan arsitektur YOLOv4 adalah 99,4%, 92,3%, dan 92,3%. Performa ini menunjukkan bahwa *deep learning* dengan arsitektur YOLOv4 dapat mengenali bahasa isyarat alfabet BISINDO dengan performa yang tinggi. Menerjemahkan alfabet BISINDO menggunakan arsitektur YOLOv4 dapat dilakukan dengan akurat.

Kata kunci: *Alfabet BISINDO, Deep learning, Disabilitas, Komunikasi, YoloV4*

Abstract

Sign language is a way for people with hearing and speech impairments to communicate. Sign language is generally learned by people with disabilities and not by normal people in general. This condition becomes a problem when communication is needed between people with disabilities and normal people. One way to help facilitate communication between people with disabilities and normal people is to use deep learning technology in translating sign language. Sign language used has a variety of interpretations ranging from letters, words and sentences. The potential of deep learning in recognizing sign language with image input can start from the basis of recognizing the BISINDO alphabet. YOLOv4 as one of the deep learning architectures that has high performance in various image recognition will be tested to recognize the BISINDO alphabet. The purpose of this research is to test the performance of YOLOv4 architecture in recognizing sign language. The results of this study show that the accuracy, precision, and recall values obtained using YOLOv4 architecture are 99.4%, 92.3%, and 92.3%. This performance shows that deep learning with YOLOv4 architecture can recognize BISINDO alphabet sign language with high performance. Translating the BISINDO alphabet using YOLOv4 architecture can be done accurately.

Keywords: *BISINDO Alphabet, Deep learning, Disabilities, Communication, YoloV4*

1. PENDAHULUAN

Komunikasi merupakan suatu kegiatan yang dilakukan setiap orang untuk dapat saling menyampaikan serta bertukar informasi [1]. Komunikasi dapat dilakukan antara dua orang atau lebih dengan cara yang tepat, sehingga informasi atau pesan yang ingin disampaikan dapat dimengerti oleh lawan bicara atau penerima informasi. Namun apa jadinya jika ingin menyampaikan informasi kepada lawan bicara yang memiliki keterbatasan khusus seperti penyandang disabilitas. Penyandang disabilitas merupakan suatu keterbatasan yang dapat dialami seseorang sehingga orang tersebut tidak mampu untuk mendengar, alhasil dapat menghambat orang tersebut untuk memperoleh informasi bahasa melalui pendengarannya [2]. Dari data yang didapatkan oleh badan Kementrian Kesehatan tahun 2015, jumlah orang penyandang disabilitas tuna wicara mencapai 637.541 jiwa [3]. Semakin banyak orang yang mengalami hal tersebut, dapat mengakibatkan hambatan yang berujung kepada minimnya akses untuk dapat saling berkomunikasi dalam kehidupan publik yang normal [4].

Para penyandang disabilitas tuna wicara menggunakan gerakan tubuh atau dikenal dengan bahasa isyarat yang dipakai sebagai media penyampaian informasi [5]. Bahasa isyarat adalah bahasa yang digunakan oleh kelompok tuna wicara dalam berkomunikasi dengan cara menggerakkan kedua tangannya. Di Indonesia dikenal dengan istilah BISINDO atau bahasa isyarat Indonesia yang merupakan bahasa alami yang tumbuh di kalangan kelompok tersebut. Namun tidak semua kaum dengar mampu untuk mengetahui bahasa tersebut, sehingga kaum disabilitas tidak mampu untuk dapat mengekspresikan dirinya kepada kaum dengar. Hal ini juga mendorong adanya diskriminasi dan stereotip terhadap penyandang disabilitas [6].

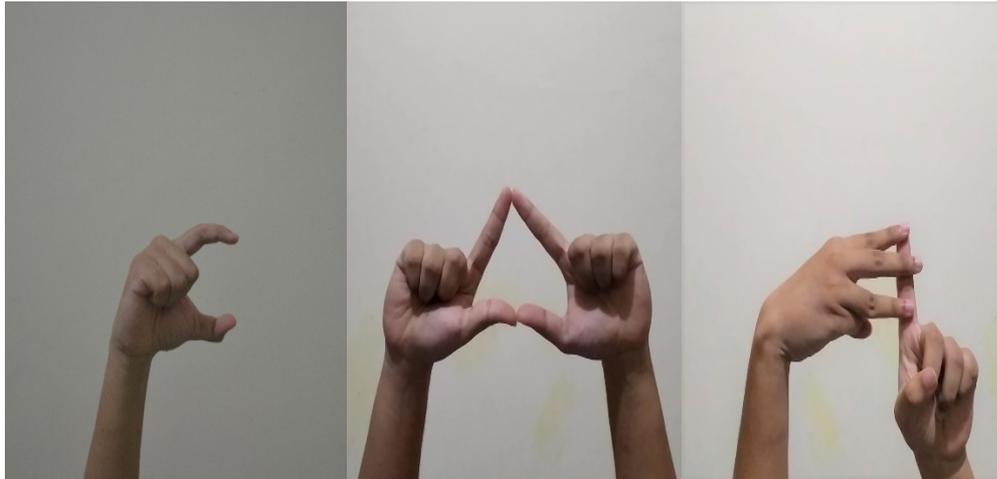
Pemanfaatan teknologi dibutuhkan untuk meningkatkan komunikasi antara kaum dengan dan disabilitas. Berbagai teknologi kunci yang dapat dimanfaatkan untuk membantu masalah tersebut adalah teknologi pengolahan citra dan kecerdasan buatan. Pengolahan citra sendiri telah dimanfaatkan di berbagai sektor seperti sektor agrikultur [7], [8], bioinformatika [9], pertahanan [10], Kesehatan [11], dan disabilitas [12]. Pemanfaatan teknologi untuk disabilitas sendiri juga mengalami peningkatan [13], [14]. Berbagai penelitian mengenai pemanfaatan pengolahan citra dan kecerdasan buatan untuk menyelesaikan permasalahan disabilitas tuna netra juga mengalami peningkatan. Penelitian mengenai bahasa isyarat menggunakan citra huruf BISINDO telah dilakukan sebelumnya menggunakan arsitektur SSD *MobileNet* [15], [16]. Hasil dari penelitian tersebut menunjukkan bahwa bahasa isyarat BISINDO dapat dikenali menggunakan metode *deep learning* (DL) namun dapat lebih ditingkatkan kembali.

Model dengan performa yang tinggi dibutuhkan dalam mengenali alfabet BISINDO untuk membantu komunikasi antara kaum disabilitas dan kaum dengar. Maka dari itu dibuatlah sistem pendeteksian bahasa isyarat dengan menggunakan algoritma *Convolution Neural Network* (CNN) dengan arsitektur yang dipakai adalah *You Only Look Once* (YOLO) yang memiliki performa baik dalam pengenalan citra. DL merupakan salah satu metode *learning* yang memanfaatkan *artificial neural network* yang berlapis-lapis, yang digunakan dalam menyelesaikan permasalahan pengenalan bahasa isyarat alfabet BISINDO. YOLO memiliki beberapa versi dan salah satu versi YOLO yang digunakan pada penelitian ini karena performanya yang baik adalah YOLOv4. Penelitian ini bertujuan untuk menguji performa YOLOv4 untuk mengenali citra bahasa isyarat BISINDO.

2. METODE PENELITIAN

2.1 Data Collection

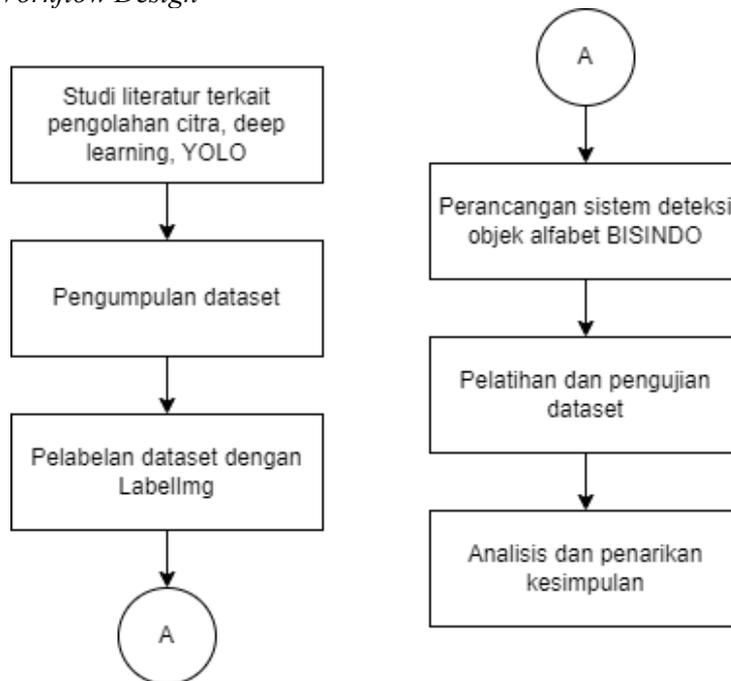
Dalam pengolahan citra, *dataset* merujuk pada kumpulan citra yang digunakan untuk pelatihan, pengujian, serta pengujian model yang telah dibangun. *Dataset* sendiri juga berperan penting guna pengembangan serta evaluasi pada pendeteksian objek. Pada penelitian ini *dataset* diperoleh dari *Kaggle* dan foto pribadi yang diambil dengan menggunakan kamera *Android*. Berikut beberapa data citra yang digunakan sebagai *dataset*



Gambar 1 *Dataset* Alfabet BISINDO

Pada Gambar 1 merupakan contoh foto *dataset* alfabet BISINDO. Data yang telah terkumpul dalam penelitian ini sejumlah 494 data citra, yang terdiri dari 26 alfabet BISINDO. Masing-masing alfabet memiliki 19 data citra, 11 data citra dari situs *Kaggle* dan 8 data citra dari foto peneliti. Citra yang diambil langsung oleh peneliti memiliki spesifikasi yang sama dengan citra yang ada pada situs *Kaggle* dengan pengaturan ukuran citra 1077 x 1077 piksel dan diambil pada jarak 70 cm.

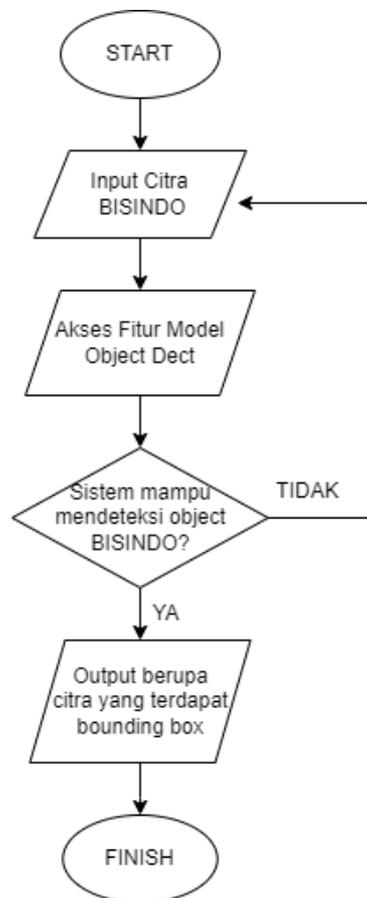
2.3 *System Workflow Design*



Gambar 2 Blok Digram Penelitian

Blok diagram pada Gambar 2 menunjukkan urutan penelitian yang dilakukan dari awal sampai akhir secara rinci. Langkah awal yang dilakukan adalah melakukan studi *literature*. Studi *literature* dilakukan untuk mencari informasi serta menambah wawasan mengenai topik yang akan diambil. Langkah selanjutnya adalah mengumpulkan *dataset* citra. Data primer dari penelitian ini menggunakan citra BISINDO yang diambil dari situs *Kaggle* serta menggunakan kamera peneliti.

Citra yang diambil dengan kamera peneliti menggunakan pengaturan yang sama sehingga tidak perlu dilakukan *preprocessing*. Hal ini juga menjadi salah satu kelebihan DL dan YOLO karena ekstraksi fitur yang umumnya dilakukan pada tahap preprocessing tidak diperlukan lagi karena dilakukan otomatis oleh model. Langkah selanjutnya adalah pelabelan citra untuk masing-masing huruf dengan menggunakan *LabelImg*. Pelabelan citra dilakukan agar nantinya model yang dibangun dapat mengenali objek alfabet, serta memperoleh hasil prediksi kelas pada objek. Selanjutnya, citra akan dibagi menjadi 2 bagian yaitu data *training* dan *testing*. Setelah dilakukan proses pelatihan *dataset* maka akan menghasilkan model sistem BISINDO yang nantinya akan digunakan sebagai pedoman dalam pengujian. Model dikembangkan menggunakan metode *Convolution Neural Network* (CNN). Peneliti juga memutuskan untuk menggunakan arsitektur CNN yang sudah pernah digunakan. Kemudian dilanjutkan pada tahap pengujian. Pengujian dilakukan dengan menguji kemampuan model berdasarkan *image processing* dalam mengenali bahasa isyarat dari huruf A sampai Z atau sebanyak 26 kali percobaan secara keseluruhan alfabet BISINDO. Seluruh hasil pengujian akan dicatat berdasarkan nilai *confusion matrix* yang dihasilkan. Apabila target berhasil tercapai akan dikategorikan sebagai berhasil ditebak dengan benar dan apabila tidak tercapai akan dikategorikan sebagai belum berhasil ditebak dengan benar.



Gambar 3 Diagram Alir Pembuatan Sistem

Pada Gambar 3 menunjukkan diagram alir pembuatan sistem pemanfaatan arsitektur algoritma YOLO dalam mendeteksi bahasa isyarat alfabet BISINDO. Pada blok *input* yang ditampilkan berupa gambar foto tangan yang memperagakan bahasa isyarat yang diambil secara *real time* menggunakan *web camera* kemudian akan di *capture* menghasilkan sebuah gambar. Selanjutnya akan diproses menggunakan arsitektur YOLOv4 dan akses fitur model *object detection*. Arsitektur YOLO akan diproses berdasarkan *dataset* yang sebelumnya telah melewati tahap *training* atau tahap pelatihan *dataset*, serta ekstraksi fitur yang didapatkan dari data mentah

darknet yang sebelumnya sudah dilatih terlebih dahulu. Maka dari itu, ketika *web camera* menangkap gambar tangan, kemudian akan menghasilkan *output* dengan menampilkan gambar yang berisi informasi berupa nilai *confident rate* serta prediksi kelas yang dideteksi. Semua informasi ini akan dibungkus pada *bounding box*.

2.4 Pengujian Sistem

Proses pengujian sistem akan dilakukan menggunakan *confusion matrix*. *Confusion matrix* adalah metode pengukuran untuk mengevaluasi kinerja dan tingkat keakuratan suatu proses klasifikasi. Di dalam *confusion matrix* dikenal dengan istilah 4 nilai prediksi dari hasil proses klasifikasi yaitu *True Positive* (TP), *False Positive* (FP), *False Negative* (FN), dan *True Negative* (TN).

		Nilai Aktual	
		Positive	Negative
Nilai Prediksi	Positive	TP	FP
	Negative	FN	TN

Gambar 4 Tabel *Confusion Matrix*

Gambar 4 merupakan *confusion matrix* yang digunakan. Penggunaan *confusion matrix* pada penelitian ini dilakukan untuk menguji kategori multikelas. Pengujian pada permasalahan multikelas memiliki perbedaan dengan kelas biner. Saat pengujian dilakukan untuk kelas “A” dan sistem mampu mengenali citra yang diuji sebagai kelas “A” maka parameter TP akan bertambah. Jika saat pengujian parameter selain A diberikan sebagai input dan diprediksi oleh sistem sebagai citra bukan “A” maka parameter TN akan bertambah. Jika citra “A” diberikan sebagai input dan diprediksi oleh sistem sebagai citra bukan “A” maka parameter FN akan bertambah. Jika diberikan parameter bukan “A” dan diprediksi oleh sistem sebagai “A” maka parameter FP akan bertambah. Dalam perhitungan parameter TP, TN, FP, dan FN parameter TN dan FP berpotensi untuk memiliki nilai yang lebih tinggi dibanding dengan TP dan FN. Hal ini dikarenakan TN dan FP diberikan input data *negative* dalam suatu kategori. Penelitian ini menggunakan total 494 *dataset* dengan 19 citra untuk masing-masing kategori dimana semua citra digunakan sebagai data latih. Citra yang digunakan sebagai data uji akan diambil secara langsung menggunakan kamera laptop dan diujikan sebanyak 3 kali pengujian. Sehingga pada saat pengujian ketika menguji kelas “A” maka akan diuji dengan 3 data positif dan 75 negatif. Begitu juga untuk kelas “B”, “C”, “D” dan seterusnya [17]. Setelah nilai TP, TN, FP, dan FN didapatkan, langkah selanjutnya untuk pengujian adalah menghitung parameter *accuracy*, *precision* dan *recall*. Adapun persamaan untuk mencari nilai *accuracy*, *recall*, dan *precision* adalah

$$accuracy = \frac{TP+TN}{TP+FP+FN+TN} \tag{1}$$

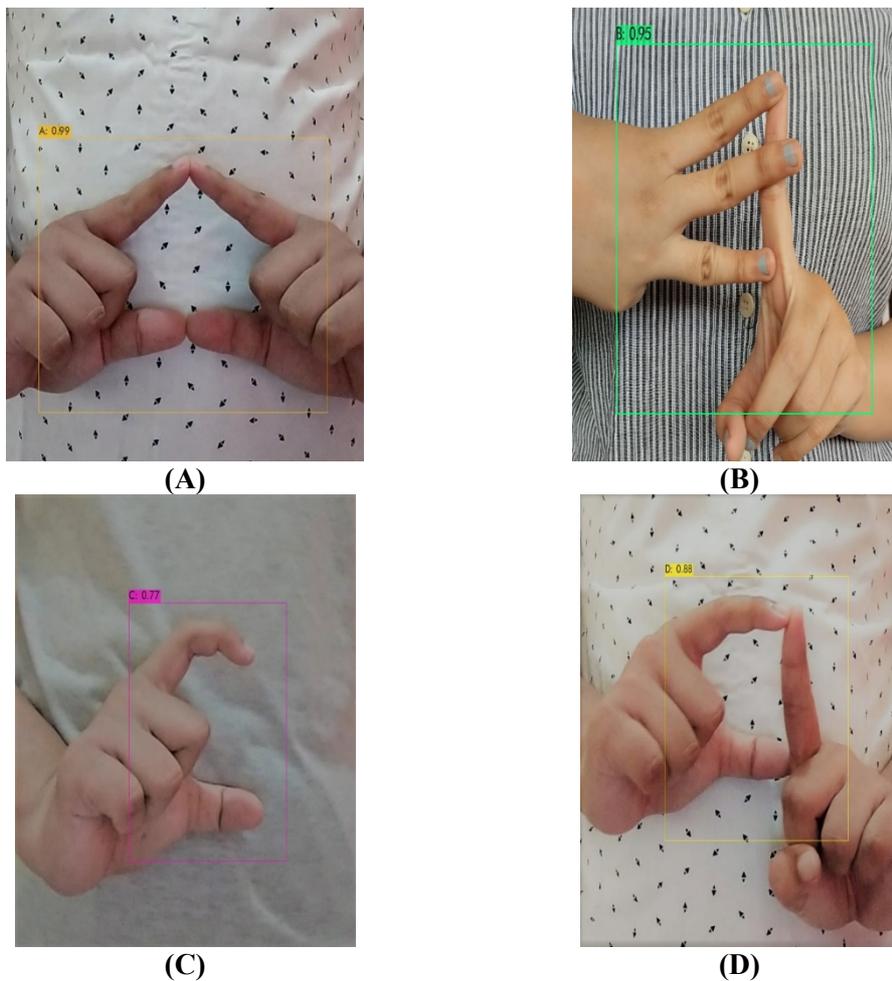
$$recall = \frac{TP}{TP+FN} \tag{2}$$

$$precision = \frac{TP}{TP+FP} \tag{3}$$

Penelitian yang dilakukan bertujuan mengenali kategori huruf yang ditampilkan oleh citra berdasarkan fitur yang dipelajari oleh sistem. Total kategori yang akan dikenali adalah 26 kategori terdapat pada huruf a hingga z. Parameter uji yang digunakan untuk menentukan performa dari sistem adalah parameter akurasi presisi dan *recall* yang dapat diamati pada persamaan 1, 2, dan 3. Parameter akurasi menunjukkan ketepatan pembacaan oleh sistem sedangkan presisi dan *recall* akan memberikan informasi terkait kecenderungan sistem dalam memberikan hasil pengenalan.

3. HASIL DAN PEMBAHASAN

Proses pengujian atau testing merupakan tahap terpenting dalam mengevaluasi kinerja suatu model deteksi objek. Testing bertujuan untuk menilai sejauh mana kemampuan sistem dalam mengenali objek secara keseluruhan dengan memberikan prediksi secara benar dan akurat. Pengujian data menggunakan citra sebagai *input*, model akan disesuaikan dengan distribusi dan karakteristik dari *dataset* pelatihan. Selama proses pelatihan atau *training*, model akan menyesuaikan bobot atau parameter lain yang nantinya digunakan untuk pengoptimalan model agar dapat memberikan prediksi yang akurat pada proses pengujian. Dengan melalui tahap pelatihan, citra-citra *training* dapat membantu model dalam mengidentifikasi pola serta fitur yang penting untuk tugas spesifik seperti pengenalan kelas objek dan ciri-ciri dari objek yang akan dikenali.



Gambar 4 *Dataset* Pengujian Huruf (A), Huruf (B), Huruf (C), Huruf (D)

Tabel 1 Hasil *Confusion Matrix*

Huruf Alfabet	TP	TN	FP	FN
A	3	75	0	0
B	3	75	0	0
C	3	75	0	0
D	2	74	1	1
E	3	75	0	0
F	3	75	0	0
G	3	75	0	0
H	3	75	0	0
I	3	75	0	0
J	3	75	0	0
K	3	75	0	0
L	3	75	0	0
M	2	74	1	1
N	2	74	1	1
O	3	75	0	0
P	2	74	1	1
Q	3	75	0	0
R	3	75	0	0
S	3	75	0	0
T	3	75	0	0
U	2	74	1	1
V	2	74	1	1
W	3	75	0	0
X	3	75	0	0
Y	3	75	0	0
Z	3	75	0	0
Total	72	1944	6	6

Hasil proses pengujian *object detection* melalui *image processing* akan menunjukkan tingkat keberhasilan dalam mengidentifikasi objek target. Pada penelitian ini, proses pengujian dilakukan sebanyak 78 kali, dimana masing-masing alfabet akan diuji 3 kali untuk tiap hurufnya. Pengujian dilakukan dengan mengambil citra baru melalui kamera laptop. Dalam deteksi objek, terdapat 2 konsep kunci yaitu *bounding box* atau kotak pembatas dan *confident rate* atau tingkat kepercayaan. *Bounding box* merupakan kotak pembatas yang akan mengelilingi objek untuk dideteksi dalam citra input, dimana akan memberikan informasi visual tentang area letak objek dalam gambar. Sedangkan *confident rate* menyediakan tingkat kepercayaan atau keyakinan model, ini membantu dalam menilai sebuah objek yang terdapat pada *bounding box* tersebut. Biasanya dinyatakan sebagai persentasi atau nilai dari 0 sampai 1.

Pada Gambar 4 merupakan beberapa hasil dari tahap pengujian *object detection* dalam mengenali huruf A, B, C, dan D alfabet BISINDO. Tabel 1 menunjukkan hasil pengujian untuk total 26 kategori alfabet BISINDO. Terlihat bahwa 3 citra kelas "A" yang diberikan untuk diuji oleh sistem dapat diprediksi dengan benar. Sedangkan dari 75 citra selain kelas "A" dideteksi sebagai citra bukan kelas "A". Sehingga total pengujian saat menguji kelas A didapatkan nilai TP adalah 3 dan nilai TN adalah 72. Sistem memiliki kesalahan prediksi pada kelas "P" dan kelas "D". Saat sistem diuji dengan citra "D" sebagai data positif, sistem memprediksi sebagai kategori "P" sehingga nilai FN pada kelas "D" bernilai 1. Saat sistem diuji dengan citra bukan "D", terdapat 1 kali sistem memprediksi sebagai huruf "D" sehingga nilai FP bernilai 1. Saat diuji menggunakan huruf "P" sebagai data positif sebanyak 3 kali, sistem memprediksi sebagai bukan "P" sebanyak 1 kali sehingga nilai TP pada kategori "P" adalah 2 dan nilai FN adalah 1. Saat diberikan input citra bukan "P" sebanyak 75 kali pengujian, terdapat 1 kali kondisi dimana sistem mengenali citra bukan "P" sebagai citra "P" sehingga nilai FP pada kategori "P" bernilai 1 dan

nilai TN pada kategori ini adalah 74. Kondisi yang sama terjadi antara huruf “M” dan “N” juga “U” dan “V”. Sistem masih tertukar antara kategori-kategori tersebut.

Berdasarkan data pada Tabel 1, Nilai *True Positive* (TP) pada pengujian ini adalah 72, nilai *True Negative* (TN) berjumlah 1944, nilai *False Positive* (FP) adalah 6, dan nilai *False Negative* (FN) adalah 6. Dengan menggunakan Persamaan 1 untuk menghitung *accuracy*, didapatkan nilai *accuracy* sebesar 99.4%. Untuk parameter berikutnya yaitu *recall*, juga dikenal sebagai *sensitivity* atau *true positive rate*, parameter ini digunakan untuk mengukur sejauh mana model dapat mengidentifikasi semua kasus positif yang sebenarnya. Semakin tinggi nilai *recall*, maka semakin baik model dapat mengidentifikasi entri positif yang sebenarnya. Dengan menggunakan Persamaan 2, didapatkan nilai *recall* sebesar 92,3%. Dalam konteks deteksi objek, parameter *precision* digunakan untuk mengukur seberapa akurat model dalam mengidentifikasi objek positif. Dengan menggunakan Persamaan 3, didapatkan nilai *precision* sebesar 92,3%.

Parameter *accuracy*, *recall*, dan *precision* yang didapatkan selama proses pengujian diketahui bahwa arsitektur YOLO memiliki kinerja yang baik. Penelitian sebelumnya [16] menggunakan arsitektur SSD *Mobilenet V2* memiliki nilai *accuracy* sebesar 71,2%, nilai *precision* sebesar 71,2%, dan nilai *recall* sebesar 74,7%. Penelitian lain [15] yang menggunakan menggunakan arsitektur yang sama memiliki nilai *accuracy* sebesar 54,8% dan tidak melampirkan nilai *recall* dan *precision*.

4. KESIMPULAN DAN SARAN

Berdasarkan data hasil dari penelitian yang dilakukan, didapatkan kesimpulan bahwa arsitektur *YOLOv4* dapat digunakan untuk mengidentifikasi gambar alfabet huruf BISINDO dengan baik. Pada penelitian ini model yang dibangun berdasarkan konsep *deep learning* dan pemanfaatan *YOLOv4* mampu untuk mendeteksi alfabet BISINDO dengan baik dan akurat sehingga dapat digunakan untuk membantu kelompok kaum tuna wicara dalam berkomunikasi dengan kaum dengar. Beberapa huruf memiliki potensi untuk tertukar dan salah baca namun performa yang diberikan sudah lebih tinggi. Pemanfaatan kamera dalam mendeteksi bahasa isyarat alfabet BISINDO dilakukan menggunakan *webcam* pada komputer, dan diakses melalui *library OpenCV* pada pemrograman *python*. Berdasarkan hasil uji performa dengan *YOLOv4* didapatkan nilai *accuracy* sebesar 99.4%, nilai *recall* sebesar 92,3%, dan nilai *precision* sebesar 92,3%. Performa ini menunjukkan bahwa tujuan penelitian yaitu menguji arsitektur *YOLOv4* untuk mengenali bahasa isyarat dapat dikenali dengan baik baik dan menjawab tujuan penelitian. Bahasa yang digunakan dalam penelitian ini adalah alfabet Bahasa Isyarat Indonesia (BISINDO). Agar semakin lengkap, penelitian berikutnya dapat menambahkan kelas baru yaitu angka Bahasa Isyarat Indonesia (BISINDO) dan berupa kalimat bahasa isyarat.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Lemabaga Penelitian dan Pengabdian Kepada Masyarakat (LPPM) Institut Teknologi Telkom Purwokerto karena telah memberikan fasilitas untuk melakukan penelitian.

DAFTAR PUSTAKA

- [1] A. M. Ambarak and A. Z. Falani, “Pengembangan Aplikasi Bahasa Isyarat Indonesia Berbasis Realtime Video Menggunakan Model Machine Learning,” *JIKA (Jurnal Inform.*, vol. 7, no. 1, p. 89, Feb. 2023, doi: 10.31000/jika.v7i1.7277.
- [2] F. Ndaumanu, “Hak Penyandang Disabilitas: Antara Tanggung Jawab dan Pelaksanaan oleh Pemerintah Daerah,” *J. HAM*, vol. 11, no. 1, p. 131, Apr. 2020, doi: 10.30641/ham.2020.11.131-150.
- [3] E. Tyas, “Gambaran Motif, Penggunaan Dan Kepuasan Pada Media Massa Sebagai Sumber Informasi Kesehatan (Studi Deskriptif Pada Siswa SMA Luar Biasa Tunarungu Karya Mulia Surabaya),” *J. Kesehat.*, vol. 15, no. 2, pp. 128–132, Aug. 2022, doi:

- 10.32763/juke.v15i2.415.
- [4] R. W. Bharata, U. Rani, N. Priyono, and I. Novitaningtyas, “Analisis Partisipasi Penyandang Disabilitas Dalam Perencanaan Pembangunan di Kabupaten Gunungkidul,” *Integralistik*, vol. 32, no. 2, pp. 83–88, Jul. 2021, doi: 10.15294/integralistik.v32i2.28818.
 - [5] L. L. Linda and A. Muliastari, “Analisis Kebutuhan dan Perilaku ABK Tunarungu dan Wicara dalam Pembelajaran Matematika Dasar di SKh Kabupaten Pandeglang,” *JP3M (Jurnal Penelit. Pendidik. dan Pengajaran Mat.)*, vol. 7, no. 1, pp. 09–22, Apr. 2021, doi: 10.37058/jp3m.v7i1.2145.
 - [6] E. Y. Lestari and S. Sumarto, “Pemenuhan Hak Bekerja Bagi Penyandang disabilitas di Kota Semarang,” *Integralistik*, vol. 32, no. 1, pp. 45–51, Jan. 2021, doi: 10.15294/integralistik.v33i1.28731.
 - [7] M. Jannah, “Tomato Maturity Detection System Using Color Histogram Method And Nearest Neighbor,” *JAICT*, vol. 7, no. 1, p. 63, Mar. 2022, doi: 10.32497/jaict.v7i1.3074.
 - [8] Lukman Priyambodo *et al.*, “Klasifikasi Kematangan Tanaman Hidroponik Pakcoy Menggunakan Metode SVM,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 1, pp. 153–160, Feb. 2022, doi: 10.29207/resti.v6i1.3828.
 - [9] M. A. Afandi, S. I. Purnama, and R. F. Crisianti, “Implementasi Metode Deteksi Tepi Laplacian dan Jarak Euclidean untuk Identifikasi Tanda Tangan,” *J. Nas. Tek. ELEKTRO*, vol. 9, no. 1, p. 34, Mar. 2020, doi: 10.25077/jnte.v9n1.756.2020.
 - [10] R. F. Christianti, H. L. Fuadi, M. A. Afandi, A. S.N., and A. Dharmawan, “Comparison of Support Vector Machine and Neural Network Algorithm in Drone Detection System,” in *2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom)*, Jun. 2022, pp. 421–426, doi: 10.1109/CyberneticsCom55287.2022.9865628.
 - [11] A. A. Ramadana Lubis, S. I. Purnama, and M. A. Afandi, “Sistem Pendeteksi Kantuk Berbasis Metode Haar Cascade Untuk Aplikasi Computer Vision,” *Techno.Com*, vol. 22, no. 3, pp. 589–598, Aug. 2023, doi: 10.33633/tc.v22i3.8464.
 - [12] Octavian Ery Pamungkas *et al.*, “Classification of Rupiah to Help Blind with The Convolutional Neural Network Method,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 2, pp. 259–268, Apr. 2022, doi: 10.29207/resti.v6i2.3852.
 - [13] A. Nugroho, R. Setiawan, A. Harris, and Beny, “Deteksi Bahasa Isyarat Bisindo Menggunakan Metode Machine Learning,” *J. Process.*, vol. 18, no. 2, Nov. 2023, doi: 10.33998/processor.2023.18.2.1380.
 - [14] L. Arisandi and B. Satya, “Sistem Klarifikasi Bahasa Isyarat Indonesia (Bisindo) Dengan Menggunakan Algoritma Convolutional Neural Network,” *J. Sist. Cerdas*, vol. 5, no. 3, pp. 135–146, Dec. 2022, doi: 10.37396/jsc.v5i3.262.
 - [15] Nasha Hikmatia A.E. and M. I. Zul, “Aplikasi Penerjemah Bahasa Isyarat Indonesia menjadi Suara berbasis Android menggunakan Tensorflow,” *J. Komput. Terap.*, vol. 7, no. 1, pp. 74–83, Jun. 2021, doi: 10.35143/jkt.v7i1.4629.
 - [16] L. N. Hayati, A. N. Handayani, W. S. G. Irianto, R. A. Asmara, D. Indra, and M. Fahmi, “Classifying BISINDO Alphabet using TensorFlow Object Detection API,” *Ilk. J. Ilm.*, vol. 15, no. 2, pp. 358–364, Aug. 2023, doi: 10.33096/ilkom.v15i2.1692.358-364.
 - [17] M. Grandini, E. Bagli, and G. Visani, “Metrics for Multi-Class Classification: an Overview,” Aug. 2020, [Online]. Available: <http://arxiv.org/abs/2008.05756>.